Long Term Activity Analysis in Surveillance Video Archives

Ming-yu Chen

CMU-LTI-10-015

September 12, 2010

Language Technologies Institute
School of Computer Science
Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA, 15213
Pittsburgh, PA 15213

Thesis Committee:

Alexander Hauptmann, Chair
Jie Yang
Rahul Sukthankar
Yihong Gong, Akiira Media Systems, Inc.

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
In Language and Information Technologies.

Keywords: Activity Analysis, Surveillance Video, Computational Perception

For my parents, Mong-fu Chen and Yuan-ling Yu.

Abstract

Surveillance video recording is becoming ubiquitous in daily life for public areas such as supermarkets, banks, and airports. The rate at which surveillance video is being generated has accelerated demand for machine understanding to enable better content-based search capabilities. Analyzing human activity is one of the key tasks to understand and search surveillance videos. In this thesis, we perform a comprehensive study on analyzing human activities from short term to long term and from simple to complicated activities in surveillance video achieves.

A general, efficient and robust human activity recognition framework is proposed. We extract local descriptors at salient points from videos to represent human activities. The local descriptor is called Motion SIFT (MoSIFT) which explicitly augments appearance features with motion information. A quantization and classification framework then applies the descriptors to recognize activities of interest in surveillance videos. We further propose constraint-based clustering ,bigram models, and a soft-weighting scheme to improve the robustness and performance of the algorithm by exploring spatial and temporal relationships between local descriptors. Detection is another essential task of surveillance video analysis. The difficulty of detection lies in identifying the temporal position in a video . Therefore, we propose a sliding window approach to search candidate positions with cascade classification to reduce false positives. Finally, we perform a study to utilize automatic human activities analysis to improve geriatric health care. We explore the statistical patterns between a patient's daily activity and his/her clinical diagnosis. Our main contributions are an intelligent visual surveillance system based on efficient and robust activity analysis and a demonstration exploring long term human activity patterns though video analysis.

Acknowledgments

First of all, I would like to thank my advisor, Alex Hauptmann, for his great guidance and support over the past seven years. I have learned not only the way to approach a hard problem but also been inspired by his passions for multimedia research. His insights have shaped my Ph.D study and my thesis topic. I am especially thankful for the freedom that I have to explore various research topics and to collaborate with different people outside the group. I couldn't imagine a more ideal advisor than Alex.

I would also like to thank my committee members, Rahul Sukthankar, Jie Yang and Yihong Gong, for their advice and feedback on the thesis. It's their comments and suggestions that make this thesis more accurate and more complete. They are also great models of how to be successful in this field.

I have been fortunate to work closely with colleagues in the Informedia project, Howard Wactlar, Michael Christel, Ashok Bharucha, Robert Baron, Datong Chen, Rong Jin, Wei-hao Lin, Rong Yan, Jun Yang, Tim Pan, and Bryan Maher. With them, I had many insightful discussions, joint publications, and collaborative projects. Moreover, it has been my pleasure to know many good friends and fellow students at CMU, including Stan Jou, Bill Chou, Ariel Lee, Eddy Liu, Ray Shih, Huan Li, Betty Cheng, Yi-jan Ho, Mike Tsang, Frank Wang, Stanley Chang, Roger Chang and many more. Their friendship and support make my Ph.D life pleasant and wonderful. I also own many thanks to my best friends, Alex Wu and Vanessa Chen, for their long-distance support during these years.

Last, it is always not enough to express my appreciation to my parents, my brother, and my partner Yi-fen for their unconditional love and support. Without this I would not have survived the long journey of my Ph.D study.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

In this thesis, we study the human activity analysis problem and we especially focus on large surveillance video archives. Human activity analysis is to understand activities which people are performing in videos. The goal of human activity analysis is to identify interested human activities in noisy environments and various circumstances. We especially target real world surveillance scenarios which contain large amounts of data and also have diverse and complex environments. Automatic human activity analysis can not only detect interested activities but also provide a way to understand the video content. Furthermore, we want to utilize the informative analysis results to understand videos over long periods of time and be able to explore long term activity patterns.

We propose to characterize human activities in surveillance video though the use of spatio-temporal interest points. A spatio-temporal interest point is an area of interest containing a distinguishing shape and sufficient motion. A descriptor is a feature extracted to describe both shape and motion around an interest point. Each interest point captures and represents small but informative components of an activity in the video. The small components can be raising a finger, bending a knee or lips moving. We assume that an activity can be described though a combination of different types of these small components. Since interest points are small, they can capture local movements and are less affected by posture, illumination and occlusion. Therefore, the task of comparing the similarity of two activities transforms into a search for similar, conceptually meaningful components exhibited in the video.

Furthermore, we propose a sliding window approach with a cascade of classifiers to attack the challenge that the same activities can deform significantly in shape and length. The reason to introduce multiple scale sliding windows is to scan though all possible locations and times. The sliding window approach generates a tremendous amount of negative windows and increases the false positive rate in the detection task. Cascade architecture is a approach to not only keep strong detection rate but also significantly reduce false positive rate.

Finally, we perform a study to utilize automatic human activity analysis to improve geriatric health care. Geriatric health care is improved by observing elder patients' daily living to predict or prevent their physical and mental illness. However, it requires a tremendous amount of human effort to keep tracking a patient's daily living. A patient's health condition can not be evaluated in a short period of time. Therefore, automatic long term activity analysis is an emerging research topic in the health care domain. We explore the statistical patterns between patient daily activities and clinical diagnoses to assist better health care. The promising experimental result directly supports the idea that even imperfect human activity analysis can still provide strong evidence to assist medical doctors in understanding elder patients' long term patterns and improving their diagnoses.

1.1 Motivation

Visual surveillance is omnipresent in our daily life. Some systems are set up for security proposes such as video recording in banks and ATMs. Some systems are designed for access control to restricted areas, e.g. to permit face identification at an entrance. Some systems aim to perform congestion analysis such as surveillance systems at highways or major streets. These surveillance systems collect a huge amount of video but most of the data needs to be reviewed by a human operator to extract informative knowledge. Currently, many research efforts focus on developing intelligent visual surveillance systems to replace traditional passive video surveillance systems which can only store surveillance videos but are not able to identify or describe interesting activities.

Most surveillance tasks focus on human activities. Therefore, human detection, human movement tracking, human activity recognition and person identifi-

Figure 1.1: Surveillance video recording is omnipresent in our daily life. They are monitoring public indoor areas, e.g. banks, airports and ATMs, and outdoor areas, e.g. traffic intersections.

cation are popular topics in computer vision. A general intelligent visual surveillance system framework usually includes the following stages: modeling environments, detecting motions, classifying moving objects, tracking, understanding and describing human activities, and human identification. We will especially focus on human activity analysis suitable for large archives of video surveillance data. There are a lot of well known difficulties in automatic activity characterization: Activities under observation can vary in posture, appearance, scale, background, and occlusions which make activity analysis extremely difficult.

Moreover, there is an important and exciting problem in the video analysis domain. What is the basic semantic unit to express the content of the video? In text documents, there are words and phrases to represent the semantic concepts. Researchers have proposed many efficient algorithms to categorize, index, retrieve and summarize documents though words and phrases. However, lack of basic semantic units makes it a big challenge to access video content efficiently. Human activities are usually the essential part in most video content. A robust human activity analysis can further provide reliable semantic units to represent the video

content.

In this thesis, we especially focus on the human activity analysis problem in the clinical domain, specifically a nursing home surveillance video archive. In a nursing home, one staff member needs to take care of several elderly patients and provide doctors with daily observations to assist treatment diagnoses. Although the staff have professional training and are able to observe clinical information from patients' daily living, they can not focus their attention on the patients every single second. Surveillance video recording is currently only a marginally useful tool to staff and doctors. Therefore, we want to design a system that not only records but also performs analysis tasks. In a nursing home environment, we want to detect unusual activities and also recognize patients' routine activities, e.g. eating, chatting, etc. In the end, the detection results can be analyzed and will provide long term activity patterns to assist doctors.

The potential benefits of human activity analysis apply not only to surveillance video but also to other areas. Video activity understanding can be widely used in many applications such as video retrieval, video gaming, video conferencing, and vision-based user interfaces. Our approach can be extended to analyze various activities in different circumstances, e.g. scoring goals in sports videos, controlling TVs and video games with gestures, detecting car accidents in the street etc. We believe though the study of activity analysis, we can develop semantic descriptors to assist others in accessing video content efficiently.

1.2 Thesis Statement

In this thesis, we aim to attack two major tasks in video analysis. The first task is to develop techniques for robust and accurate human activity analysis based on real-world surveillance video archives. The second task is to extend activity analysis to describe human behaviors over a long period of time.

To robustly and accurately analyze human activity, our approach is inspired by object recognition approaches which rely on sparsely detected features to characterize an object. We extract spatio-temporal descriptors called MoSIFT at salient points from the video to represent human activities. These video descriptors decompose complicated human activities into small location-independent units. We

then propose a constraint-based clustering algorithm to cluster video descriptors into conceptually meaningful sets and improve the quantization process. A bi-gram model is also proposed to capture structure information of activities to make the algorithm more robust. A bag-of-word feature is then constructed for each video clip to represent its content. A soft-weighting scheme is applied to improve the traditional bag-of-word representation directly borrowed from text domain. A classification framework applies the bag-of-word features to recognize activities of interest in surveillance video. Furthermore, a brute-force scan and cascade classifier approach is applied to extend the activity recognition framework into a detection framework.

Detecting and recognizing human activities in a video provides fundamental tools for users to analyze the content in that video. Current video analysis techniques detect or recognize a short term activity. Surveillance video systems often record a long period of time and this continuous recording provides valuable information. Analyzing long term activity is a very challenging task and it is domain specific. In this thesis, we especially focus on elderly patient health care since it has become an growing need in our aging society. We demonstrate that automatic video analysis of patients' daily lives over time is informative to a doctor's diagnosis and is able to further improve the quality of life to nursing home residents. This case study shows a promising research direction for the multimedia community.

1.3 Thesis Contribution

This dissertation makes four contributions in computer vision and multimedia analysis.

- The first contribution is to develop a robust video feature descriptor, MoSIFT, and a solid activity recognition framework. MoSIFT explicitly describes both appearance and motion of an interest region at multiple scale from a video. The activity recognition framework consists of interest point extraction, video codebook construction/mapping, bag-of-word feature representation, and modeling. The constraint-based clustering, bigram and soft-weighting scheme are introduced to enhance the bag-of-word representation

to improve recognition performance. Detecting and describing motions explicitly improves the activity recognition performance significantly. Efficient bag-of-word representation gives us the ability to build a recognition system on hundred hours of video.

- The second contribution comes from building an activity detection framework. A brute-force search strategy is achieved by sliding a fixed length window over a video to generate candidate windows. A cascade SVM classifier is built to identify interesting activities among all the candidate windows. The false positive rate is decreased by the good properties of the cascade architecture and concatenating positive prediction strategy. This algorithm has the top performance in official surveillance video event detection benchmark in TRECVID [86].
- The third contribution comes from a successful case study in analyzing the long term activity from a surveillance video achieve in the nursing home health care domain. A long term activity analysis is domain dependent and there is no general solution. The case study we perform in the CareMedia [90] project is to detect activities in residents' daily lives over time to better estimate their health conditions. We demonstrate that observations in surveillance video are informative. Furthermore, we successfully simulate automatic video analysis and prove the inaccurate automatic video analysis over a long period of time can assist medical doctors to estimate patients' health conditions more accurately. This work as we know is the first to demonstrate that the video surveillance can assist health care by observing patients over time.
- The fourth and last contribution is to build two video analysis applications to demonstrate that the proposed techniques are practical. We successfully parallelize MoSIFT activity recognition by the Sprout [70] architecture to achieve real time activity analysis. This technique enables us to build real-world applications. We demonstrate the proposed activity analysis techniques in two aspects: a interactive interface and a intelligent store surveillance system. The success in building these real-world applications gives us confidence that the proposed work can be applied to many emerging ar-

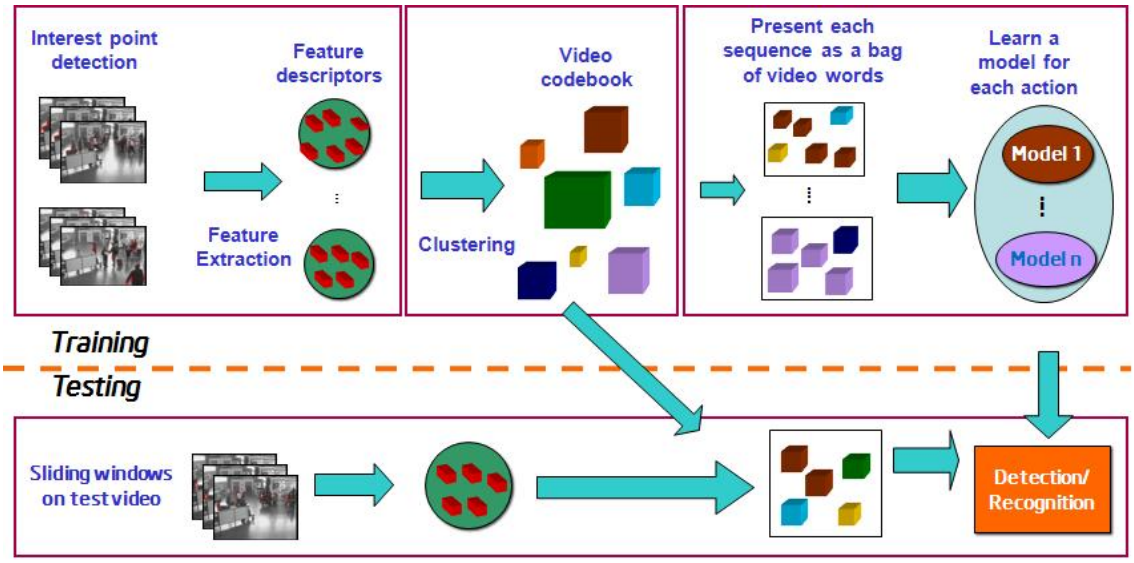

Figure 1.2: System framework of visual activity recognition/detection. There are three major steps in the training phrase: local feature extraction, video codebook construction, activity model training. The test video will be mapped by video codebook and be classified into associated activities.

eas, e.g. content-based video retrieval, traffic load analysis, tracking, day care surveillance, etc. Given the exponential growth of video content, our proposed techniques help users to access video content efficiently.

1.4 Visual Activity Analysis

In this thesis, our framework of visual activity analysis is based on a local feature approach. Local feature (interest point) approaches, such as SIFT, have demonstrated great successes in object recognition/detection. An interest point is a point in the image/video which has several desired properties. First, the local structure around the interest point should be rich in terms of local information content. Second, the interest point should be stable under local and global perturbation, including deformations from perspective transformation as well as illumination/brightness variations. Given these properties, the interest points can be reliably computed with a high degree of reproducibility.

Figure 1.2 illustrates the framework of an activity analysis system. We define

activity analysis as comprehensive activity recognition and detection. In terms of comprehensiveness, we want to detect an activity and recognize it regardless of its form and duration. The form of a human activity can be roughly described by three categories: single person, person with object, and multiple persons. Each form has very different appearances and characteristics. The duration of a human activity can vary from a couple of seconds to several minutes. These variations make the activity analysis a challenging task. In our framework, we apply a local feature approach to visual activity analysis. In a local feature approach, there are three major parts: local feature extraction, video codebook construction and activity model training. Local feature extraction has two key tasks: interest point detection and description. The local feature extraction method we developed, MoSIFT, not only detects and describes interest points in local appearance from spatial and temporal domains but also further captures explicit motion information. Video codebook construction is a quantization process to transfer arbitrary numbers of interest points from video segments into fixed length feature vectors. An activity model is then trained by a machine learning algorithm. We apply a Support Vector Machine (SVM) [17] here due to its robust and solid performance.

Originally, this framework was designed to accomplish a recognition task. A recognition task identifies a specific video pattern such as people running in a video segment. The assumption of the recognition task is that a video segment is provided and it should be classified as a given activity. A detection task is to localize and identify the pattern in a video. To extend our framework to achieve detection, we build a fixed length sliding window to scan through the video. Each sliding window is a video segment to which we can apply our method and recognize the desired activity. However, the sliding window approach normally generates a tremendous amount of potential examples and the target activity we want to detect is usually very rare in the video. This fits well into the framework of cascade classifiers which have been proven to significantly reduce the false positive rate.

1.5 Long Term Activity Analysis

Beside comprehensive activity analysis, we would like to further explore possible ways to utilize these analysis results to understand long term changes or trends. This work is valuable in many areas. For example, we can model customers' shopping behaviors via surveillance cameras which are common in a lot of stores. Over a long period of time, we would be able to analyze customers' shopping trends by observing touching, surveying, and trying products in stores. In our study, we will focus on geriatric health care to explore long term activity analysis. Figure 1.3 shows the conceptual overview of geriatric patient behavior monitoring and analysis. In this thesis, we focus on activity analysis from surveillance video and employ a case study on long term activity analysis to predict patients' health conditions.

In our study, we try to show that comprehensive activity analysis results are strongly correlated with doctors' diagnoses. In geriatric domain, diagnoses are based on several evaluation methods which are proved to strongly reflect patients' health conditions in the medical domain [5, 22, 23, 51, 61, 69]. Our promising results give us confidence that surveillance video can further assist doctors to make more accurate diagnoses. This study employs an example to demonstrate that we can analyze long term activity with surveillance videos.

1.6 Datasets

In this thesis, we will evaluate our methods and analysis on five video datasets: the KTH dataset [78], the Hollywood dataset [50], the Gatwick Airport Surveillance video archive [85], the TRECVID 2009 Sound and Vision dataset [86], and the CareMedia dataset [82, 90]. The KTH and Hollywood are standard datasets used by researchers to evaluate activity recognition performances. The Gatwick archive was collected for activity detection tasks and features a complicated real world environment. The Sound and Vision collection is widely used to evaluate video analysis tasks, e.g. semantic video feature extraction and video retrieval. The CareMedia dataset is mainly used to explore long term activity analysis and is also captured in a complex real world environment.

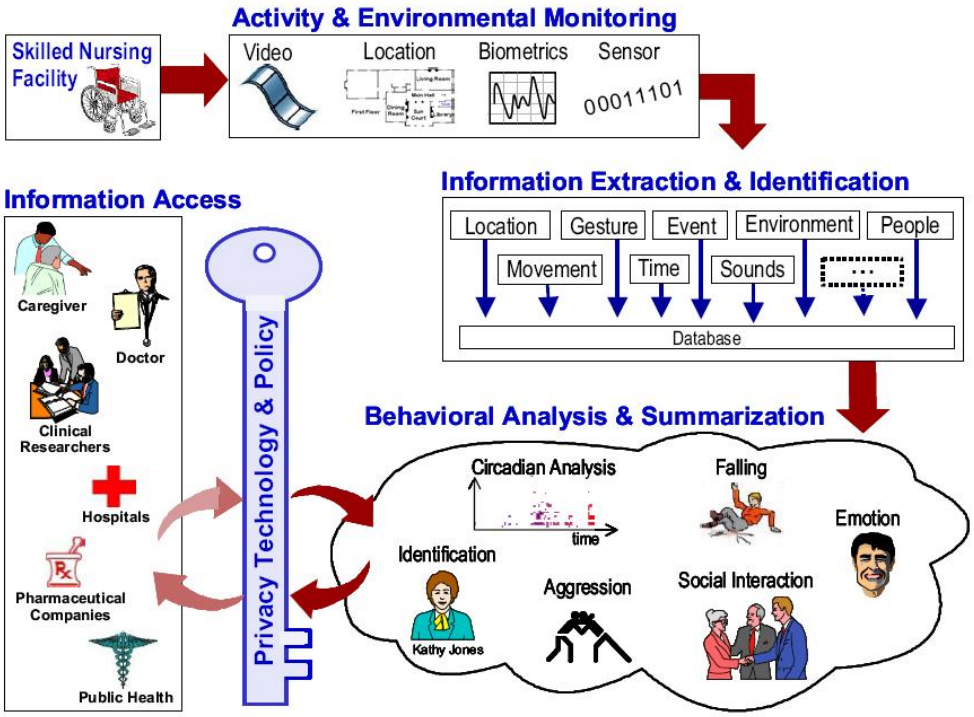

Figure 1.3: Conceptual overview of geriatric patient behavior monitoring and analysis. The ultimate goal is to extract various information from multiple sources, analyze social interactions and interested behaviors, and provide an information access to medical doctors. In this thesis study, we focus on activity analysis from surveillance video and employ a case study on long term activity analysis to predict patients' health conditions.

Dataset	# activities	# examples	Size	Description
KTH [78]	6	598	2 hours	Static background. Standard dataset.
Hollywood [50]	8	663	64+ hours	Movie scenes. Camera motions. Edited cuts.
Gatwick [85]	10	14081	100+ hours	Static background. Surveillance video.
Sound and Vision [86]	20	93902	380 hours	TV programs.
CareMedia [90]	19	6904	14976+ hours	Static background. Surveillance video.

Table 1.1: Dataset used in our experiments. In CareMedia dataset, we only use the examples from one chosen camera during dining periods.

1.6.1 KTH

The KTH human activity dataset is widely used by researchers to evaluate activity detection and recognition [28, 29, 43, 47, 50, 54, 60, 64, 67, 72, 76, 78, 83, 92, 93]. The dataset contains six types of human actions (**walking**, **jogging**, **running**, **boxing**, **hand waving**, and **hand clapping**) performed by 25 different persons. Each person performs the same action four times under four different scenarios (*outdoors*, *outdoors at a different scale*, *outdoors with camera moving*, and *indoors*). The whole dataset contains 598 video clips and each video clip contains only one action. In KTH, each action is performed by a single person in a relatively simple environment. The KTH dataset provides a common benchmark to evaluate and compare activity detection and recognition algorithms. Figure 1.4 gives some examples from KTH dataset. In the figure we can see that several actions are quite similar, such as jogging and running, and this makes the dataset more challenging.

1.6.2 Hollywood

The Hollywood dataset contains video samples with human action from 32 movies. Each sample is labeled according to one or more of 8 action classes: (**Answer Phone**, **Get Out Car**, **Hand Shake**, **Hug Person**, **Kiss**, **Sit Down**, **Sit Up**, and **Stand Up**). The dataset is divided into a test set from 20 movies and two training

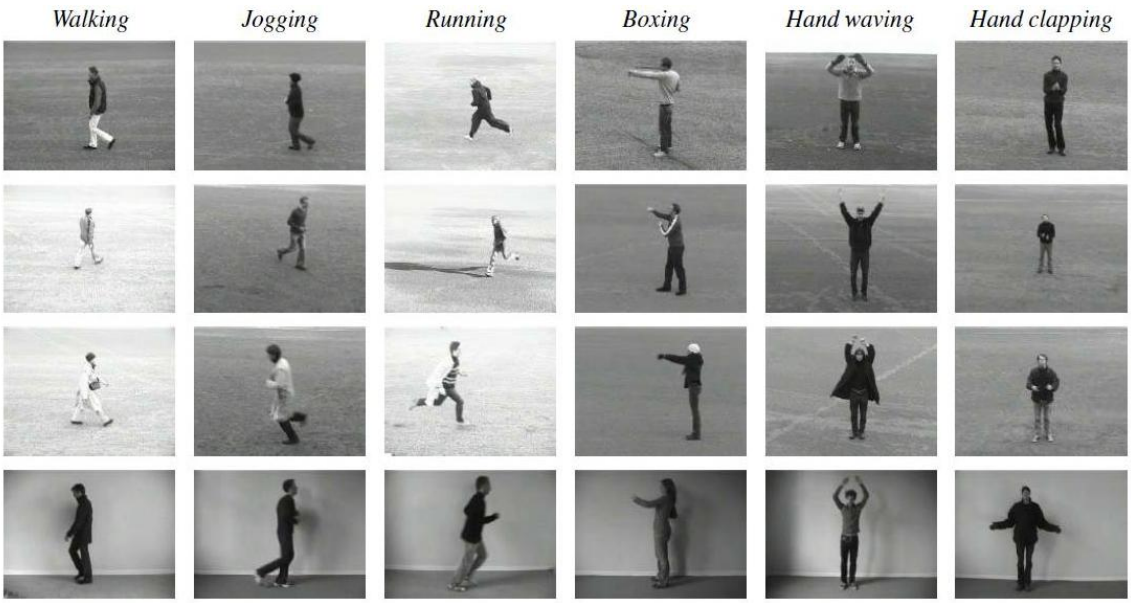

Figure 1.4: Some examples of the KTH dataset. Figure adapted from [78]

sets of 12 movies different from the test set. The *Automatic* training set is obtained using automatic script-based action annotation and contains 233 video samples with approximately 60% correct labels. The *Clean* training set contains 219 video samples with manually verified labels. The test set contains 211 samples with manually verified labels. Figure 1.5 shows some examples from the Hollywood dataset. The dataset is frequently used to evaluate human action recognition algorithms and is more challenging than the KTH dataset due to camera motion, cluttered backgrounds and various deformation of interesting activities.

1.6.3 Gatwick

The TRECVID 2008 [85] surveillance event detection dataset was recorded of London Gatwick International Airport provided by NIST [65]. It consists of 50-hours (5 days x 2 hours/day x 5 cameras) of video in the development set and another 50-hours in the evaluation set. There are around 190K frames per 2-hour video with an image resolution 720 x 576. This dataset contains highly crowded scenes, severely cluttered background, large variation in viewpoints, and very different expressions of the same activities; all embedded in a huge amount of data. To-

Figure 1.5: Some examples of the Hollywood dataset. The first row shows ”kiss” activities. The second row demonstrates ”Answer Phone” activities. The bottom row shows ”Get out Car” activities. Figure adapted from [50]

gether, these characteristics make activity detection on this dataset a formidable challenge. To the best of our knowledge, human activity detection on such a large, challenging dataset with these practical concerns has not been evaluated and reported prior to TRECVID 2008. In this dataset, 10 human activities are evaluated:

(**Object Put**, **People Meet**, **People Split Up**, **Pointing**, **Cell To Ear**,

Embrace, **Person Runs**, **Elevator No Entry**, **Take Picture**, and **Opposing Flow**).

Standardized annotations of activities in the development set were provided by NIST [65]. In this dataset, NIST uses the term ”event” instead of activity. A video event usually indicates a visible incident performed by human in a video which is actually an human activity. To be consistent in this thesis, we will use the term ”activity” to reduce confusion. Figure 1.6 shows all five camera views in the Gatwick dataset.

Figure 1.6: Some example views of the Gatwick dataset. Each example corresponds to a different camera.

1.6.4 Sound and Vision

The 2009 TRECVID [86] Sound and Vision dataset was collected to perform high-level feature extraction and retrieval tasks. In video content retrieval, high-level (semantic) features are believed to be important meta-data to enable searching in video content [34]. Among possible semantic features, some can be detected by still images but many can be only analyzed from appearance with motions. In the TRECVID 2009 evaluation, the dataset contain 280 hours of videos; 100 hours of videos for training and the other 180 hours for evaluation. Twenty concepts were evaluated by concept recognition performance: (**Airplane flying**, **Boat and ship**, **Bus**, **Cityscape**, **Classroom**, **Demonstration or protest**, **Hand**, **Nighttime**, **Singing**, **Telephone**, **Chair**, **Infant**, **Traffic intersection**, **Doorway**, **Person playing musical instrument**, **Person playing soccer**, **Person riding a bicycle**, **Person-eating**, and **Female human face closeup**). Among those concepts, many can be recognized by analyzing human activity or motions. The Sound and Vision dataset is a collection of news magazine, science news, news reports, documentaries, educational programming and archival videos by Netherlands Institute of Sound and Vision. This dataset contains a lot of variety and we want to demon-

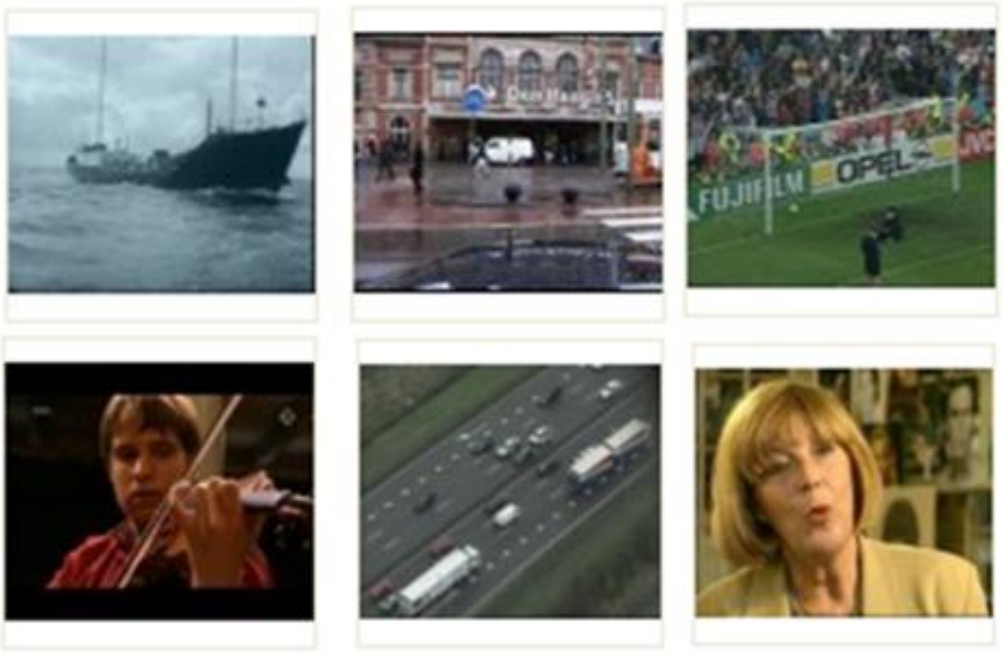

Figure 1.7: Some examples of TRECVID 2009 Sound and Vision dataset. For the first row, from left to right are ”Boat and Ship”, ”Doorway”, and ”Person playing soccer”. For the second row, from left to right are ”Person playing musical instrument”, ”Bus”, and ”Female human face closeup”.

strate our proposed algorithm is solid to analyze the real world video programs. Figure 1.7 shows some examples from the Sound and Vision dataset.

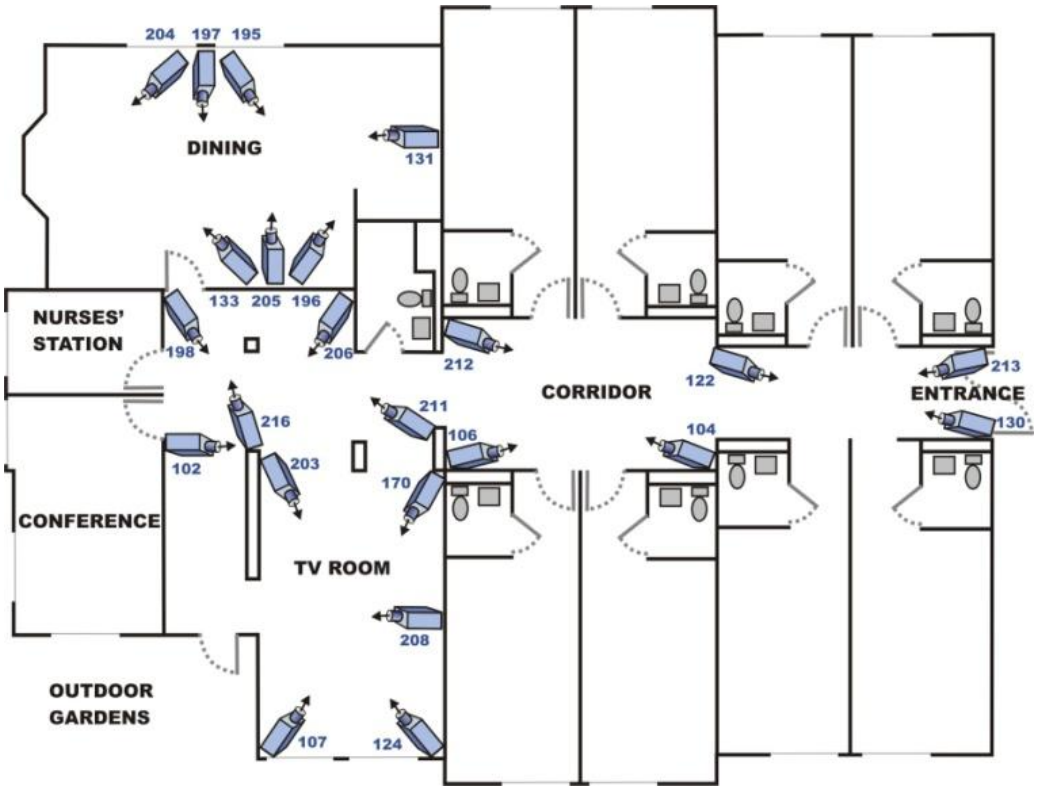

Figure 1.8: Camera placement in the nursing home in the CareMedia dataset.

1.6.5 CareMedia

The CareMedia dataset is a surveillance video data collection from a geriatric nursing home collected by the Carnegie Mellon University Informedia group. We placed 23 cameras in public areas such as the dining room, TV room, and hallway in the nursing house. We recorded patients' lives for 25 hours per day for 25 days with 23 cameras. The recording is at 640x480 resolution and 30 fps MPEG-2 format. In total we collected over 13,000 hours of videos which occupy about 25 terabytes. Figure 1.8 shows the camera set up in the nursing home. Figure 1.9 gives some examples showing the environment in the nursing home. From this dataset, we specifically choose camera 133 in the dining room as our evaluation

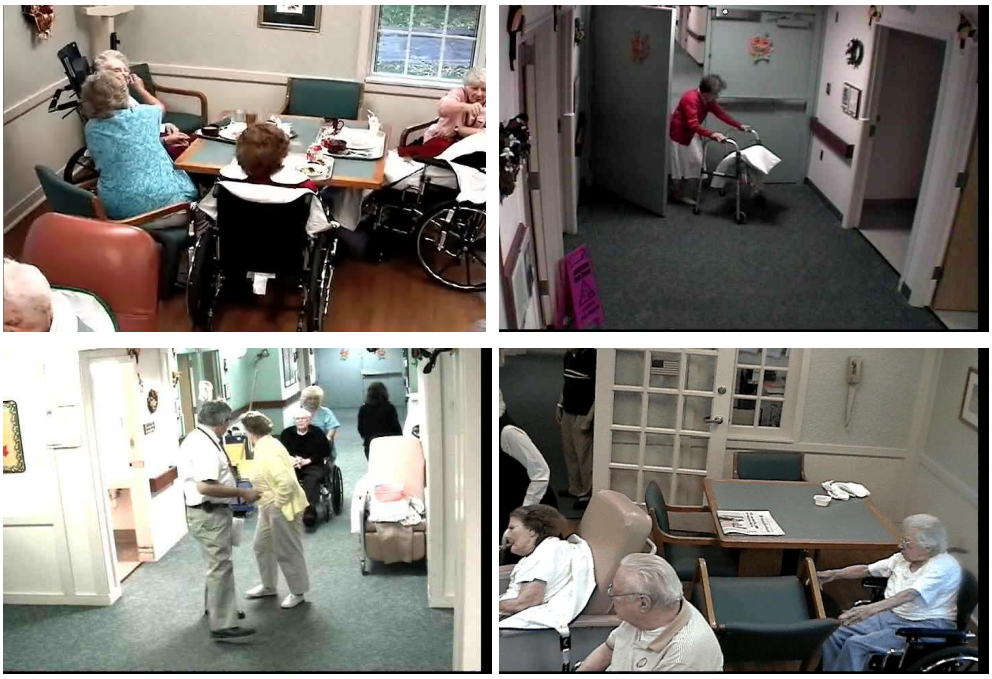

Figure 1.9: Some examples of the CareMedia dataset. In the first row, from left to right are ”Staff activity: Feeding” and ”Walking though” activities. In the second row, from left to right are ”Wheelchair movement” and ”Physically aggressive: Pulling or tugging”.

set. This camera captures patients' activities during lunch and dinner time. In total, we have 6904 activities annotated in this evaluation set. From the examples shown in Figure 1.9, the CareMedia dataset is a very challenging dataset which contains crowded scenes, severely cluttered background, large variance in viewpoints, very different performances of the same activities, and severely changing illumination. The tempo of patients' activity is much slower than usual which creates a big challenge for robust activity analysis.

1.7 Applications

Human activity analysis is a fundamental function of video understanding. A robust and stable activity analysis algorithm could be widely used in many video applications. We will discuss two different applications in this dissertation. One is an intelligent surveillance video system which not only records surveillance video but also shows activity detection results to help the surveillance administrator easily catch interesting events in the video. The other set of applications we will demonstrate here are vision based interactive applications. The system can detect and recognize human activities such as gestures as control input. It can be applied to video gaming, TV control, and interactive computer input methods.

1.7.1 Intelligent Surveillance Video Systems

Figure 1.10 shows the interface of an intelligent surveillance video system for Gatwick airport surveillance videos. The system is able to detect and summarize a set of pre-defined human activities. A threshold bar can be set to control the amount of data you want to analyze. It is a advanced surveillance system that saves a surveillance administrator a tremendous amount of time. A robust visual human activity analysis algorithm is a key component in this intelligent surveillance video system. In our chapter on applications (Chapter 7), we demonstrate another intelligent surveillance video application which analyzes customers' shopping behaviors in a shopping store.

1.7.2 Interactive Applications

Interactive vision-based applications require not only robust visual activity analysis algorithms but also low latency. Currently, it is computationally expensive to achieve robust visual activity analysis. Parallelism and cluster-based distributed systems now can improve these vision-based systems not only in terms of throughput but also latency. Figure 1.11 demonstrates a system which detects human gestures to control a television at interactive speeds. This implementation gives us confidence that the visual activity analysis technique could be practical in our life soon.

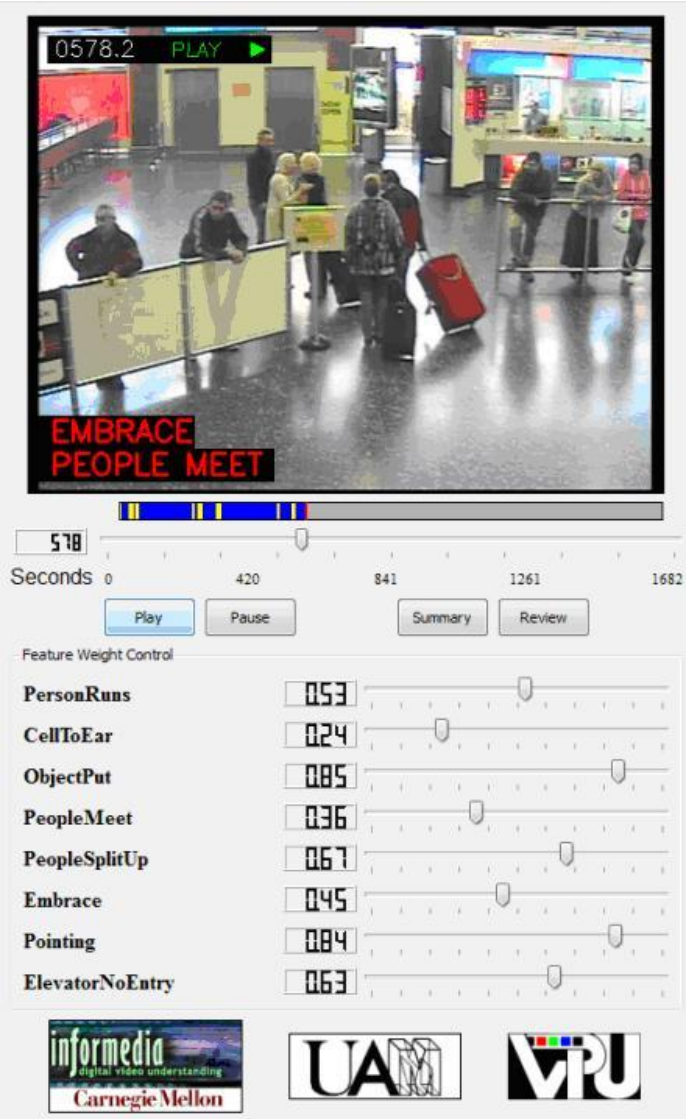

Figure 1.10: A intelligent surveillance video system on the Gatwick airport surveillance video. Our system detects specific activities and users can set up thresholds to show specific activities or summarize surveillance videos. The application can speed up video play and fast forward when there isn't any interesting activity.

Figure 1.11: Setup of TV/camera for gestural control system.

Chapter 2

Related work

Automatic analysis and interpretation of human activities have received a great deal of attention from both industries and academic research in recent years. This is motivated by many real-world surveillance applications that require tremendous amounts of observation by human operators. An intelligent surveillance system is usually composed of computer vision and information retrieval techniques. In computer vision, environment modeling, motion segmentation, object classification, tracking, activity understanding and person identification are all active research topics. In information retrieval, data mining, question answering and information summarization can provide essential tools to access the surveillance data efficiently. Human activity detection and recognition are the core techniques in visual surveillance systems. Researchers are looking to develop robust video concept detection and recognition which is a strong semantic basis for further video search and mining. In activity detection and recognition analysis, there are three main approaches: Model-based, Appearance-based and Part-based methods. In information retrieval, semantic concept detection is a popular research topic that includes much image and video analysis research. Furthermore, the TRECVID event detection task provides a platform for researchers to evaluate their human activity detection algorithms on real-world surveillance video datasets. In the end of this chapter, we will discuss some related work on assisting health care by sensors and other computer tools.

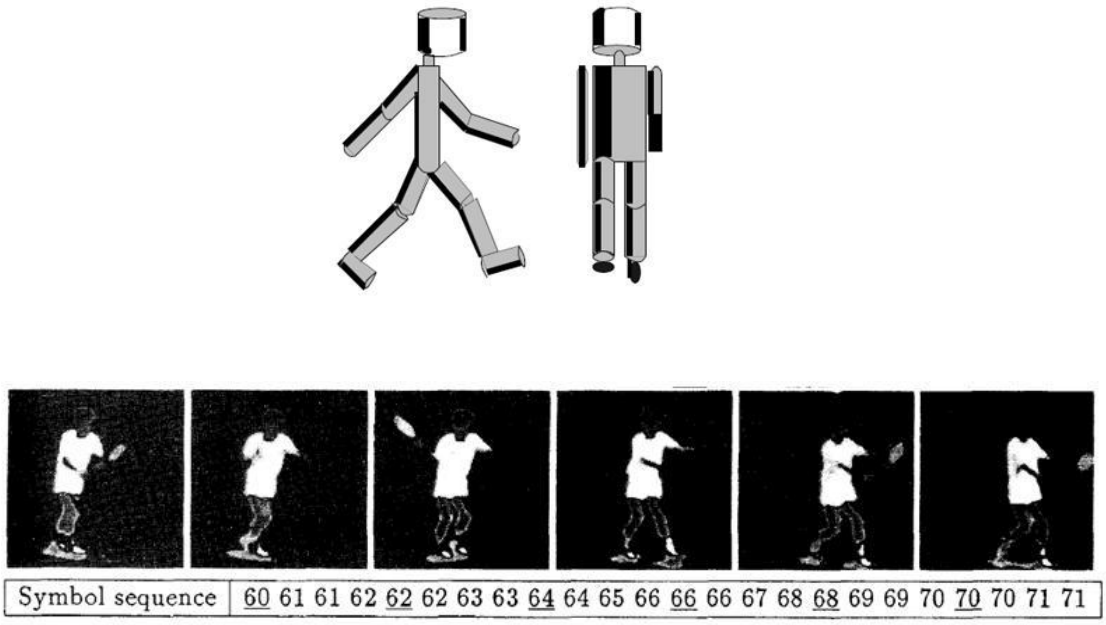

Figure 2.1: Two model based approaches. The top figure shows how to decompose a human body into fourteen elliptical cylinders to simulate walking. The bottom figure demonstrates a tennis image sequence which is modeled by HMM. The figures are adapted from [38, 94].

2.1 Model-based Approaches

Model-based approaches attempt to build motion or action models by estimating model parameters, such as pose and scale. Researchers first try to extract a body outline to analyze human motions. Akita [4] decomposed a human body into six parts: head, torso, arms and legs. A cone model is built which consists of six segments corresponding to their counterparts in stick images. Hogg [38] used elliptical cylinder models to describe human walking. A human body is represented by 14 elliptical cylinders and each cylinder is described by three parameters: the length of the axis, and the major and minor axes of the ellipse cross sec-

tion. This approach attempts to recover the 3D structure of a walking person. Hidden Markov Models (HMMs) have been used to recognize tennis actions. Yamato et al. [94] extracted a symbol sequence from a image sequence and built HMMs to model tennis actions. Bregler [15] further extended HMMs by applying dynamical models which contain spatial and temporal blob information extracted from human bodies. Model-based approaches require not only a good model which can describe the motions and actions but also must track body parts consistent with the constructed models. It has been shown that tracking body parts is a very difficult problem by itself and models are usually built for limited domains and environments. Figure 2.1 gives some examples of model-based approaches.

2.2 Appearance-based Approaches

Appearance-based methods attack the problem by measuring similarity to previously observed data. Template matching is a widely used technique. Polana et al. [71] compute a spatio-temporal motion magnitude template as the basis for recognizing activities. They first detect activities by measuring periodicity and then classify them by comparing the motion magnitude to training examples. Bobick et al. [11] construct Motion-Energy Images (MEI) and Motion History Images (MHI) as temporal templates and then search for the same patterns in test data. Dalal et al. [24] propose grids of Histograms of Oriented Gradients (HoG) descriptors to describe the appearance and significantly improve pedestrian detection. Appearance models can be generally extended to detect various actions without constructing domain specific models. However, they rely fundamentally on segmentation to extract the actors out from the background, which is also a very difficult task. Detecting pose and scale are also essential factors that determine the detection and recognition performance. Deformation in shapes is another challenge to appearance-based approaches. Figure 2.2 shows some examples of MHI and HOG approaches. From the examples, it is clear that appearance-based approaches can be heavily affected by cluttered background, occlusion, and deformation.

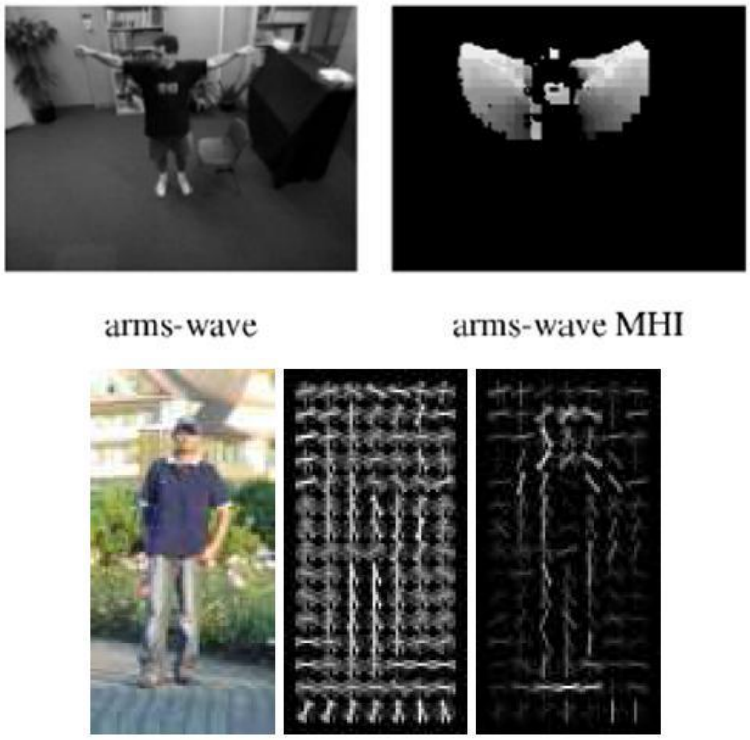

Figure 2.2: Two appearance-based approaches. The top figure shows a template of arms-wave by Motion History Image (MHI). The button figure demonstrates a pedestrian image and corresponding HoG and weighted HoG images. The figures are adapted from [11, 24]

2.3 Part-based Approaches

Part-based approaches have been received attention in recent years. They do not require constructing specific models, unlike the model-based approaches. They also have fewer assumptions than appearance-based methods about capturing the global appearance. These approaches were first inspired by object recognition in static images. They first detect salient points from interested objects and then decompose the object into a combination of these salient points. This has several advantages. Instead of observing the global appearance, a part-based approach tries to search for small discriminative components extracted from the object. This results in an advantage helping to overcome occlusion and posture variations.

Since we only extract informative components, we obtain robustness to deal with variations. The salient points normally contain specific lighting-invariant characteristics and this reduces the effect from illumination change.

In part-based approaches, the essential part is salient point detection, or so called interest point detection. There are a variety of methods to detect interest points from static images in the spatial domain. Typically, a response function is calculated at every location in the image and salient points correspond to local maxima of the response function. One of the most popular approaches to detect interest points is to detect corners, such as the Harris corner detector [31]. The spatial corners are defined as the regions which contain large variations in orthogonal directions, which are the x and y coordinates in still images. The variation is measured by gradient vectors. The gradient vectors are the derivatives of a smoothed image $L(x, y, \sigma) = I(x, y) * g(x, y, \sigma)$, where g is the Gaussian smoothing kernel, σ denotes the smoothing scale and I is the original image. The response function at each point is the rank of the second moment matrix of gradients calculated in a local window which is related to eigenvalues in both directions. A high response strength means large variations in both x and y direction which is a spatial corner. Another popular method to detect interest points is to use a Difference of Gaussians (DoG), such as SIFT [55]. The image is first convolved with Gaussian filters at different scales, and then the differences of successive Gaussian-blurred images are taken. Salient points are taken as maxima/minima of the difference of Gaussians that occur at multiple scales. Specifically, a DoG image is given by $D(x, y, \sigma) = L(x, y, k_i\sigma) - L(x, y, k_j\sigma)$ where $L(x, y, k\sigma) = I(x, y) * G(x, y, k\sigma)$ is the original image convolved with a Gaussian blur function at scale $k\sigma$ which k indicates scale. Once DoG images have been obtained, salient points are identified as local minima/maxima of the DoG images across scales.

In videos, we need to extract points not only with informative spatial locations but also interesting temporal information. We call these points spatio-temporal interest points. Spatio-temporal interest points are used to decompose complicated motions and actions into small and independent components. Laptev et al. [49] extended the Harris interest point detector to detect spatio-temporal corners in video sequences. Instead of a 2-D Gaussian smoothing kernel in a still image, a 3-D Gaussian smoothing kernel is applied to the video. A video can

Figure 2.3: Some spatio-temporal interest point examples from a walking sequence. The figures are adapted from [49].

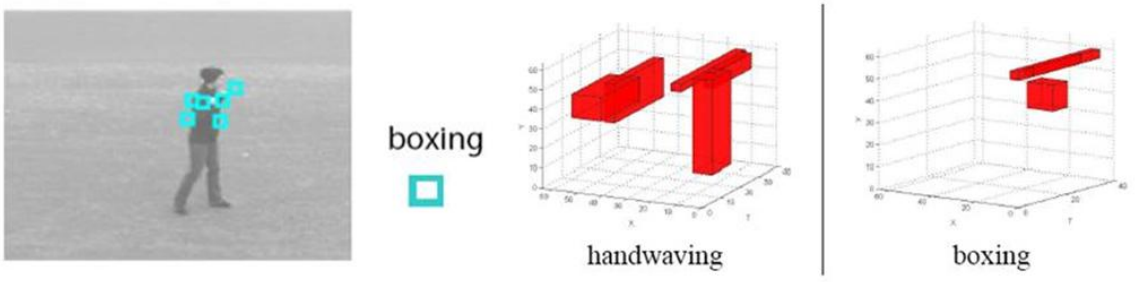

Figure 2.4: Some examples from Dollar's interest point detection and volumetric features. The most left figure shows interest point detection from a boxing action. The other two figures illustrate hand waving and boxing volumetric features. The figures are adapted from [27, 45]

be seen as a cuboid of successive images. Therefore, a smoothed video clip is $L(x, y, t, \delta, \tau) = I(x, y, t) * g(x, y, t, \delta, \tau)$, where g is the Gaussian smoothing kernel, δ controls spatial scale, τ controls temporal scale and I is the original video. Similar to the Harris detector, Laptev constructs a second-moment matrix which is a 3-by-3 matrix composed of first order spatial and temporal derivatives. The detector searches for points which have both high eigenvalues in all three dimensions from the second-moment matrix. Therefore, an interest point is a region which has large variations in both spatial and temporal directions. To be more specific, a spatio-temporal corner is a spatial interest corner corresponding to the moments with non-constant motion. Figure 2.3 gives some examples of spatio-temporal interest points detected in a walking image sequence.

Dollar et al. [27] attempted to detect periodic frequency components. The response function has the form $R = (i * g * h_e v)^2 + (I * g * h_o d)^2$ where $g(x, y, \delta)$ is the 2D Gaussian smoothing kernel, applied only on the spatial dimensions,

and h_ev and h_od are a quadrature pair of 1D Gabor filters which are applied in the temporal direction. They are defined as $h_ev(t,\tau,w) = -\cos(2\pi tw)e^{-t^2/\tau^2}$ and $h_od(t,\tau,w) = -\sin(2\pi tw)e^{-t^2/\tau^2}$. Dollar set up $w = 4/\tau$ to constrain the response function R with two parameters δ and τ which correspond to the spatial and temporal scale of the detector. The response function is applied to the video cuboid and local maxima are extracted as interest points. Periodic motions represent one important type of motions but can not represent every complicated activity. However, this approach has shown very impressive recognition results and it is widely used.

Both of the above approaches attempt to decompose human behaviors into small, characteristic and location independent components with shape and motion information. Ke et al. [45, 46] proposed volumetric features to describe events. The features are extracted from optical flow and are represented as a combination of small volumes. This method combines the part-based method with a motion model. It still decomposes the complicated motions into small units. However, the combination of the volumes can capture the outline of the whole action. It does not achieve as robust recognition results as the interest point method, but it provides another informative feature for analyzing actions.

2.4 Video Content Mining

In addition to robust recognition techniques, researchers are also interested in applying inference mechanisms to analyze recognition results to understand video content. The recognition results explore what is in the video; however, integrating spatial and temporal relationships with recognition results provides a clear understanding of the whole video content. This includes interaction between people, interaction between people and the environment and description of the environment. Event detection usually has very complicated circumstances with a combination of people, objects, time, and environment. Therefore, researchers try to build up graph models to monitor event processing and to incorporate the observations into recognition results.

David et al. [25] proposed a system which is able to answer a user's queries about human activities. The system returns video clips that satisfy the users'

queries, removing any other clips that are not relevant to the query. A query usually describes a scenario, and a scenario is built up using a set of spatial relations, temporal relations and logical operators. An inference mechanism is applied to object or motion recognition results to infer the presence of the predefined scenarios. A bipartite network represents each query graphically. Each node represents a video feature detected by a vision algorithm, such as object and behavior recognitions, and the network maps low-level raw features to higher-level semantics, such as ”a person opens a car door”.

Boger et al. [12] proposed a Markov decision process framework to assist people with Dementia. A Markov decision process framework is a plan graph which contains four different state variables: environment variables, activity status variables, system behavior variables and user variables. This graph connects human actions with system behavior and its environment. This plan graph decomposes a complicated action into several steps described by state variables which contain information not only from the patient but also from the environment and the assisting system. Using sensors and detectors, the system can collect information from all three aspects: user, environment and system, and the system can also monitor which step the user is attempting in order to give appropriate assistance.

2.5 Semantic Feature Extraction

The semantic gap is a fundamental challenge in content based video retrieval [32, 35]. Semantic concept detections can be a promising approach to bridge the semantic gap by adding understandable meta-data provided by semantic detectors [34]. Generic approaches for large-scale concept detection have received a lot of attention recently. However, most research efforts still focus on keyframe classification, and motion-related concept detection is an understudied research topic. Cees et al. [81] proposed extracting multiple frames in the same video segment to capture motion related to semantic concepts. Inoue et al. [40] proposed aggregating image features from every frame inside a video segment to capture motions inside the sequence. Those state-of-the art motion-related concept detectors actually do not analyze motions at any level of detail. Therefore, robust activity analysis could be helpful to extract semantic concepts which are related

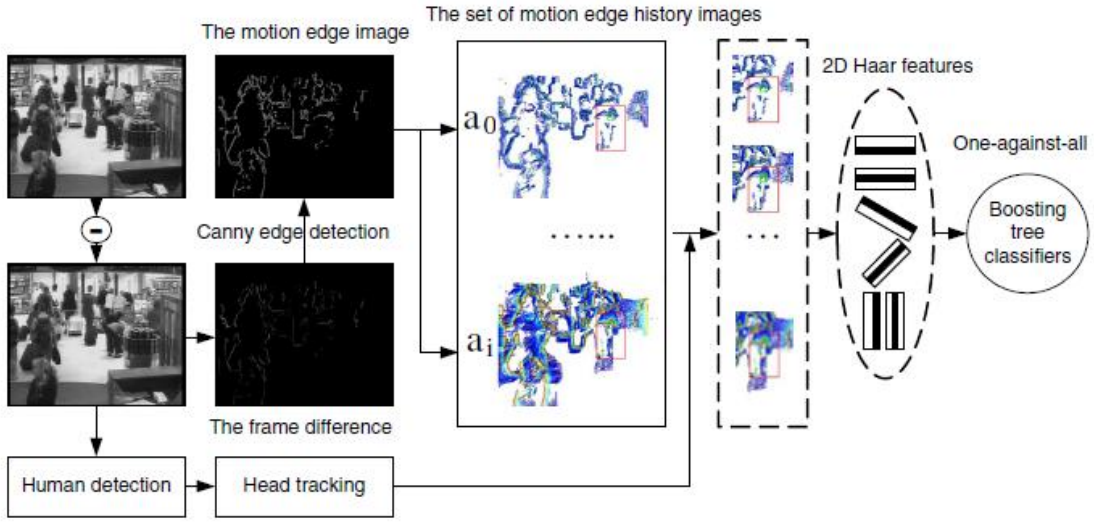

Figure 2.5: An example of using human detection and tracking to help detecting activities in a surveillance video. The motion edge image and edge detection are extracted from video. Human detection and tracking results help the system to focus on person related regions. A cascaded classifier is applied to identify interesting activities. The figure is adapted from [96]

to motion.

2.6 Activity detection in a surveillance video

Although many activity analysis techniques have been demonstrated to perform robustly in selected datasets, a real-world surveillance video archive is still extremely challenging, due to complicated environments, cluttered backgrounds, occlusions, illumination changes, multiple activities, and great deformations of an activity. NIST provides researchers a platform to study and evaluate activity detection algorithms by annotating 100 hours of airport surveillance video. Zhu et al. [100] proposed detecting activities by describing appearance and motions from person tracking results. A person tracking result first filters the background, then spatio-temporal cubes are extracted from the tracked person. A spatio-temporal cube is described by gradients and optical flows and a SVM classifier is applied to identify an interesting activity. The proposed method is strongly affected by

human detection results and occlusions. This algorithm is not able to analyze person to person and person to object activities as well. Yang et al. [96] proposed an activity representation scheme using a set of motion edge history images and human trackers. The false positive rate is reduced significantly by a cascaded Adaboost classifier. The algorithm again relies on human tracking and is only able to handle single person activities. Human detection and tracking are widely applied in activity detection task in surveillance video [97, 98]. This is a efficient way to reduce the search space because human detectors and trackers filter non-person related regions directly. However, current human detection and tracking algorithms still have high error rates. Accumulating errors from human detectors and trackers should be avoided to build a robust activity detector in surveillance video domain. Figure 2.5 illustrates an approach to use human detection and tracking results to detect interesting activities, which is adapted from Yang et al. [96].

2.7 Health care analysis

More and more researchers are starting to utilize sensors and other tools to monitor and analyze human behaviors to assist health care. Adami et al. [2] proposed a system for unobtrusive detection of movement in bed that uses load cells installed at the corners of a bed. The movement detection during sleeping provides doctors a useful diagnostic feature to estimate quality of sleep. Michael et al. [58] proposed to use Global Position System (GPS) enabled cell phones to track people to understand their social interactions. It is believed that an elderly person with more social interactions tends to be more healthy. Unay [88] proposed fusing clinical and patient-demographics related observations with visual features computed from brain longitudinal MRI (magnetic resonance imaging) data for improved dementia diagnosis. This work demonstrates that processed sensor data (MRI can be treated as a sensor) can slightly improve the diagnosis. All these related works attempt to use sensors to collect desired data to improve health care. However, the information from a sensor is limited and can not really reflect the details of a person's daily life. Surveillance recording, in the other hand, requires more difficult post processing but provides comprehensive views of a person's daily life. In conclusion, the surveillance method is a complementary method to the sensor

approach but reveals detailed observations.

Chapter 3

Motion SIFT

This chapter presents our Motion SIFT (MoSIFT) algorithm to detect and represent interest points in videos. Interest point detection [55] reduces the video from a volume of pixels to a sparse but descriptive set of features. Ideally, interest points should densely sample those portions of the video where activities occur while avoiding regions of low movement. Therefore, our goal is to develop a method that generates a sufficient but manageable number of interest points that can capture the information necessary to recognize arbitrary human activities. In contrast to previous work that either focuses entirely on appearance or spatio-temporal extrema, MoSIFT identifies spatially-distinctive regions that exhibit sufficient motion at a variety of spatial scales (see Figure 3.1). The information in the neighborhood of each interest point is expressed using a descriptor that explicitly encodes both an appearance and a motion component. The former aspect is captured using the popular SIFT descriptor [55] and the latter using a SIFT-like encoding on local optical flow. Details of our algorithm are described in the following sections.

3.1 MoSIFT interest point detection

Popular spatio-temporal interest point detectors [27, 49] generalize established 2D interest point detectors (such as the Harris corner detector [31]) to 3D. While this is arguably elegant from a mathematical perspective, such detectors are restricted to encoding motions in an implicit manner, thus providing limited sensitivity for

Figure 3.1: Interest points detected with SIFT (left) and MoSIFT (right). Green circles denote interest points at different scales while magenta arrows illustrate optical flows. Note that MoSIFT identifies distinctive regions that exhibit significant motion, which corresponds well to human activity while SIFT fires strongly on the cluttered background.

smooth gestures, such as circular motions which lack sharp space-time extrema. The philosophy behind the MoSIFT detector is to treat appearance and motion separately, and to explicitly identify those spatially-distinctive regions in a frame that exhibit sufficient motion.

Like other SIFT-style keypoint detectors, MoSIFT finds interest points at multiple spatial scales. MoSIFT's fundamental operations are performed on a pair of consecutive video frames. Two major computations are employed: SIFT interest point detection on the first frame to identify candidate features; and optical flow computation between the two frames, at a scale appropriate to the candidate feature, to eliminate those candidates that are not in motion. The MoSIFT detector scans through every frame of the video (overlapping pairs) to identify keypoints in each frame.

The candidate interest points are determined using SIFT [55] on the first frame of the pair. For completeness, we now briefly review this interest point detector.

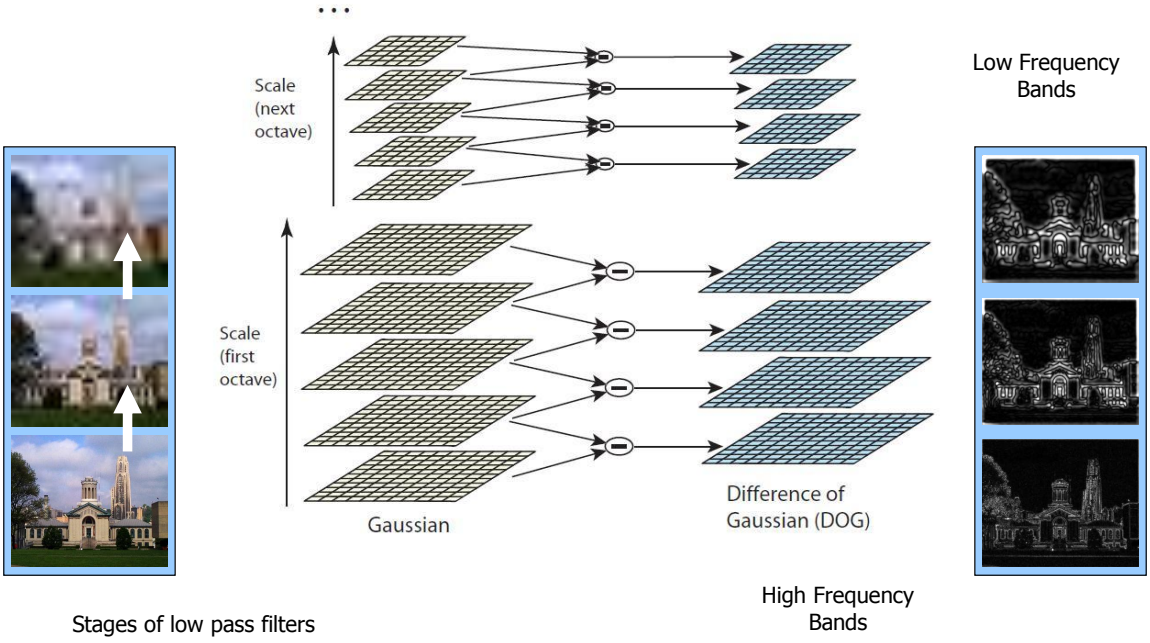

Figure 3.2: For each octave, the initial image is repeatedly convolved with Gaussians to produce images with different scales on the left. After each octave, the image is down-sampled by a factor of 2. Adjacent Gaussian images are subtracted to produce the difference-of-Gaussian (DoG) images on the right. The DoG is approximate to a band-pass filter that discards all but a handful of spatial frequencies that are present in the original grayscale image. Figures are adapted and revised from [55].

3.1.1 Scale-invariant feature transform

SIFT interest points are scale invariant and all scales of a frame image must be considered. A Gaussian function is employed as a scale-space kernel to produce a scale space transform of the first frame. The whole scale space is divided into a sequence of octaves and each octave is further subdivided into a sequence of intervals, where each interval is a scaled frame. The number of octaves and intervals is determined by the frame size. The first interval in the first octave is the original

frame. In each octave, the first interval is denoted as $I(x, y)$. We can denote each interval as

$$L(x, y, k\sigma) = G(x, y, k\sigma) * I(x, y) \tag{3.1}$$

where $*$ denotes the convolution operation in x and y, and $G(x, y, k\sigma)$ is a Gaussian smoothing function:

$$G(x, y, k\sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \tag{3.2}$$

In the next octave, the first image is down-sampled by factor of 2 from the current octave. Difference of Gaussian (DoG) images, which approximate the output of a band-pass Laplacian of Gaussian operator, are then computed by subtracting adjacent intervals

$$D(x, y, k\sigma) = L(x, y, k\sigma) - L(x, y, (k-1)\sigma) \tag{3.3}$$

A band-pass filter discards all but a handful spatial frequencies that are present in the original grayscale image. Figure 3.2 illustrates the idea of Gaussian and DoG pyramids. Once the pyramid of DoG images has been generated, the local extrema (minima/maxima) of the DoG images across adjacent scales are used as the candidate interest points. In the implementation, a local extremum is determined within 3x3 regions at the current and adjacent scales (see Figure 3.3). The algorithm scans through each octave and interval in the DoG pyramid and extracts all of the possible interest points at each scale.

3.1.2 Motion SIFT

The original SIFT algorithm was designed to detect distinctive interest points in still images, and therefore considers only appearance information. Thus, the candidates include a large number of interest points on a cluttered but stationary background that are not useful for describing human activities. Therefore, MoSIFT only seeks to retain those interest points that are in motion. This is done by calculating the optical flow [56] between the pair of frames. Optical flow pyramids are constructed over two Gaussian pyramids from consecutive frames. Opti-

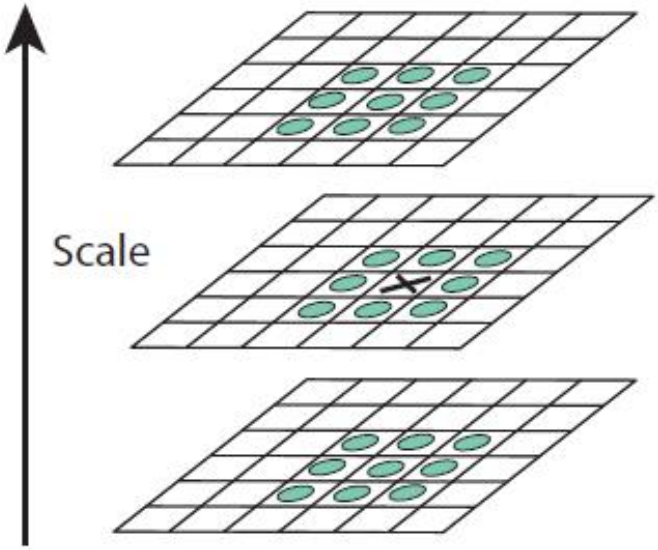

Figure 3.3: A local extrema of the DoG images is detected in 3x3 regions at the current and adjacent scales. Figure adapted from [55].

cal flow is computed at each of the multiple scales used in SIFT. Candidate points (local extrema from DoG pyramids) are selected as MoSIFT interest points only if they contain sufficient motion in the optical flow pyramid at the appropriate scale. Thus, MoSIFT identifies interest points on distinctive regions that are in motion. Compared to video cuboids or spatio-temporal volumes, the optical flow representation explicitly captures the magnitude and direction of a motion, rather than implicitly modeling motion through appearance change over time. Our hypothesis (supported by our experiments in Section 3.4.1) is that MoSIFT's explicit representation of motion, described below, plays a critical role in its ability to accurately recognize activities. Figure 3.1 contrasts the interest points detected by the original SIFT algorithm with those identified by MoSIFT; note that we focus primarily on regions of the image with significant human activity.

MoSIFT interest points are scale invariant in the spatial domain. However, they are not scale invariant in the temporal domain. Temporal invariance is a complicated and ill-defined problem. If the temporal invariant is defined by the completeness of a simple and straightforward motion such as eyelids moving up.

MoSIFT can achieve this temporal invariant by calculating optical flow on multiple scales in time. However, a complete motion such as blinking contains at least two different simple motions, eyelids moving up and down. The temporal invariant is then hard to define with this assumption. Normally, a human activity is composed of a lot of simple motions. Therefore, we decide to implement temporal invariance at the activity level instead of at the interest point level by segmenting videos into different temporal intervals. We will discuss more about activity level temporal invariance in Chapter 5.

3.2 MoSIFT feature description

Since MoSIFT interest points combine distinctive appearance with sufficient motion, it is natural that the MoSIFT descriptor should explicitly encode both appearance and motion. We are not the first to propose representations that do this; several researchers [50, 76] have reported the benefits of augmenting spatio-temporal representations with histograms of optical flow (HoF). However, unlike those approaches, where the appearance and motion information is separately aggregated, MoSIFT constructs a single feature descriptor that concatenates appearance and motion, as described below.

The appearance component is the 128-dimensional SIFT descriptor for the given patch, briefly summarized as follows. The magnitude and direction for the intensity gradient are calculated for every pixel in a region around the interest point in the Gaussian-blurred image. An orientation histogram with 8 bins is formed, with each bin covering 45 degrees. Each sample in the neighboring window is added to a histogram bin and weighted by its gradient magnitude and its distance from the interest point. Pixels in the neighboring region are normalized into 256 (16×16) elements. Elements are grouped as 16 (4×4) grids around the interest point. Each grid contains its own orientation histogram to describe sub-region orientation. This leads to a SIFT feature vector with 128 dimensions ($4\times4\times8 = 128$). Each vector is normalized to enhance its invariance to changes in illumination. Figure 3.4 illustrates the SIFT descriptor grid aggregation.

MoSIFT adapts the idea of grid aggregation in SIFT to optical flow. The optical flow describing local motion at each pixel is a 2D vector with the same structure

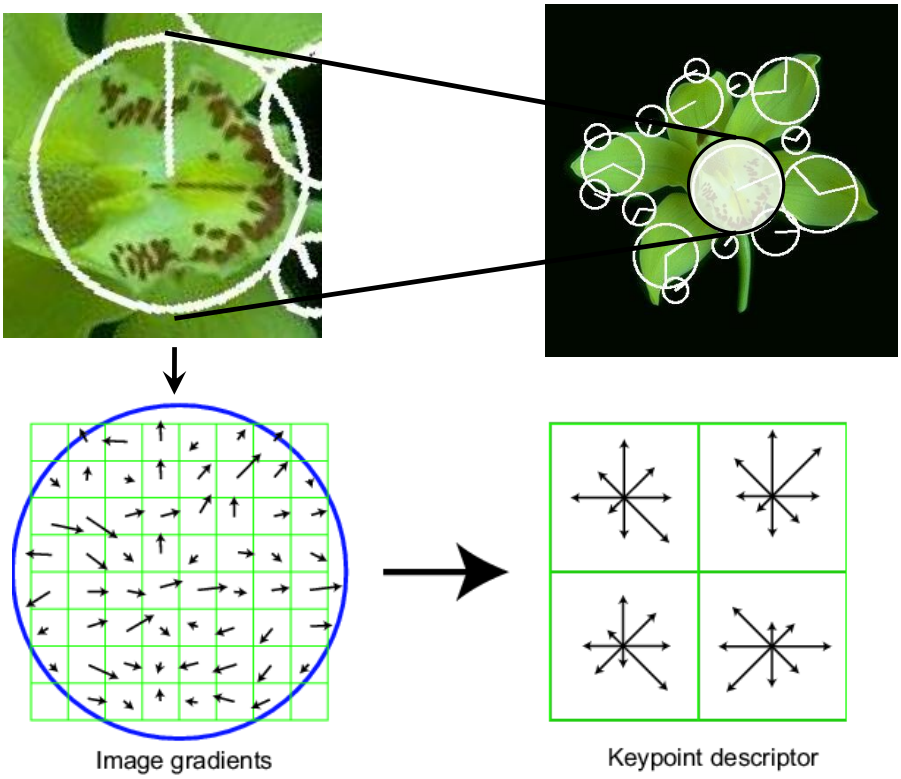

Figure 3.4: MoSIFT aggregates appearance and motion information using a SIFT-like scheme. The region of interest is normalized into 256 elements. Elements are grouped as 16 grids and each grid is described by an 8 dimensional vector. This makes MoSIFT a 256 dimensional descriptor where 128 dimensions describe appearance and the other 128 dimensions represent motion. Figure adapted from [55].

as the gradient describing local appearance. This enables us to encode motion with the same scheme as that used by SIFT for appearance. A key benefit of this aggregation approach is that our descriptor becomes tolerant to small deformations and partial occlusion (just as standard SIFT was designed to be tolerant to these effects). The two aggregated 128-dimensional histograms (appearance and optical flow) are concatenated to form the MoSIFT descriptor, which is a vector of 256 dimensions. Since directions of appearance and motion indicate the shape of an activity, we don't do rotation on either appearance or motion. Rotation in-

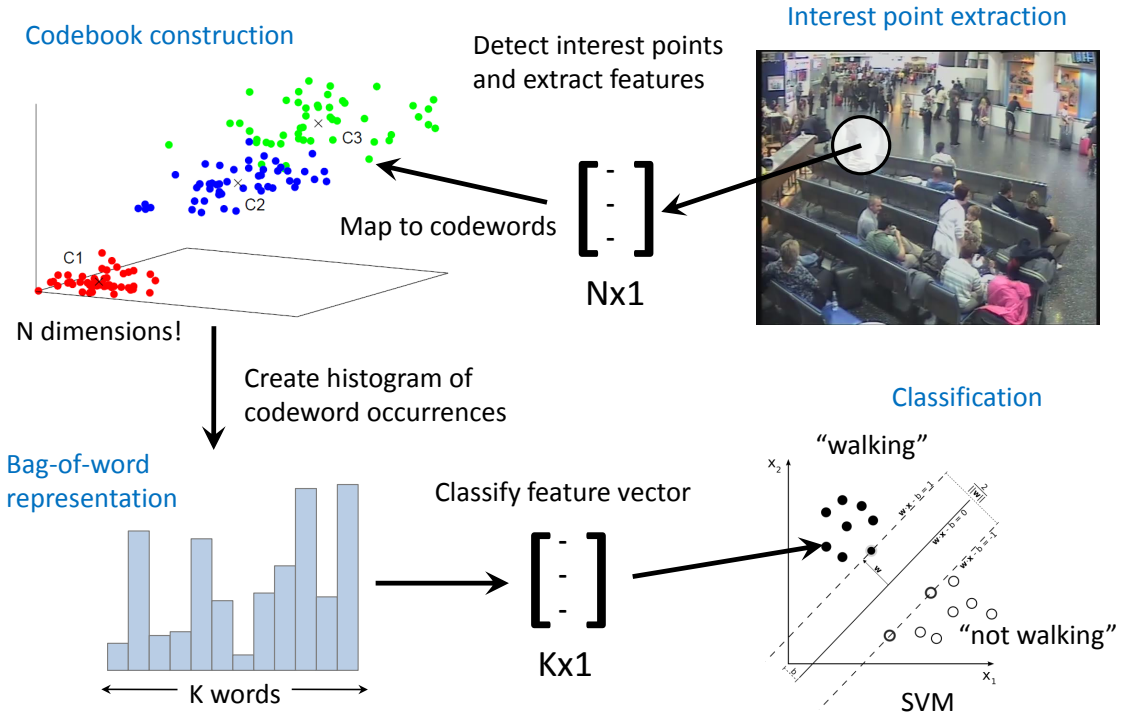

Figure 3.5: The four major steps of MoSIFT activity recognition: interest point extraction, codebook constrcution, bag-of-word representation, and classification. In this figure, each interest point is represented by a N dimensional vector ($N = 256$ in MoSIFT), and each video segment is denoted as a K (decided by cross-validation) dimensional bag-of-word feature.

variance is achieved in SIFT but we are not convinced that is helpful for analyzing activities. For example, raising one's hand has a different meaning than pushing one's hand forward. We want to be able to distinguish these two activities by the direction of motion.

3.3 MoSIFT activity recognition

In MoSIFT activity recognition (illustrated in Figure 3.5), there are four major steps: interest point extraction, video codebook construction/mapping, bag-of-

word feature representation, and modeling. Here, we discuss more details of how we implement this in our experimental setting.

3.3.1 Interest point extraction

In MoSIFT feature extraction, sufficient motion is determined by the size of the frame. In our implementation, we extract interest points which contain either vertical or horizontal movements which are larger than 0.5% of frame height or width. In different scales or octaves, the frame size changes and the sufficient motion is then determined by the current scale.

3.3.2 Video codebook construction/mapping

The video codebook is constructed by the standard K-means clustering algorithm. Two major issues arise here: sampling and number of codewords. The first problem is sampling. Normally, a couple hundred interest points would be extracted from each frame pair. This equals at least one hundred thousand interest points extracted per hour. It is not practical to run a clustering algorithm on all interest points from training data due to memory limitations. Sampling is required to reduce the number of interest points for the clustering process and sampling the right distribution is an important step to get a better video codebook. In our experiments, we applied standard random sampling. However, our experimental results also demonstrated that the capability to train clustering on all extracted interest points can significantly improve the recognition result. The second issue is the size of the video codebook (k in K-means clustering). From our experimental results, it is clear that the size of the codebook is a strong factor in recognition performance. Unfortunately there is no clear objective function to optimize the size of the codebook. In our experimental setting, we use cross-validation to determine the size of video codebook.

3.3.3 Bag-of-word representation and classification

We adopt the popular bag-of-features representation and discriminant classification for action recognition, summarized as follows. Each video clip is represented

by a histogram of occurrence of each codeword (bag of features). This histogram is treated as a K-element input vector for a support vector machine (SVM) [13], with a χ^2 kernel. The χ^2 kernel is defined as:

$$K(x_i, x_j) = exp(-\frac{1}{A}D(x_i, x_j)), \tag{3.4}$$

where A is a scaling parameter that is determined empirically though cross-validation. $D(x_i, x_j)$ is the χ^2 distance defined as:

$$D(x_i, x_j) = \frac{1}{2}\sum_{k=1}^{m}\frac{(u_k - w_k)^2}{u_k + w_k}, \tag{3.5}$$

with $x_i = (u_1, ..., u_m)$ and $x_j = (w_1, ..., w_m)$. Prior work has shown that this kernel is well suited for bag-of-words representations [99]. SVM is a binary classifier. we adopt the standard one-vs-rest strategy to train multiple SVMs for multi-class learning.

3.4 MoSIFT evaluation: activity recognition

In this section, we evaluate our MoSIFT algorithms on four different datasets: KTH, Hollywood, Gatwick, and CareMedia. The KTH and Hollywood datasets are standard datasets and are widely used in academia to evaluate activity recognition algorithms. The Hollywood dataset is from edited movie scenes and has many camera motions. The Gatwick and CareMedia datasets are real-world surveillance datasets in two different domains. Their cluttered backgrounds and multiple activities provide exciting challenges to automatic activity recognition algorithms.

3.4.1 The KTH dataset

The KTH human motion dataset [78] has become a standard benchmark for evaluating human activitiy recognition algorithms. Although KTH is much smaller than the datasets that form the focus of our research, it serves as a consistent point of comparison against current state-of-the-art techniques. Figure 3.6 illustrates

Figure 3.6: Some examples of MoSIFT from the KTH dataset. In the left two columns, from top to bottom are boxing hand waving and walking. In right two columns, from top to bottom are hand clapping, jogging and running. Green circle indicates interest points and purple arrows show the direction of motion. As seen in these sequences, jogging and running are very similar.

some examples of MoSIFT interest points detected in different activities in KTH dataset. As seen in the examples, jogging and running are very similar and hard to distinguish.

We follow [27, 45, 64, 93] in performing leave-one-out cross-validation to evaluate our approach. Leave-one-out cross-validation uses 24 subjects to train activity models and then tests on the remaining subject. Performance is reported as the average accuracy over 25 runs.

As we discussed earlier, the size of the video codebook is a significant factor in recognition performance. Therefore, cross-validation is used to determine the size of codebook. A small codebook size will cause coarse clustering in which small changes can't be distinguished. A large codebook size will increase the dimension of the bag-of-word feature resulting in worse performance due to "curse of dimensionality" in the classification process. Figure 3.7 shows the relationship be-

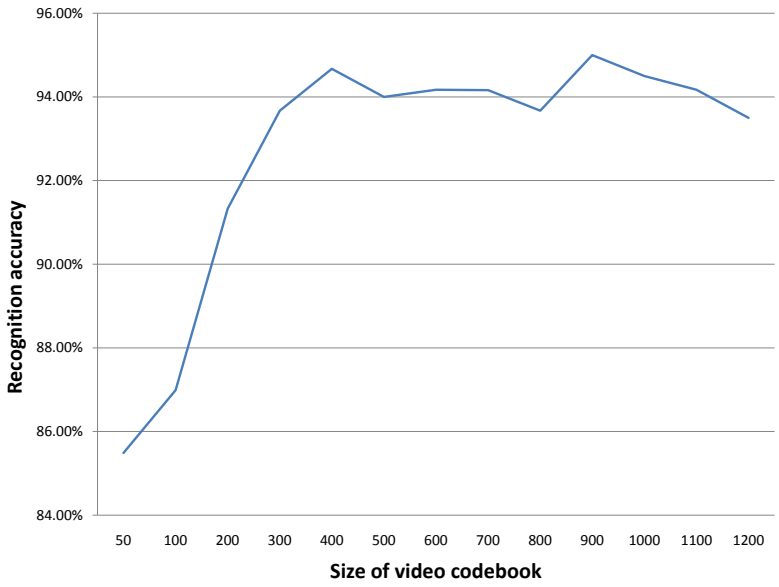

Figure 3.7: Codebook size is an important factor in recognition performance. A small codebook size leads to coarse clustering and loses detail of the activities. A large codebook size captures motion details of the activities but it results in high dimensionality in classifier vectors which may weaken the performance. In the KTH dataset, 900 video codewords result in the best performance. The size of codebook is determined by cross-validation.

tween codebook size and recognition performance. In the KTH, a video codebook of size 900 gives the best performance according to cross-validation. The confusion matrix for 900 video codewords is given in Figure 3.8. The major confusions occur between jogging and running.

Table 3.1 summarizes our results on the KTH dataset. We observe that MoSIFT demonstrates a significant improvement over current methods, many of which employ bag-of-features with different descriptors. In particular, Laptev et al. [50] employed a bag-of-features approach on feature descriptors which describe appearance (histogram of gradient, HoG) and motion (histogram of optic flow, HoF) with aggregating neighborhoods which gives the second best published results. By applying the t-test, the improvement is statistically significant given a 95% confidence interval. Wong et al [93] and Niebles et al. [64] both use HoG to describe spatio-temporal cuboids around interest points which only implicitly describe motions. This leads to less efficiency to fully describe activities. The final

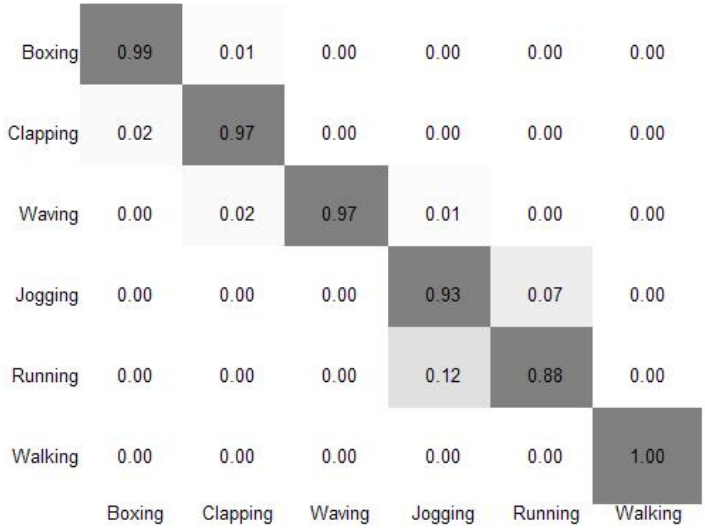

Figure 3.8: Confusion matrix for the KTH activities. This is achieved by 900 video codewords. The major confusions occur between jogging and running.

Method	Accuracy
MoSIFT	**95.83**%
Laptev et al. [50]	91.8%
Wong et al. [93]	86.7%
Niebles et al. [64]	83.3%
Dollar et al. [27]	81.5%
Schuldt et al. [78]	71.7%
Ke et al. [45]	62.7%

Table 3.1: MoSIFT significantly outperforms current methods on the standard KTH dataset.

comparison (Ke et al. [45]) is against a boosted cascade that operates solely on optical flow without modeling appearance. Clearly, an explicit representation of motion alone is insufficient for human activity recognition. These results are a strong validation for our decision to combine appearance and motion into a single descriptor.

Figure 3.9: Some examples of MoSIFT from the Hollywood dataset. Top left is a handshaking activity. Top right is a man getting out from a car. Bottom left is a kissing activity and a standing up activity in bottom right. A green circle indicates interest points and the purple arrows show the direction of motion.

3.4.2 The Hollywood movie dataset

The Hollywood dataset is another standard dataset used to evaluate activity recognition algorithms. The Hollywood dataset collects human activity clips from real-world movies, which is the major difference from the laboratory collection of the KTH dataset. Since the dataset is selected from movie scenes, it contains more dynamic backgrounds and the activities in the dataset have more variety than the KTH dataset. This dataset also includes a large number of camera motions in the video clips. Camera motion will produce MoSIFT interest points that are not related to interesting activities. However, in most cases, the activity we want to recognize is the main focus of the shot which leads to fewer problems distinguishing multiple activities in this dataset.

In the Hollywood dataset, we apply a video codebook of size 1000 to construct our bag-of-word features via cross-validation. We train our models with clean training examples which contain 219 video samples with manually verified la-

Activity	Random	Laptev [50]	MoSIFT
AnswerPhone	10.6%	13.4%	17.5%
GetOutCar	6.0%	21.9%	45.3%
HandShake	8.8%	18.6%	18.9%
HugPerson	10.1%	29.1%	39.7%
Kiss	23.5%	52.0%	49.5%
SitDown	13.8%	29.1%	34.7%
SitUp	4.6%	6.5%	7.5%
StandUp	22.6%	45.4%	44.3%
Average	12.5%	27.0%	**32.2**%

Table 3.2: MoSIFT significantly improves recognition performance on the Hollywood movie dataset. The performance is measured by average precision.

bels. The test set has 211 samples. The result is shown on Table 3.2. Following the same experimental setting as [50], we measure the performance by average precision (AP). Comparing this with Laptev's spatio-temporal interest point approach, MoSIFT outperforms significantly by t-test given 95% confidence. MoSIFT demonstrates robustness on the Hollywood dataset and proves its consistent activity recognition performance in different domains (both the KTH and Hollywood datasets).

3.4.3 The Gatwick dataset

The 2008/2009 TRECVID surveillance event detection dataset [85, 86] was collected by 5 cameras at London Gatwick International Airport. We evaluate recognition performance in a forced-choice setting (i.e., "which of the 10 events is this?") using the annotations provided by NIST. There were a total of 6,439 events in the development set. The size of the video codebook was fixed at 2000 after cross validation on the development set. Since the data were captured by 5 cameras over 5 different days, we evaluated each camera independently using 5-fold cross-validation and averaged their results. There were not enough annotated examples for **OpposingFlow**, **ElevatorNoEntry** and **TakePicture** to run cross validation; therefore, we do not report performance results of these three tasks. We use average precision as the metric, which is typical for TRECVID high-level feature recognition.

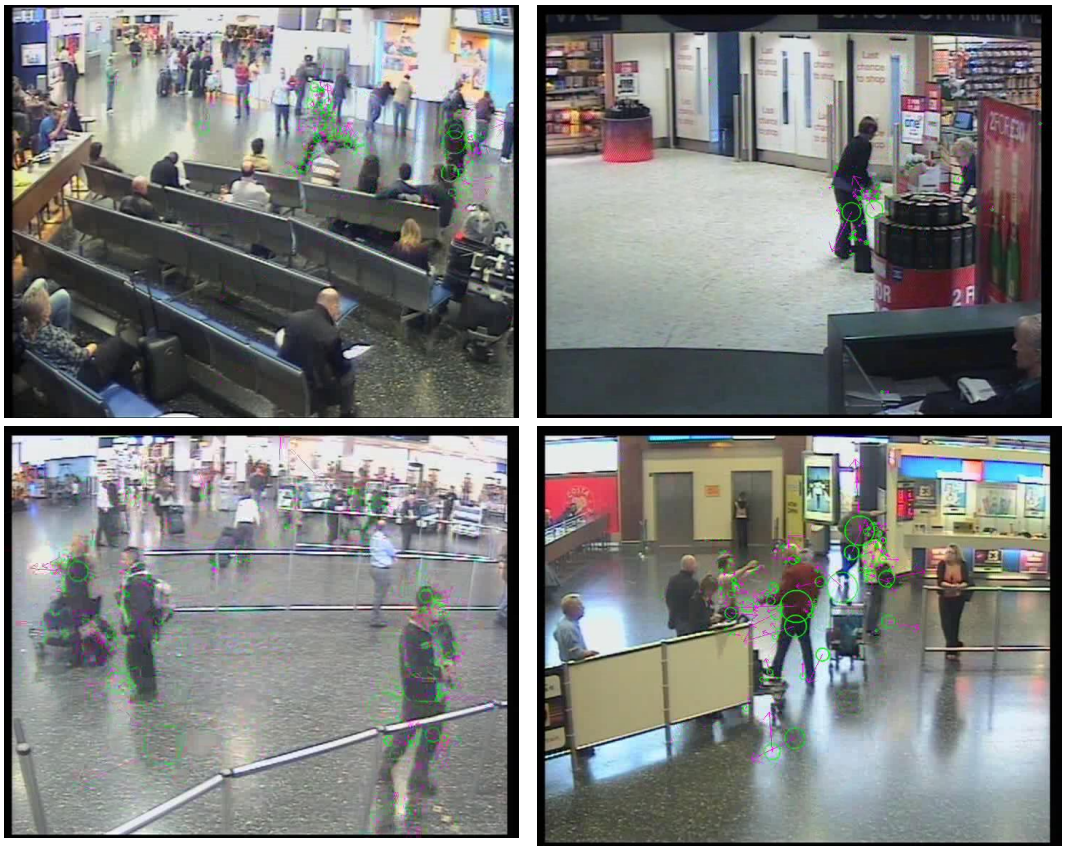

Figure 3.10: Some examples of MoSIFT from the Gatwick dataset. Top left is a person running through the scene. Top right is an ”object put” activity. Bottom left is a ”pointing” activity (the lady left in the scene) in a busy environment. Bottom right is an embracing activity. A green circle indicates interest points and purple arrows show the direction of motion.

In the Table 3.3, we compare MoSIFT again with Laptev et al. [50] which has the second best performance in the KTH dataset. In the comparison, MoSIFT outperforms Laptev's method on five of seven activities (CellToEar, ObjectPut, PeopleSplitUp, Pointing, and PersonRuns) and on the average of all seven activities. By applying the t-test, the improvement is considered to be statistically significant. The improved performance of Latptev's method mainly comes from aggregated descriptors and the ability to detect slow or smooth motions in videos. Compared with a random classifier result, MoSIFT appears to be a robust algorithm for real-world surveillance video archives.

Activity	Random	Laptev [50]	MoSIFT
CellToEar	6.98%	19.42%	22.61%
Embrace	8.03%	29.35%	29.97%
ObjectPut	18.03%	44.24%	47.22%
PeopleMeet	22.32%	44.69%	41.68%
PeopleSplitUp	13.63%	56.91%	57.88%
Pointing	26.11%	41.54%	44.61%
PersonRuns	4.95%	32.56%	36.12%
Average	14.29%	38.39%	**40.01**%

Table 3.3: MoSIFT significantly improves recognition performance on the 100-hour Gatwick surveillance dataset. The performance is measured by average precision.

3.4.4 The CareMedia dataset

The CareMedia dataset is a collection of surveillance video data from a geriatric nursing home. The surveillance system was designed to collect information about patients' daily activities and to provide useful statistics to help doctors' diagnosis. With the help of doctors whose patients were in this elder nursing house, we defined 19 different human actions that doctors are interested in. They can be categorized into two types. The first type (pass 1) is concerned with patients' movement activities and the second type (pass 2) is about patients' detailed behaviors (See Appendix B). The movement activity category contains 12 activities. The detail behavior category has 7 superordinate behavior codes and each superordinate code contains couple more subordinate codes. Figure 9 shows some examples from the four activities.

We choose camera 133 in the dining room as our evaluation set. This camera captures patients' activities during lunch and dinner time. In total, we labeled 2528 activities from the movement category and 4376 activities from the patients' detailed behavior category. We did a cross-validation on the data and discovered that 1000 video codewords represent the best vocabulary size. Five-folder cross validation was applied in our evaluation. In this evaluation, we want to understand how accurate the proposed algorithm might be. Therefore, we chose to use Average Precision (AP) which is commonly used in retrieval tasks. AP not only reflects correct predictions but it also considers the ranking provided by the

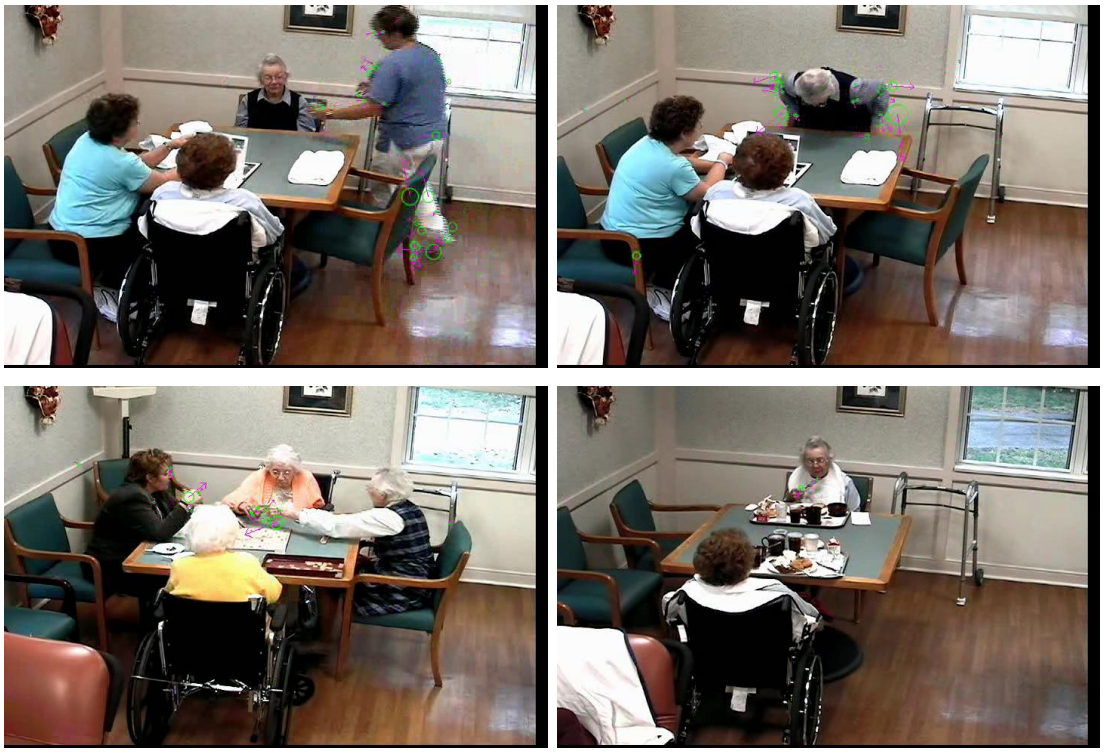

Figure 3.11: Some examples of MoSIFT from the CareMedia dataset. Top left is a ″Object paced on table″ activity. Top right is a ″standing up″ activity. Bottom left is an activity where one patient is pulling the other patient's fingers. Bottom right is a eating activity. Green circle indicates interest points and purple arrows show the direction of motion.

classifiers. We first show the performance of movement activities in Table 3.4. MoSIFT results a strong performance on the movement activity category which has clear definitions and distinguishing motion patterns. In the CareMedia collection, MoSIFT outperforms the Laptev's method by a large margin because MoSIFT captures smooth activities better than the Laptev's method. In a nursing home, residents move slowly and this criterion gives MoSIFT a substantial performance improvement. By applying the t-test, both movement activity and detailed behavior categories significantly outperform Laptev's method by a 95% confidence interval. Among movement activities, "Communicates with staff" has a poor performance comparing to other activities. "Communicates with staff" contains a lot of verbal activities which can't be recognized from video.

Table 3.5 shows the performance in the detailed behavior category. The detailed behavior category is more complicated than the movement activity category. Each behavior in this category contains a set of activities. For an example, there are 19 sub-category activities defined and annotated in "Physical aggressive behaviors": *Splitting*, *Grabbing*, *Banging*, *Pinching or squeezing*, *Punching*, *Elbowing*, *Slapping*, *Tackling*, *Using object as weapon*, *Taking from others*, *Kicking*, *Scratching*, *Throwing*, *Knocking over*, *Pushing*, *Pulling or tugging*, *Biting*, *Hurting self*, *Obscene gestures*, and *other*. Each sub-category activity contains very few positive examples. Due to insufficient positive examples, we decided to train models for superordinate behaviors instead of each sub-category activity. Due to complexity of the detailed behavior category, the activity recognition performance drops dramatically from the movement behavior category. We still believe our framework can achieve a robust performance for each sub-category given enough training data. Our performance here also shows that we need to incorporate audio features to explore some activities related to verbal behaviors.

The CareMedia dataset is a real world surveillance video dataset, containing interactions between people, cluttered background, occlusion to activities, and changes in the environment. It is not a clean labratory dataset for researchers just to evaluate their algorithms. The data from Camera 133 was collected over 25 days which exhibited a lot of varieties and presented a big challenge for recognition.

Activity	Random	Laptev	MoSIFT
Walking though	36.67%	69.97%	84.68%
Walking to a standing point	22.94%	54.24%	72.31%
Standing up	3.48%	32.75%	47.29%
Sitting down	3.61%	34.41%	53.11%
Object placed on table	17.80%	29.91%	51.17%
Object removed from table	13.49%	36.90%	42.87%
Wheelchair movement	1.70%	18.10%	16.83%
Communicates with staff	0.32%	1.31%	1.77%
Average	12.50%	34.70%	46.25%

Table 3.4: MoSIFT provides the robust activity recognition performance in the CareMedia dataset on the movement activity category. MoSIFT significantly outperforms the Laptev's method here because MoSIFT is able to better capture smooth activities. The performance is measured by average precision.

Activity group	Random	Laptev	MoSIFT
Pose and/or motor action	12.13%	20.98%	26.13%
Positive activities	32.38%	30.45%	37.83%
Physical aggressive activities	1.46%	4.12%	4.02%
Physical non-aggressive activities	22.90%	28.12%	28.24%
Verbal aggressive activities	0.80%	1.12%	1.99%
Verbal non-aggressive activities	8.20%	9.91%	11.32%
Staff activities	20.68%	24.81%	27.11%
Average	14.08%	17.07%	19.87%

Table 3.5: MoSIFT provides robust activity recognition performance in the CareMedia dataset for the detail behavior category. Given each behavior here contains many sub-category activities. The performance drops dramatically from the movement activity category. We believe more positive training examples from each sub-category can significant improve the detail activity recognition results. The performance is measured by average precision.

3.5 Summary

A new video feature descriptor, MoSIFT, is proposed in this chapter. MoSIFT explicitly describes both appearance and motion of a interest region at multiple scale from a video. We successfully build an activity recognition framework based on MoSIFT. The activity recognition framework consists of interest point extraction, video codebook construction/mapping, bag-of-word feature representation, and modeling. Robustness is demonstrated by applying the framework to four different datasets. The evaluation on the KTH dataset shows the proposed algorithm outperforms the state-of-the-art methods significantly. The evaluation on the Hollywood dataset demonstrates that the proposed method performs well with camera motions on the edited movie scenes. The evaluations on the Gatwick and CareMedia datasets further shows that our framework is able to recognize interesting activities accurately in real-world surveillance video archives.

Chapter 4

Improving the robustness of MoSIFT activity recognition

In the bag-of-feature (BoF) framework, building a efficient video codebook can be the key factor to the performance. In BoF, each codeword is independent of the others. This assumption simplifies the relationships between different codewords and allows BoF to be constructed easily and efficiently. In video analysis, this assumption generally ignores the sequence information in both spatial and temporal domains which also provide essential information. Exploring spatial and temporal sequence information in BoF representations is an on-going research topic.

In this chapter, we try to improve the robustness of our MoSIFT activity recognition by constructing a more informative BoF representation. Three algorithms are proposed here: a constraint-based video interest point clustering approach, a bigram model, and a soft-weighting scheme. Constraint-based video interest points add temporal constraints during the clustering process to construct a video codebook with sequential information. The bigram model tries to embed spatial and temporal sequence information by adding frequent co-occurring interest point pairs in both spatial and temporal domains. The soft weighting scheme changes the codebook mapping process to a probabilistic mixture model. Each interest point is represented by a mixture of several codewords through probabilities instead of being assigned to one codeword (hard weighting).

4.1 Constraint-based Video Interest Point Clustering

The MoSIFT interest point detector tends to detect a good number of interest points from moving objects. Therefore, we frequently extract interest points from the video which are both spatially and temporally nearby. By visually examining our clustering results, we discovered that the clustering algorithm is sometimes too sensitive. It occasionally separates continuous components into different clusters. These components come from the same image location along a time sequence, and one would intuitively expect them to be clustered into the same group. This mainly happens for two reasons. First, the method we use to detect interest point tends to extract rich features with large dimensionality. Ideally, we would only extract points along representative moving points from local maxima in one area. However, our approach extracts a large number of video interest points and some of these only have small differences in the high-dimensional feature space. During the clustering process, this small difference can cause conceptually similar interest points to be separated into different clusters due to an over-sensitivity of the clustering algorithm. The second reason is related to the cluster center point initialization and distance function in the clustering algorithm. These two factors can greatly impact the clustering result and ultimately the activity classification accuracy. Cluster center point initialization makes the clustering result unstable because the initial center points may not be appropriate for the current dataset, and forcing clustering result to descend into a locally optimal solution which isn't well suited to the recognition task. In a high dimensional feature space, a distance metric can dramatically affect the shape of clusters' boundaries and the clustering result as well. In our proposed method, we would like the spatially and temporally co-located components to be clustered into the same cluster. Therefore, we introduce a pair-wise constraint clustering algorithm to force video interest points which are spatially and temporally nearby to be clustered into the same cluster during the clustering process. Figure 4.1 shows a pair of constraints from the boxing action in the KTH dataset.

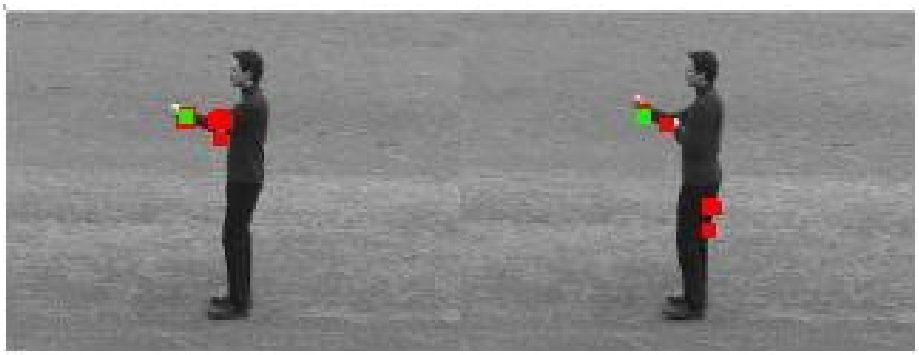

Figure 4.1: Red points indicate interest points extracted from the motion and green points show a pair of constraints which are considered as continuous, related components. The right frame is 5 frames after the left frame.

4.1.1 K-means Clustering

K-Means is a traditional clustering algorithm which iteratively partitions a dataset into K groups. The algorithm relocates group centroids and re-partitions the dataset iteratively to locally minimize the total squared Euclidean distance between the data points and the cluster centroids. Let $X = \{x_i\}_{i=1\sim n}$, $x_i \in \Re^m$ be the set of data points. n denotes the total number of data points in the dataset. m is the dimensionality of feature for data points. We denote $U = \{u_j\}_{j=1\sim K}$, $u_j \in \Re^m$ as centroids of clusters and K is the number of clusters. $L = \{l_j\}_{j=1\sim n}$, $l_j \in \{1 \sim K\}$ denotes cluster label for each data point in X. The K-Means clustering algorithm can be formalized to locally minimize the objective function as follows:

$$O_{k-means} = \sum_{x_i \in X} D(x_i, u_{l_i}) \tag{4.1}$$

$$D(x_i, u_{l_i}) = ||x_i, u_{l_i}||^2 = (x_i, u_{l_i})^T (x_i, u_{l_i}) \tag{4.2}$$

where $O_{k-means}$ is the objective function of K-Means and $D()$ denotes a distance function, which is the Euclidean distance. The EM algorithm can be applied to locally minimize the objective function. In fact, K-Means can be seen as mixture of K Gaussians under the assumption that Gaussians have the identity matrices as covariance matrices and uniform priors. The objective function is the total squared Euclidean distance between a data point to its center point. There are

three steps to achieve K-Means with the EM process: initialization, the E-step and the M-step. We first initialize K centroids in the feature space and then start to execute the E-step and M-step iteratively until the objective function converges or the algorithm reaches the maximal number of iterations. In the E-step, every point is assigned to the cluster that minimizes the sum of the distance between data points and centroids. The M-step updates centroids based on the grouping information computed in the E-step. The EM algorithm is theoretically guaranteed to monotonically decrease the value of objective function and to converge to a locally optimal solution. As we mentioned before, an unfortunate centroid initialization can sometimes result in a less-than-ideal locally optimal solution and clustering result.

4.1.2 EM Clustering with Pairwise Constraints

In the original K-Means algorithm, data points are independent of each other. However, in our proposed method, video interest points could have either spatial or temporal dependencies between each other. Our idea is to add constraints to video interest points which are both spatially and temporally nearby, increasing their chance of being clustered into the same prototype. Although we are not tracking interest points in our framework, we want to pair video interest points which are from the same activity motion component and encourage them to cluster into the same prototype.

Semi-supervised clustering algorithms have been getting more attention in recent years. These methods use data labels in the clustering process and significantly improve the clustering performance. Basu et al. [8] proposed adding pairwise constraints in a clustering algorithm to guide it toward a better grouping of the data. Their algorithm reads manually annotated data and applies this information to the clustering process. They have two different types of relationships between data: must-link pairs and cannot-link pairs. Their idea is very simple. Penalties will be added to the objective function if two data points which are labeled as must-link belong to different clusters during the clustering process. If two points are labeled cannot-link but belong to the same cluster during the clustering process, penalties will also be added. In our proposed method, we will

only penalize pairs which are spatially and temporally nearby (which we therefore consider potential continuous components) but belong to different clusters. This is the same as the must-link relation in Basu's method. However, we do not need to manually label the data points. The constraint pairs we generate are purely from the observed video interest points, and their spatial and temporal proximity; therefore they are pseudo-labels in our framework.

To achieve this, we revise the objective function of the K-Means clustering process as follows:

$$O_{constraint} = \sum_{x_i \in X} D(x_i, u_{l_i}) + \sum_{(x_i, x_j) \in X_{near}} \frac{1}{D(x_i, x_j)} \delta(l_i \neq l_j) \tag{4.3}$$

$$\delta(true) = 1, \delta(false) = 0 \tag{4.4}$$

The first term of the new objective function remains the same as K-Means. The second term represents our idea to penalize pairs which are considered to be continuous components but do not belong to the same cluster. X_{near} denotes to the set which contains spatially and temporally nearby pairs. The function equals one if two data points are not in the same cluster. In the second term, we can see that the penalty is correlated to the inverse distance between the two data points. Theoretically, two continuous components should be very similar in feature space because they are part of the same motion unit over time. Based on this assumption, the penalty is high if they do not belong to the same cluster. However, two exceptions may happen. The motion is too fast or the motion is changing. If the motion is too fast, we may link different parts together no matter how we define "spatially and temporally nearby". We can try to set up a soft boundary instead of a hard boundary to weaken the strict definition. In practice, we extract thousands of video interest points from our data set. It is not tractable to use soft bounds for all interest points, given that n-squared pairs are involved in the EM process. Therefore, we may occasionally mis-label two different interest points as must-link and penalize them if they are not in the same cluster. The other reason we may mis-label data pairs comes from changing motion. Since we try to constrain spatially and temporally nearby interest points as pairs, we have a good chance of linking two points from two different actions which transition seamlessly. Since

we neither track interest points nor analyze the points' spatial relationship, we can not avoid these exceptions when we try to connect video cubes with clustering constraints. However, we can reduce the penalty for these mis-labeled pairs. In both types of exceptions, we believe these pairs should have large differences in the feature space. This means that the distance between the two video interest points should be large, resulting in a small penalty. Instead, the objective function will be penalized more when a pair that looks similar in feature space is not in the same cluster. The objective function will be penalized less if the pair is actually quite different in feature space which hopefully means the pair does not originate from one continuous motion.

In our work, we replace the Euclidean distance in K-Means by the Mahalanobis distance to satisfy the Gaussian assumption for partitioning data points. The Mahalanobis distance function is:

$$D(x_i, u_{l_i}) = ||x_i, u_{l_i}||^2 = (x_i, u_{l_i})^T A_{l_i} (x_i, u_{l_i}) \tag{4.5}$$

A_{l_i} is a m by m diagonal matrix called covariance matrix. Because we update our distance function, we need to also revise the distance function between two points since they may belong to two different Gaussians. The formula for our pair-wised constraint clustering algorithm can be written as:

$$O_{constraint} = \sum_{x_i \in X} D(x_i, u_{l_i}) + \sum_{(x_i, x_j) \in X_{near}} \frac{1}{D'(x_i, x_j)} \delta(l_i \neq l_j) \tag{4.6}$$

$$D(x_i, u_{l_i}) = ||x_i, u_{l_i}||_{A_{l_i}}^2 = (x_i, u_{l_i})^T A_{l_i} (x_i, u_{l_i}) \tag{4.7}$$

$$D'(x_i, x_j) = \frac{1}{2}(||x_i, x_j||_{A_{l_i}}^2 + ||x_i, x_j||_{A_{l_j}}^2) \tag{4.8}$$

$$\delta(true) = 1, \delta(false) = 0 \tag{4.9}$$

The distance function, $D'(x_i, x_j)$, between two data points considers a mix of distances from both Gaussians. The optimization process still relies on the EM process. The only difference is in the M-Step, where we not only update centroids but also update the covariance matrices for the clusters. Figure 4.2 illustrates the idea of K-mean clustering with pair-wise constraints.

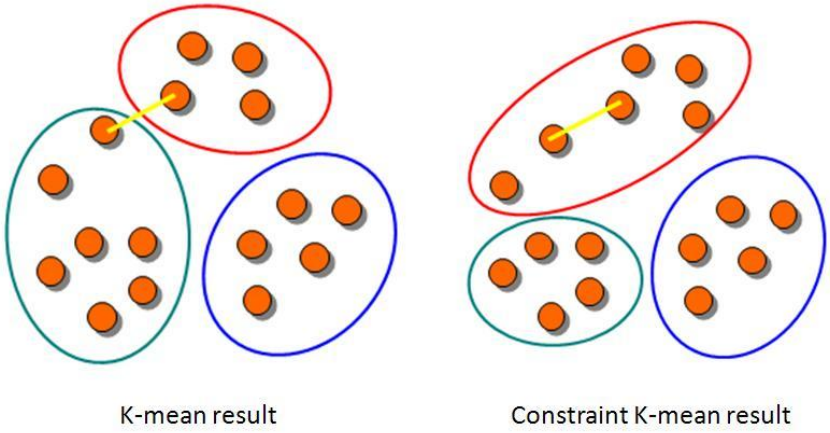

Figure 4.2: The left picture demonstrates regular K-mean clustering result. The yellow line here indicates a constraint. The right picture demonstrates how clustering result can be changed based on the added constraint.

4.1.3 Experimental results

We tested our proposed constraint-based clustering in the standard KTH dataset. In this experiment, we evaluated our constrained clustering on a more general video interest point descriptor, HoG. We did not apply this in MoSIFT because MoSIFT has reached 95% accuracy and it would be difficult to demonstrate performance improvements. The HoG descriptors basically extract histograms of gradients from interest points. We used 600 video codewords determined via cross-validation. We set up a hard boundary of "spatially and temporally nearby points" with a 2x2x5 window size, 2 pixel distance difference in both the x and y axis and for interest points extracted within 5 frames. This may not be the optimal setup, however, we want to evaluate in principle if constraints can improve recognition performance. Among 1.6 million video interest points extracted from the KTH dataset, we obtained around 0.38 million pairs fulfilling our definition. We randomly sampled constraints and added them into the clustering process in different amount. Figure 4.3 shows the recognition performance with different numbers of constraints added to clustering process. Figure 4.3 demonstrates that if we don't provide enough constraints, less accurate recognition will result. When we provide around 2500 pairs of constraints, the performance is statistically signifi-

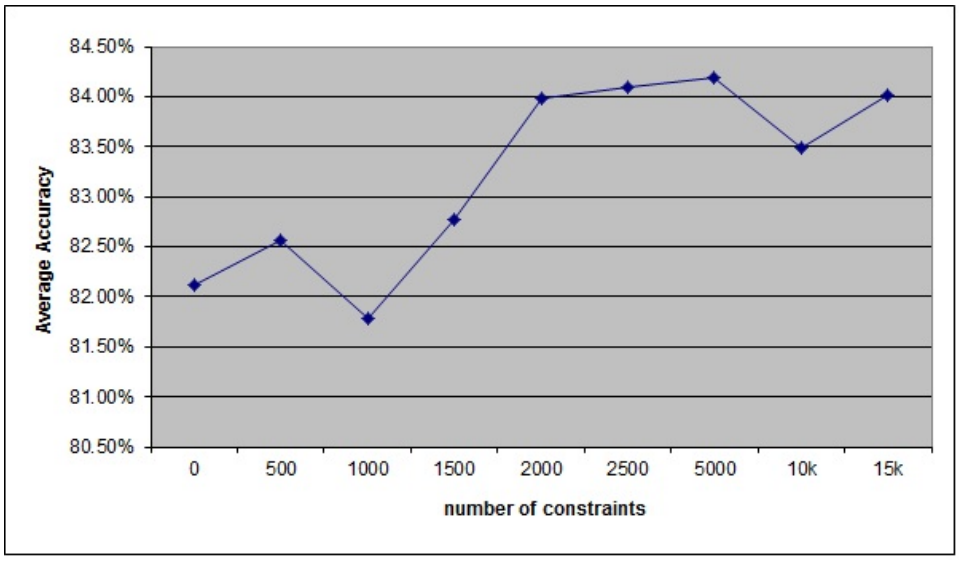

Figure 4.3: We evaluated how sensitive the performance of our algorithm to the number of constraints in KTH dataset. In the KTH dataset, it shows 2500 constraints will significantly improve activity recognition results.

cantly better than the baseline (84.28% vs. 86.39%) by a 95% confidence interval. In any case, additional constraints do not hurt performance. The performance numbers after 2500 constraints are not statistically different. The constraint-based clustering indeed stabilizes the clustering process and results significantly better recognition accuracy. Beside its improved result, the proposed constraint-based clustering algorithm can also apply to Dollar's and Laptev's methods. It does not require additional assumptions as long as a "spatially and temporally nearby" boundary can be defined. In general, constraint-based clustering stabilizes the clustering result and makes a more consistent video codebook.

4.2 Bigram model of video codewords

The bag-of-words feature representation is often used to represent an activity using spatio-temporal interest points. A video codebook is constructed by clustering spatio-temporal interest points. Each interest point is then assigned to its closest vocabulary word (a cluster) and the histogram of video words is computed over a space-time volume to describe an activity.

A bag-of-words feature representation is easy to compute and efficient for describing an action. However, its histogram does not contain any spatial or temporal constraints, which leads to loss of shape and periodicity information. In text analysis, a bigram model is often used to capture the co-occurrence of adjacent words in order to boost classification results [9]. This inspired us to build a bigram model in video codewords. Although it is computationally intractable to model all possible sequences of video codewords in a space-time volume, co-occurrence of only two video words requires minimal computation and provides some spatial and temporal constraints that help model shapes and motions.

4.2.1 The bigram model

Bigrams are a way to apply pair-wised constraints in a bag-of-word representation. Through these constraints on video codewords, additional spatial structure and temporal information can be embedded into bigrams. We first define adjacent video words as a pair of video words which co-occur in a kernel where d_s and d_t denote the spatial and temporal boundary. Experience has shown that good vocabulary sizes for action recognition are in the range of a hundred to a thousand words. Pair-wise correlations can result in very large numbers of pairs. Some research [74, 75] reduces the number of correlations by clustering. Instead, we select bigrams based on their tf-idf weights (term frequency-inverse document frequency) which is common in information retrieval and text classification. Term frequency (tf) is the frequency of a bigram in the dataset. Inverse document frequency (idf) indicates how informative a bigram is by dividing the number of all activities by the number of activities containing this bigram, and then taking the logarithm of the quotient. All bigrams can then be ranked by their tf-idf weights and we pick a sufficient number of bigrams to provide extra constraints to enrich the bag-of-word features and boost activity classification performance.

As we pick n bigrams with video codebook of m vocabularies, the histogram size will be $n + m$. We calculate the histogram as a vector:

$$H(i) = \frac{1}{|p_i|} \sum_{p \in \{p_i\}}^{|p_i|} \frac{1}{|C|} \sum_{c \in C}^{|C|} h(p, c) \tag{4.10}$$

$$h(p,c) = exp(-gD(p,c)) \tag{4.11}$$

where p_i is the set of interest points with vocabulary label i and $|p_i|$ is the size of this vocabulary. C is the set of interest points around interest point p and $h(p,c)$ is a weighting function for a pair of interest points. If the pair is far apart, it contributes less to the histogram. g is a fixed parameter of $h(p,c)$ and $D(p,c)$ measures the distance between interest points, a Euclidean distance in our case.

4.2.2 Experimental results

We first evaluate bigram constraints on the KTH dataset. We obtained pair-wise constraints to enrich local features with shape and time sequence information by using a bigram model. We added bigrams our bag-of-word representations in two different ways: the MoSIFT detector with non-aggregated HoG and HOF descriptors and the MoSIFT detector with full MoSIFT descriptor (aggregated HoG and HOF). The size of the kernel is 5x5x60, which is 5 pixels in the spatial dimensions and 60 frames in the temporal dimension. The number of bigrams we used was 300, which was determined to be reasonable through cross-validation. In fact, cross-validation shows that the first 300 bigrams significantly improve recognition performance. Beyond that, performance initially remains stable and eventually declines slightly as the number of bigram increases further. Table 4.1 shows that the bigram model improves weaker descriptors by a substantial amount from 89.2% to 93.3% and statically significant by a 95% confidence interval. However, it provides only a small improvement over the MoSIFT descriptor (95.83% to 96.2%). The high accuracy of the MoSIFT detector and descriptor at 95.83% means that among 24 actions a subject performs, only 1 action is misrecognized. For certain actions in KTH such as running vs. jogging, we found that even humans have difficulties in distinguishing them.

We further evaluate the bigram model on the Gatwick surveillance video collection. The kernel size is again set up as 5x5x60. 600 bigrams are applied though cross-validation in Gatwick collection. Table 4.2 again demonstrates improvement by adding global information though bigrams. The good bigram model slightly improves recognition performance on all activities in the Gatwick collection. By applying a t-test, the improvement is statistically significant given a 95% confi-

Method	Accuracy
MoSIFT with Bigram	$\mathbf{96.2}\%$
MoSIFT	95.83%
HoG + HoF with Bigram	$\mathbf{93.3}\%$
HoG + HoF	89.15%

Table 4.1: Adding bigrams into the bag-of-word representation significantly improves weak video interest point descriptors (HoG + HoF). Due to the already high performance of the MoSIFT descriptor, the improvement of adding the bigram model is limited. The evaluation is applied to KTH dataset.

Activity	Random	MoSIFT	MoSIFT with Bigrams
CellToEar	6.98%	22.72%	22.79%
Embrace	8.03%	29.55%	31.13%
ObjectPut	18.03%	46.81%	49.12%
PeopleMeet	22.32%	41.12%	45.57%
PeopleSplitUp	13.63%	58.33%	61.13%
Pointing	26.11%	44.24%	44.35%
PersonRuns	4.95%	36.78%	40.79%
Average	14.29%	39.94%	$\mathbf{42.13}\%$

Table 4.2: Bigrams capture some global information and slightly improve activity recognition performance in Gatwick surveillance video collection. The performance is measured by average precision.

dence interval.

4.3 Keyword weighting

Term weighting is known to have critical impact on text document categorization. Visual codewords are fundamentally different than text words. Each text word has its semantic meaning and naturally contains language context. Visual codewords are formed by data clustering where each codeword is distinguished from other codewords in the feature space. In other words, each codeword is not guaranteed to contain any semantic meaning but is only statistically similar. In the worst case, different codewords can actually represent the same context due to unsuitable clustering methods.

In visual bag-of-features, conventional term frequency (tf) and inverse docu-

ment frequency (idf) are widely used [52, 79, 99]. In [66], binary weighting, which indicates the presence and absence of a visual word with values 1 and 0 respectively, is used. However, all the conventional weighting schemes are performed after visual codeword construction which is the nearest neighbor search in the vocabulary (codebook) in the sense that each interest point is mapped to the most similar visual code (i.e., the nearest cluster centroid). This process is critical. Each interest point is then a code without its raw feature after this stage. A wrong assignment can not be corrected later. For example, two interest points assigned to the same visual codeword are not necessarily equally similar to that visual codeword, meaning that their distances to the cluster centroid are different. Ignoring their similarity with the visual word during weight assignment causes the contribution of two interest to be points equal, thus making it more difficult to assess the importance of a visual codeword in an image or a video. Therefore, the direct assignment of an interest point to its nearest neighbor is not the best choice.

4.3.1 Soft weighting

In order to tackle this problem, Agarwal et al. [3] proposed a probabilistic mixture model approach to train the distribution from local features and code new features by posterior mixture probabilities. This method is sophisticated and solves the aforementioned problem. However, it requires a training process which is not efficient for large scale datasets.

We propose a straight forward approach called *soft-weighting* to weight the significance of visual codewords. The basic idea is that one interest point will not be only assigned to one video codeword (cluster) but also share its importance with several related codewords in BoF. For each interest point in a video clip, we select the top-N nearest visual codwords instead of searching only for the nearest one. Suppose we have a visual codebook of K visual codewords, we use a K-dimensional vector $W = w_1, ..., w_k, ..., w_K$ with each component w_k representing the weights of a visual codeword k in a video clip such that

$$w_k = \sum_{i=1}^{N} \sum_{j=1}^{M_i} \frac{1}{2^{i-1}} sim(j, k) \tag{4.12}$$

Weighting schemes	Accurancy
tf	95.83%
soft-weighting	$\mathbf{96.58}\%$

Table 4.3: The soft weighting scheme slightly improves performance of KTH dataset. Given MoSIFT has a very high baseline already (95.83%), the improvement isn't significant.

$$sim(j,k) = \frac{1}{rank_{j,k}} \tag{4.13}$$

where M_i represents the number of interest points whose ith nearest neighbor is visual codeword k. $sim(j,k)$ is a measurement which represents the similarity between interest point j and visual codeword k. $rank_{j,k}$ is the rank of visual codeword k to interest point j. Empirically, inverse rank ($\frac{1}{rank_{j,k}}$) gives more stable performance than the distance functions from our experimental results. We find $N = 5$ is a reasonable setting from cross validation.

By using the proposed soft-weighing scheme, we expect to address the fundamental problems of weighting schemes which are originally designed for the text categorization domain.

4.3.2 Experimental results

We first evaluated the soft-weighting scheme on the KTH dataset. We set up soft-weighting on distributing video codeword weights to 4 closest clusters instead of the closest cluster used in hard-weighting. The performance is shown on Table 4.3. From the result, the soft-weighting doesn't improve the performance significantly. The reason is MoSIFT already has very high performance on the KTH dataset (95.8%). Therefore, we try to evaluate soft-weighting on a large, real TV-program dataset, the TRECVID 2009 Sound and Vision dataset.

We evaluate the soft-weighting scheme on the TRECVID 2009 Sound and Vision dataset - a popular and huge video dataset for semantic retrival. We applied the MoSIFT activity recognition algorithm for high-level feature extraction evaluation. In the experiment, we also want to demonstrate that MoSIFT is an efficient and robust video feature to detect semantic concepts in video content.

TRECVID temporally segments videos into basic units called shots. The high-

Weighting schemes	SIFT	MoSIFT
tf	6.64%	8.95%
tf-idf	6.71%	9.17%
soft-weighting	**8.90%**	**11.66%**

Table 4.4: The soft weighting scheme significantly improves performance of both SIFT and MoSIFT from hard weighing schemes imported from the text retrieval domain. MoSIFT is demonstrated as a powerful video feature for semantic video concept extraction. The evaluation is applied in TRECVID 2009 Sound and Vision dataset and measured by average precision.

level extraction task is to classify each shot and recognize target concepts. In our framework, we construct a BoF for each shot with 2000 video codewords by cross-validation. In the experiments, we use the 20 semantic concepts which are selected in the TRECVID-2009 evaluation. These concepts cover a wide variety of types, including objects, indoor/outdoor scenes, people, activities, etc. Note this dataset is a multi-label dataset, which means each shot may belong to multiple classes or none of the classes.

Currently, SIFT is a robust and popular feature to extract semantic concepts. Here, we evaluate our soft weighting scheme on both SIFT and MoSIFT to demonstrate that the algorithm can generally improve BoF of any type. Average precision (AP) is used to measure the performance here. The result is summarized in Table 4.4. The experimental result shows that the soft-weighting algorithm outperforms the popular weighting schemes from text retrieval domain. The result is not surprising since the soft-weight scheme preserves more information from low level features which is the key difference to the text domain. The result also demonstrates that the soft-weighting scheme works for both image and video BoF representation.

We further compare performance of SIFT and MoSIFT in more detail. We first defined activity related concepts as dynamic concepts which are 7 concepts among 20 concepts: {**Airplane flying**, **Singing**, **Person playing a musical instrument**, **Person riding a bicycle**, **Person eating**, and **People dancing**}. The performance comparison is shown in Table 4.5. It is not surprising that MoSIFT significantly outperforms SIFT in this category (15.22% vs 9.02%). However, the experimental result also shows that MoSIFT still outperforms SIFT in static concepts which

Concept category	SIFT	MoSIFT
Static concepts (13)	8.85%	**9.73**%
Dynamic concepts (7)	9.02%	**15.22**%

Table 4.5: MoSIFT outperforms SIFT in both static concept and dynamic concept categories. There are 13 static concepts which include object and scene concepts. 7 concepts are related to activities and defined as dynamic concepts. The evaluation is applied in TRECVID 2009 Sound and Vision dataset and is measured by average precision.

are objects, scenes, and people related concepts. By analyzing the result, we discover that MoSIFT gives the focus to moving objects in video shots by filtering background noise. It then improves performance for object and people related concepts but SIFT retains its advantage on analyzing scene concepts.

4.4 Summary

In this chapter, we introduced three algorithms to enhance the bag-of-feature representation. The constraint-based interest point clustering approach tends to cluster spatially and temporally similar video interest points into the same clusters. This approach considers the spatial and temporal relationships in the clustering process which improves the recognition performance in the KTH dataset. Bigrams capture pairwise relationships based on co-occurrence within a spatial and temporal kernel. The bigram is represented as additional dimensions in the bag-of-word representation. In the Gawick surveillance video collection, we successfully demonstrate the improved performance from the bigram model. The soft-weighting scheme releases one-to-one video codeword mapping by share the similarities to several codewords. This is similar to building a probabilistic mixture model from local features. This approach significantly improves the recognition performance in the TRECVID 2009 Sound and Vision dataset. In summary, modeling spatial and temporal relationships is a promising way to capture global information and enhances the bag-of-word representation. Our proposed methods successfully validate this idea.

Chapter 5

Activity detection

The proposed activity recognition framework from Chapter 3 extracts MoSIFT features from a video segment, represents this segment as a bag-of-feature, and classifies this representation into an interesting activity. The framework has an important assumption: the video segmentation has to be provided. A video is a sequence of still images and an activity happens in a sub-sequence of the images. An activity may start in any position of the sequence and last for an arbitrary length. The sub-sequence is the video segment we mentioned which is required by our proposed activity recognition framework. To determine a sub-sequence which contains an interesting activity is very challenging because it requires understanding the structure of the activity, which is what the recognition system attempts to learn. Therefore, the assumption of having the video segmentation is not realistic in real-world video. Activity detection detects when an activity starts and ends, and identifies what the activity is. It is the essential technique required in surveillance video analysis.

Activity detection not only identifies an activity of interest but also specifies when it happens and how long it lasts. In contrast to activity recognition, activity detection has to specify the time period of an activity, which is a temporal segment. A temporal segment defines the starting and ending time of an activity. Defining a temporal segment is a very subjective task. In our experience, even human users will have large disagreements in temporal segmentation when they annotate activities in a video. Therefore, detecting a temporal segment in a video is a very tough task. Inspired by face detection [73, 77, 84], we attempt to avoid

segmenting a video. Instead of a temporal segmentation, we formulate activity detection as a search and classification problem: a search strategy generates potential video segments and a classifier determines where or not they contain the interesting activities. A standard search approach is brute-force search, in which the video is scanned in a temporal order and over multiple scales. Each window will then be classified by activity models to determine the likelihood that the specific activity occurs in this window.

A brute-force search strategy usually faces the *rare event* problem when only very few windows are positive among a large amount of negative windows. This results in a very challenging task to train an accurate classifier. A classifier will usually be biased to negative examples given the priors and thus has very high false positive rates. Viola and Jones [89] proposed a face detection method based on a cascade of classifiers to solve the rare event problem and speed up face detection. Each classifier stage is designed to reject a portion of the non-face regions and pass all faces. Most image regions are rejected quickly, resulting in very fast face detection performance which also maintains high detection rates but low false positive rates. Inspired by Viola's method, we propose a cascade SVM classifier to reduce the false positive rate but keep high detection rates in activity detection.

5.1 Video temporal segmentation

Since accurate video segmentation is a subjective problem, we try to avoid predicting definite segmentation in an activity detection task. Instead, a general brute-force method is applied by sliding a fixed length window over time to generate potential video segments. The sliding window will have overlaps to cover all possible video segments. Note that we apply a fixed length window instead of multiple scale windows; we will discuss this decision. Figure 5.1 illustrates how we partition a video and segment an activity into a small number of temporal segments.

There are two advantages of applying this sliding window approach: *efficiency* and *robustness*:

- *Efficiency*: The sliding window strategy does not require computational efforts to analyze the content inside the window. Therefore, this brute-force

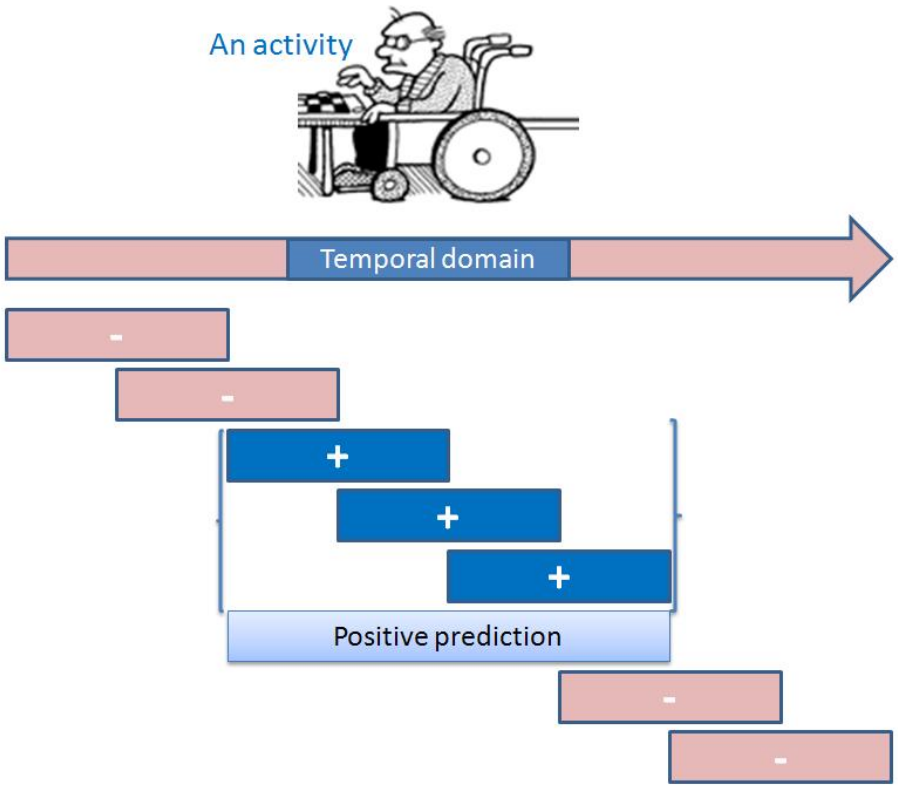

Figure 5.1: Illustration of the sliding window strategy. Blue windows indicate positive window and purple windows are annotated as negative. Concatenating positive windows (CPW) approach concatenates positive windows as an activity prediction shown by a light blue window.

approach can generate potential segments quickly. Furthermore, the strategy does not scan through multiple scales. The number of candidate windows is controlled in a reasonable number. For example, a window slides every 5 frame in a 25 fps video. 18,000 candidate windows are generated per hour. Activity models can be trained efficiently by the number of candidate windows generated by this approach.

- *Robustness*: This search strategy will not miss any potential segments since it slides every short temporal distance. A question arises here: given that we do not scan a video at multiple scales, how can ensure that we detect all activities of all lengths? A long activity is decomposed into couple candidate windows and a short activity is covered by a candidate window in this

search strategy. As long as an activity is not shorter than sliding temporal distance, it is covered by our candidate windows.

This strategy is based on two assumptions. First, each window has to be small enough to capture a unique portion of an activity but large enough to contain sufficient information to be classified accurately. Second, it requires a combination method which combines consecutive windows of an activity to achieve temporal invariance of the activity detector. The fundamental idea of this search strategy is that each window has unique and sufficient motion and shape information to be distinguished by classifiers. The classifier learns components of an activity instead of the whole activity. A simple combination strategy is applied by concatenating positive predicted windows (CPW) as a positive prediction. This search strategy provides an alternative way to achieve temporal invariance of activity recognition. Overall, this strategy will heavily rely on activity recognition performance. Our proposed MoSIFT activity recognition was proved to be a state-of-the-art method [20] to support this activity detection strategy.

5.2 Cascade SVM classifier on activity detection

Although the sliding window search approach has good properties, such as *efficiency* and *robustness*, it also has a major disadvantage: too many negative windows are generated. This results in a *rare event* problem when only very few windows are positive among a large number of negative windows. The classifier trained on this data will be biased to negative examples due to the priors and then has a high false positive rate. The cascade architecture fits well to this problem of maintaining a high detection rate but a low false positive rate. We propose a cascade SVM classifier to utilize the advantage of a cascade architecture and the robust performance of SVM. We will briefly introduce the concept of cascade architecture first.

A cascade architecture is illustrated in Figure 5.2. The key idea is inherited from AdaBoost which combines a collection of high precision classifiers to form a strong classifier. The classifiers are called weak because they are not expected to

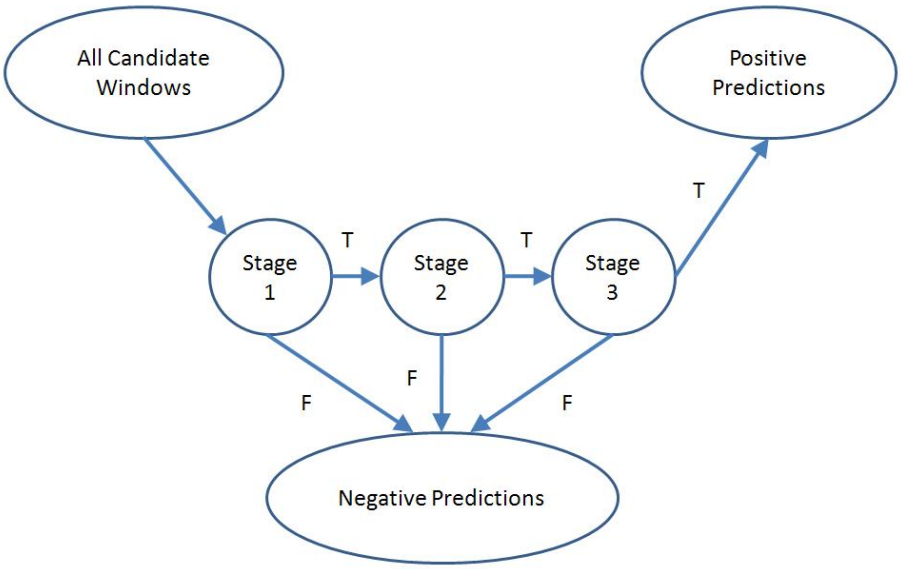

Figure 5.2: Illustration of the cascade architecture with 3 stages.

have the best performance in classifying all examples in the training data. In order to boost weak classifiers, each classifier emphasizes the examples which are incorrectly classified by the previous weak classifiers. In our detection task, simpler classifiers are first used to reject the majority of windows before more complex classifiers are called upon to achieve low false positive rates. In Figure 5.2, each stage demonstrates as a weak classifier. Each weak classifier keeps most of the positive examples but rejects a good number of negative examples. Face detection has shown that a cascade architecture can reduce false positives rapidly but keep a high detection rate. While we could have used AdaBoost similar to Viola and Jones [89], AdaBoost is sensitive to noisy data and outliers. Given that the interest points may be extracted from unrelated motions and the same activity has large variations, an activity classifier will face noisy data and outliers. We therefore proposed a cascade SVM classifier which is more robust to noisy data and outliers. We summarize our cascade SVM classifier implementation below.

Given a set of positive examples P and negative examples N, we construct a cascade SVM classifier s that achieves a high detection rate on the positive examples and a low false positive rate on the negative examples. For each node in the cascade, we randomly choose a set of negative examples $N^{'} \subset N$ that have been

Algorithm 5.1 Train a cascade SVM classifier

Input: positive examples P, negative examples N, number of maximum stages k
Output: a binary classifier s

1. $N^{'} \subset N$ that have been misclassified by pervious stages, where $|N^{'}| = |P|$. if $|N^{'}| < |P|$, then return s
2. Train a SVM s_i on $N^{'} + P$
3. Adjust SVM threshold in s_i which pass all P
4. $s = s + s_i$
5. if $i >= k$ return s else goto step 1

misclassified by previous stages, where $|N^{'}| = |P|$. A SVM [17] classifier is trained on P and $N^{'}$. We adjust the SVM threshold so that it passes all positive examples as true positive predictions and minimizes the false positive rate. Note that we don't train on selected features as in AdaBoost, we train the SVM classifier using the whole feature set. The reason is that the SVM classifier has better tolerance on noisy data and outliers. We then eliminate the negative examples that were correctly classified as negative and train the next stage in the cascade using the remaining examples. The stopping criteria for a cascade SVM classifier is when there aren't enough negative examples, or when it reaches the maximal number of stages we want to train. In the testing phase, if at any point in the cascade a classifier rejects the window under inspection, no further processing is performed and the search moves on to the next window. Only the windows that pass all classifiers are predicted as positive. The cascade therefore has the form of a degenerate decision tree. The algorithm of cascade SVM classifier is shown in algorithm 5.1.

The cascade architecture has interesting implications for the performance of the individual classifiers. Because the activation of each classifier depends entirely on the behavior of its predecessor, the false positive rate for an entire cascade is:

$$F = \prod_{i=1}^{K} f_i \tag{5.1}$$

Similarly, the detection rate is:

$$D = \prod_{i=1}^{K} d_i \tag{5.2}$$

where k indicates the number of stages in the cascade, and f_i, d_i indicates the false positive rate and detection rate for each stage respectively. Thus, to match the expected false positive rate, each classifier can have surprisingly poor performance. For example, for a 32-stage cascade to achieve a false positive rate of 10^{-6}, each classifier needs only to achieve a false positive rate about 65%. At the same time, each classifier needs to be exceptionally capable if it is to achieve a adequate detection rate. For example, to achieve a detection rate about 90%, each classifier in the aforementioned cascade needs to achieve a detection rate of approximately 99.7%.

5.3 Experimental results

We evaluated our proposed methods on the TRECVID 2008 surveillance dataset [85] which was collected at Lodon Gatwick International Airport. This dataset is evaluated in the official TRECVID event detection benchmark which multiple research groups participated in. This is the first surveillance dataset published which multiple detection algorithms developed from international research groups are evaluated. There were a total of 6,439 events in the development set which was annotated by NIST. It consists of 50-hours (5 days x 2 hours/day x 5 cameras) of videos in the development set and another 50-hours in the evaluation set which makes 9406519 frames in total (4709896 frames in the development set). Our sliding window search approach generates 941979 candidate windows in the development set (around 1.88 million windows total) by sliding every 5 frames with each window 25 frames (1 sec) in length.

The detection performance is measured as a tradeoff between two error types: missed detections (MD) and false positive (FP). The two error types will be combined into a single error measure using the Detection Cost Rate (DCR) model, which is a linear combination of the two errors. The DCR model distills the needs of a hypothetical application into a set of predefined constant parameters that in-

clude the event priors and weights for each error type. DCR is used to evaluate detector performance in TRECVID 2008 event detection evaluation.

An activity can occur at any time and for any duration. Therefore, in order to compare the output to the reference annotations, an one-to-one temporal mapping is needed between the system and reference observations. A system observation here is an activity detection and a reference observation indicates an annotation. The mapping is required because there is no pre-defined segmentation in the video. The mapping basically aligns an activity detection to an annotation if they have a overlap. If an activity overlaps more than one annotation, the activity will be mapped to the annotation which has a longer overlap and a higher detection score. The alignment formulas below assume the mapping is performed for a single event (E_i) at a time.

$$M(O_{s_i}, O_{r_j}) = \begin{cases} 0 & \text{if } Mid(O_{s_i}) > End(O_{r_j}) + \bigtriangledown_t \\ 0 & \text{if } Mid(O_{s_i}) < Beg(O_{r_j}) - \bigtriangledown_t \\ 1 + E_t * TimeCongru(O_{s_i}, O_{r_j}) + E_{DS} * DecScoreCongru(O_{s_i}) \end{cases} \tag{5.3}$$

$$TimeCongru(O_{s_i}, O_{r_j}) = \frac{Min(End(O_{s_i}), End(O_{r_j})) - Max(Beg(O_{s_i}), Beg(O_{r_j}))}{Max(\frac{1}{25}, Dur(O_{r_j}))} \tag{5.4}$$

$$DecScoreCongru(O_{s_i}) = \frac{Dec(O_{s_i}) - MinDec(s)}{RangDec(s)} \tag{5.5}$$

$$Detect(O_{s_i}) = \max_{\forall r_j \in r}(M(O_{s_i}, O_{r_j})) \tag{5.6}$$

where O_{s_i} is the ith observation of the event for the detector s, O_{r_j} is the jth reference observation of the event (from annotation), $Beg()$ indicates the beginning of the observation, $End()$ indicates the end of the observation, $Mid()$ indicates the middle point of the observation, $Dec(O_{s_i})$ is the detection score of the observation O_{s_i}, $MinDec(s)$ is the minimum decision score of s, $RangeDec(s)$ indicates the range of decision score from s, E_t and E_{DS} are two constants to weight time and decision score(set to $1e^{-}8$ and $1e^{-}6$ respectively), and $\bigtriangledown_t$ is set to 0.5 seconds (12.5 frames). $Detect()$ maps the system observation to the reference observation which comes up the highest mapping score $M()$. If $Detect()$ ends up 0, it is a false positive. Any reference observation which is not mapped with a system observation counts as a missed detection.

Activity	# positive	positive ratio
CelltoEar	8044	0.17%
Embrace	21920	0.47%
ObjectPut	13147	0.28%
PeopleMeet	52804	1.12%
PeopleSplitUp	63136	1.34%
PersonRuns	7987	0.17%
Pointing	24470	0.52%
Total	151908	4.07%

Table 5.1: Activity detection is a rare event problem. In the development set, there are 4.7 million candidate windows. Totally, only 4.07% of candidate windows contain interesting activities.

Given the definition of missed detection and false positive, the DCR model is formulated as follows:

$$DCR(s, E_i) = P_{Miss}(s, E_i) + Beta * P_{FP}(s, E_i) \tag{5.7}$$

where $P_{Miss}()$ is the rate of missed detection and $P_{FP}()$ is the false positive rate. $Beta$ is the weight to combine missed detection and false positive rates and it is set up as 0.005 in the evaluation provided by NIST. The measures unit is in terms of Cost per Unit Time which has been normalized so that an $DCR = 0$ indicates perfect performance and an $NDCR = 1$ is the cost of a system that provides no output, i.e. $P_{Miss} = 1$ and $P_{FP} = 0$.

Activity detection is a typical *rare event* problem. Using our search strategy (a 25 frame fixed window sliding for 5 frames), only 4.07% candidate windows contain at least one of interesting activities. The positive ratio of individual activity is shown in Table 5.1. From the table, it is noticeable that the positive ratios of interesting activities are mostly lower than 1%. This statistic demonstrates the need to train a cascade classifier to solve the *rare event* issue in the activity detection task.

We designed experiments to evaluate the cascade SVM classifier in the TRECVID 2008 event detection task. Since there are five cameras, we built a cascade SVM classifier for each activity in each camera. In each stage of the cascade, a SVM classifier is trained on MoSIFT bag-of-word features. We build a MoSIFT video codebook of 1,000 video vocabulary size from cross-validation. In these experiments,

Activity	single SVM	2 stages	6 stages	10 stages
CelltoEar	47.4	11.70	3.79	3.07
Embrace	45.2	11.67	4.08	3.33
ObjectPut	38.8	9.46	4.28	4.07
PeopleMeet	43.5	11.75	4.87	3.93
PeopleSplitUp	44.5	10.47	6.48	6.76
PersonRuns	54.8	13.42	6.33	4.52
Pointing	41.9	13.08	5.52	4.86
Average	45.2	11.65	5.05	**4.36**

Table 5.2: The comparison of cascade SVM classifiers with different numbers of stages. The cascade SVM classifier significantly improves detection performance on the TRECVID 2008 surveillance video dataset. The performance is measured as DCR.

our evaluation is measured as DCR proposed by NIST. The activity models are trained on the development set and tested on the evaluation set. Table 5.2 shows the performance. We build activity models for single SVM, 2 stage, 6 stage, and 10 stage cascade SVM classifier. The DCR keeps improving as we keep adding stages. However, after 10 stages, some activity models start to run out of negative examples to train further stages. In our experimental results, the DCR improvements mainly come from rapidly reducing the false positive rate but maintaining a high detection rate.

With sufficient and robust cascade SVM classifiers, we evaluate our activity temporal invariance strategy by concatenating positive windows (CPW) as a single positive prediction. The performance of CPW is demonstrated in Table 5.3. It is obvious that CPW further improves detection results in terms of reducing DCR. Our observations tell us that the concatenation strategy can further reduce the false positive rate but does not decrease the detection rate much.

5.4 Summary

We introduced a sliding window search strategy and a cascade SVM classifier to extend our MoSIFT activity recognition framework to achieve robust activity detections. This approach extends Viola and Jones' work for static-scene object detection to the spatio-temporal domain. Applying this framework on the

Activity	Cascade SVM	CPW
CelltoEar	3.07	2.75
Embrace	3.33	2.94
ObjectPut	4.07	3.30
PeopleMeet	3.93	3.28
PeopleSplitUp	6.76	4.19
PersonRuns	4.52	4.47
Pointing	4.86	3.57
Average	4.36	**3.50**

Table 5.3: The concatenating positive windows (CPW) approach not only significantly improves detection performance on the TRECVID 2008 surveillance video dataset but also achieves activity temporal invariance. The performance is measured by DCR. The proposed method is the top performance in the official TRECVID evaluation.

TRECVID 2008 surveillance video dataset, we learn this detection framework can detect activities in real-world surveillance videos and our detection system tops the performance at the official TRECVID evaluation. We successfully demonstrated that a cascade SVM classifier can reduce the false positives rapidly while maintaining a high detection rate. Our concatenating positive window approach not only achieves temporal activity invariance but also improves the detection performance.

In summary, the proposed activity recognition and detection algorithms constitute a comprehensive study of a video activity analysis framework. These techniques allow us to discover and identify interesting activities in the video. Especially in the health care domain, this study provides essential tools to build surveillance systems which automatically analyze patients' daily lives.

Chapter 6

Long term activity analysis

In the previous chapters, we discussed how to recognize (Chapter 3) and detect (Chapter 5) activities. In this chapter, we will discuss how to utilize activity analysis to study long term human activity from surveillance video archives. Long term activity analysis is a very challenging topic that is not well studied in surveillance video system research. We first give our definition of a long term activity analysis. In our definition, there are two types of long term activity analysis. The first is to measure the change over time from a person's daily activities to discover interesting trends. The second type is to summarize a person's activities over time to understand his/her daily life. Specifically, an observation longer than several weeks will be considered a long term analysis in this thesis. For example, observation of a person's eating habits over a month is a long term activity analysis. This analysis detects when and how much he/she eats every day. This analysis can provide information related to his/her weight and health. Multiple disciplines, computer vision, information retrieval, data mining and machine learning, jointly frame this research. In our opinion, there are three major topics to study to achieve long term activity analysis: *video activity analysis*, *temporal data collection*, and *long term pattern extraction*.

- ***Video activity analysis***: Activity analysis which includes activity recognition and detection is a research topic which is increasingly popular in computer vision research [29, 44, 53, 54, 60, 72, 91]. These techniques extract semantic units (activity related) from videos to improve the ability to search and mine. However, diversity of activities combined with camera motions

and cluttered backgrounds make video activity analysis extremely difficult for real-world applications. Our proposed methods in Chapter 3 and Chapter 5 give us a solid ability to analyze video content and further explore long term understanding.

- ***Temporal data collection***: Long term analysis is based on studying a topic over a long period of time. Learning over time is a growing research area in the machine learning and information retrieval fields, e.g. discovering trends in discussion forums [48, 80]. Collecting a suitable dataset to study is a challenging task. The collected data must not only last a long time but must also exhibit temporal changes or meaningfully different observations over that time.
- ***Long term pattern extraction***: Given observations over a period of time, finding a pattern can provide useful information to users [6, 19]. The pattern can be a summarization of the observations or a trend discovered from the observations. Finding a long term pattern is very domain specific. Domain knowledge is used to understand the information needed over time. Therefore, transforming the information need to a machine-learnable task that extracts the long term pattern is the key research goal that we want to explore.

Considering the three components we discussed above, we propose a case study of long term activity analysis on the CareMedia dataset [90]. CareMedia is a surveillance video collection where video activity analysis can be applied. Many activities can be observed visually by automatic systems. The CareMedia collection records the daily lives of the residents in a nursing home over one month. The dataset provides a suitable dataset to analyze long term activities. For example, a resident may walk less and less over the course of a month, which is observable from the dataset. Furthermore, there is a great desire to understand elderly patients' daily lives and medical doctors believe this is strongly related to the patients' overall health. Elderly patients' long term activity observations can provide an assistance to diagnose their health more accurately. For example, if we discover that a patient performed more positive activities, e.g. eating and walking, this normally indicates that his/her health is not getting worse.

6.1 Long term health care in nursing homes

Nearly 2.5 million Americans currently reside in nursing homes and assisted living facilities in the United States, accounting for approximately 5% of persons 65 years and older [63]. The aging of the "Baby Boomer" generation is expected to lead to an exponential growth in the need for some form of long-term care (LTC) for this segment of the population within the next twenty-five years. In light of these sobering demographic shifts, it is urgent to address the profound concerns that exist about the quality-of-care (QoC) and quality-of-life (QoL) of this frailest segment of our population. We will discuss traditional nursing home health care and computer aided health care in the following.

6.1.1 Traditional nursing home health care

Traditional nursing home health care is performed mainly by nursing staff. In nursing homes, nursing staff members not only provide care for residents' daily lives but also make notes of the interesting activities which have been designated by medical doctors. These notes help doctors to understand the patients' daily lives and make accurate diagnoses. Nursing staff members have been trained professionally to be able to maintain QoC and QoL of residents. Professional training not only gives them the necessary knowledge to provide health care but also to notice unusual mental and physical behaviors. Therefore, nursing staff members can be assumed to be capable to maintain QoC and QoL, and collect information to assist medical doctors.

However, the United States General Accounting Office (GAO) reported that in 2003 [68],

> One in five nursing homes nationwide (about 3,500 homes) had serious deficiencies that caused residents actual harm or placed them in immediate jeopardy ... Moreover, GAO found significant understatement of care problems that should have been classified as actual harm or higher - serious avoidable pressure sores, severe weight loss, and multiple falls resulting in broken noses and other injuries...

The GAO attributes the underreporting of such problems to:

- lack of clarity regarding the definition of harm
- inadequate state supervisory review of surveys
- delays in timely investigation of complaints
- predictability of the timing of annual nursing home surveys

Equally importantly, without methods to continuously record, monitor and document the care of these residents, it is exceedingly difficult to verify resident-specific data reported by nursing staff and review complaint investigations. These tasks would be greatly aided by automatic tools that enable accurate assessments of patient care and treatment. For example, we analyzed 320 camera-hours of data collected with 4 video cameras. In this data collection, nursing staff observed 4 physical aggressions but missed 3. Video recording observed all 7 physical aggressions [10]. This small analysis gives us a confidence that automatic tools (e.g. surveillance recording) can be a great help to current nursing homes health care.

In summary, although professional training gives the nursing staff the ability to maintain QoC and QoL for nursing home residents, deficiencies in nursing staff and lack of 24 hour supervision create a need to develop computer aided health care systems, which provide auxiliary protection in addition to nursing staff to ensure QoC and QoL of nursing home residents.

6.1.2 Computer aided health care

In the past decade, more and more devices have been developed to monitor and observe people's physical or mental state for health care purposes. For example, devices can be attached to beds, wrists, or heads to record brain waves and pose changes during sleep to understand sleep quality (see Figure 7.3). These data collected by health care devices can provide medical doctors insight into a patient, which doctors can use to make more accurate diagnoses or adopt more efficient treatments based on individual needs.

These devices are currently designed for special purposes only and are usually attached to the patient's body. The specialization enables us to collect interesting information accurately. These devices capture specific information about a narrow aspect of health, e.g. blood pressures or brain waves. However, health care in nursing homes requires not only these specific computer aided devices but

Figure 6.1: Several health care aided device examples. Top left one is a sensor attached to a bed to detect sleeping posture. Top right is a binary sensor to detect door status. Bottom left is a accelerometer in a watch to measure motor activities. Bottom right is a head band which collects brain waves to aid health care (sleeping quality).

also a general and unobtrusive approach which collects and observes residents' daily activities naturally. The reason to have a general method is that unexpected activities happen frequently in our daily lives and some activities are too complicated to be detected or measured by one device. An unobtrusive approach can observe patients naturally and decrease the inconvenience to the patient. Surveillance video recording is the prime example of a general and unobtrusive method. This method can capture complicated information but also increases the difficulty of developing an automatic analysis system. Recently, many researchers proposed utilizing sensors and video cameras to analyze people's daily activities to assist

QoC and QoL [36, 37, 57]. The sensors capture time, location and coarse appearances of specific activities in an area. Radio-frequency identification (RFID) is also widely applied to identify people in a nursing home. Combining multiple devices with different purposes is a way to observe a nursing home more generally. However, surveillance video provides an alternative way to observe and analyze people's behaviors naturally and directly. Although video recordings are more difficult to process automatically than sensors, video is complementary to sensor approaches because it can monitor interesting activities without requiring patients to wear devices. The CareMedia nursing home health care project was proposed to provide a general and unobtrusive solution to assist health care in nursing homes by video monitoring of the public portion of the nursing home environments.

6.2 CareMedia health care

Due to the great QoC and QoL needs of nursing home residents, the CareMedia project attempts to expose all aspects of residents' ongoing daily lives to medical doctors to help improve their health care through video monitoring. The 24-hours/7 days a week surveillance video monitoring not only records a lot of data but also stores detail which is required to understand patients' physical and mental conditions. Modern computer vision, information retrieval, data mining and machine learning techniques provide a good foundation to study behaviors associated with senile dementia from surveillance video.

The three CareMedia collaborative efforts are: data collection, human manual observation, and automatic observation. As we mentioned in section 1.6.5, the data collection was done by recording all the public areas of a nursing home over 25 days using 23 ceiling mounted cameras. A tremendous effort was made to locate cameras to ensure an un-occluded view of every point in the recorded space, synchronize video streams, and store a huge amount of encoded video. Post processing of this data is our major research focus. We categorize post processing into two types: human manual observation and automatic observation. Interactive multimedia retrieval techniques are applied to achieve efficient human manual observation, and computer vision and machine learning algorithms help us to

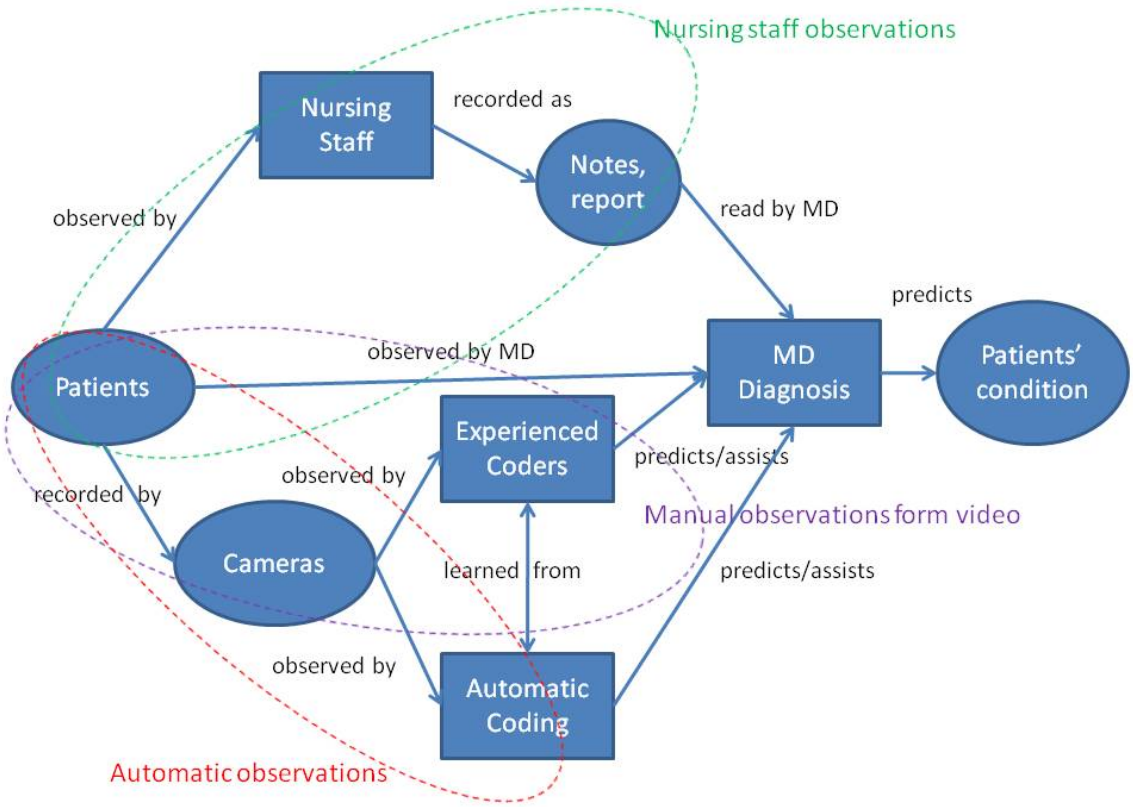

Figure 6.2: The CareMedia long term health care system conceptual architecture. Predicting patients' health condition accurately is an important goal of the CareMedia nursing home health care project. Patients' health conditions are represented by medical doctors' diagnoses. Therefore, combining three major approaches would significantly improve the quality of diagnosis. The three major approaches are nursing staff observations, manual observations by coders and automatic observations from surveillance videos.

observe interesting activities automatically.

The ultimate goal of the CareMedia is to help medical doctors to understand residents' health conditions. In this long term activity analysis work, we will focus on studying how to analyze activities over long periods of time to help medical doctors make a better diagnosis. Figure 6.2 illustrates our framework of this study. The upper part of the diagram shows the traditional nursing home health care system. The nursing staff members observe the daily activities from patients and record these observations to assist medical doctors in making the diagnoses. Of

course, doctors also observe patients directly in addition to these reports/notes. This process has many drawbacks that we discussed earlier. The major problem is that nursing staff can not keep their eyes on all residents all the time to observe every detail. However, some details may provide the critical information which medical doctors would require to improve their diagnoses. Surveillance video should theoretically record every single detail. Video recording not only contains the informative data but also has a tremendous amount of useless content. Post processing is then important to the extract useful information and then reduce the size of the data doctors must look at. Two post processing steps are applied in the CareMedia project: manual observation and automatic observation which are shown in the bottom part of the diagram. Combining these three sources (nursing staff observations, manual observations, and automatic observations) of patients' daily lives, we hope to improve the quality of medical doctors' diagnoses significantly and relate it more closely to the true health condition of the patients.

6.2.1 Manual observations

In addition to real time health care provided by nursing staff members, surveillance video can be used as an auxiliary method to improve QoC and QoL. Although this approach is not a real time process, it stores all the recorded activities and can be to reviewed repeatedly. There are three major steps to post process the collection: *indexing*, *annotating*, and *summarizing*. The indexing step enables the efficient annotating step. The summarizing step communicates the annotation to researchers and medical doctors clearly and efficiently.

1. ***Indexing***: Due to the large amount of video recorded for the CareMedia project, it's not possible to access video efficiently without indexing the data. The intuitive way to index the collection is sorting and storing by time and camera location. The basic retrieval method is to search a video by time and location.
2. ***Annotating***: Given a coding manual designed by medical doctors, experienced coders can annotate interesting activities in the surveillance video collection. This is the observation process which enters information observed from the collection into the database. The trained coders must understand

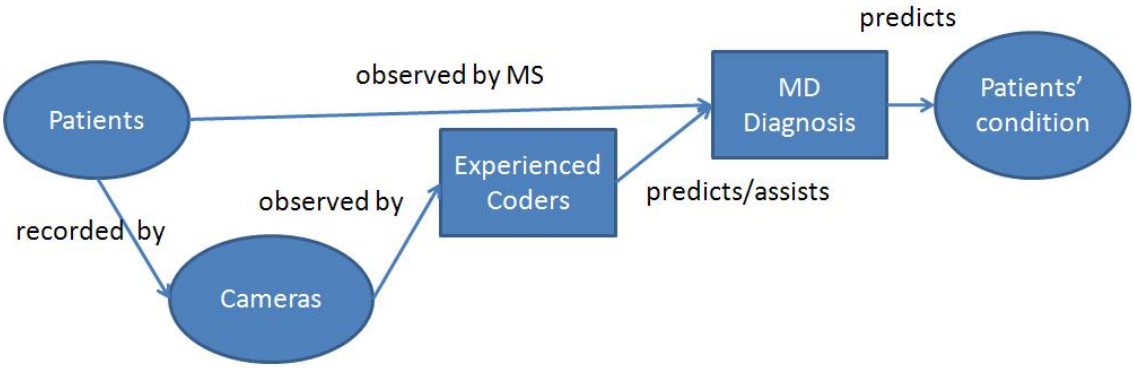

Figure 6.3: The CareMeedia long term manual observation diagram. Experienced coders annotate surveillance video which captures resident's daily activities and the observation coded by coders can become an informative source for medical doctors to make better diagnoses.

the clear definition of each activity to be annotated. They play the same role as nursing staff in observing residents' daily activities. The two major differences to real time nursing staff observations are the ability to review activities repeatedly, and the ability to comprehensively observe all public areas of the nursing home.

3. ***Summarizing***: The annotated data is stored in a database which can be searched by time, category, resident's name and location. The system can also generate histograms or statistical analysis to summarize a resident's daily activities to provide more information.

Annotating videos not only costs a tremendous amount of human time but also is a tedious task. Efficient and accurate annotations are needed to provide high quality information for further use [18, 95]. An annotation codebook was designed by medical doctors (see Appendix B). In the CareMedia project, there were two classes of codes. The first class contains 12 activities which have clear definitions and are highly related to movements. We call this class the movement activity category. The second class contains 7 superordinate behavior codes which are called the detailed behavior category. Each superordinate behavior code is composed of some subordinate behavior codes. The full CareMedia coding manual is included as Appendix B. To code efficiently, a coder is assigned a period of time and a location to observe. The video coding interface is shown in figure 6.4. Each

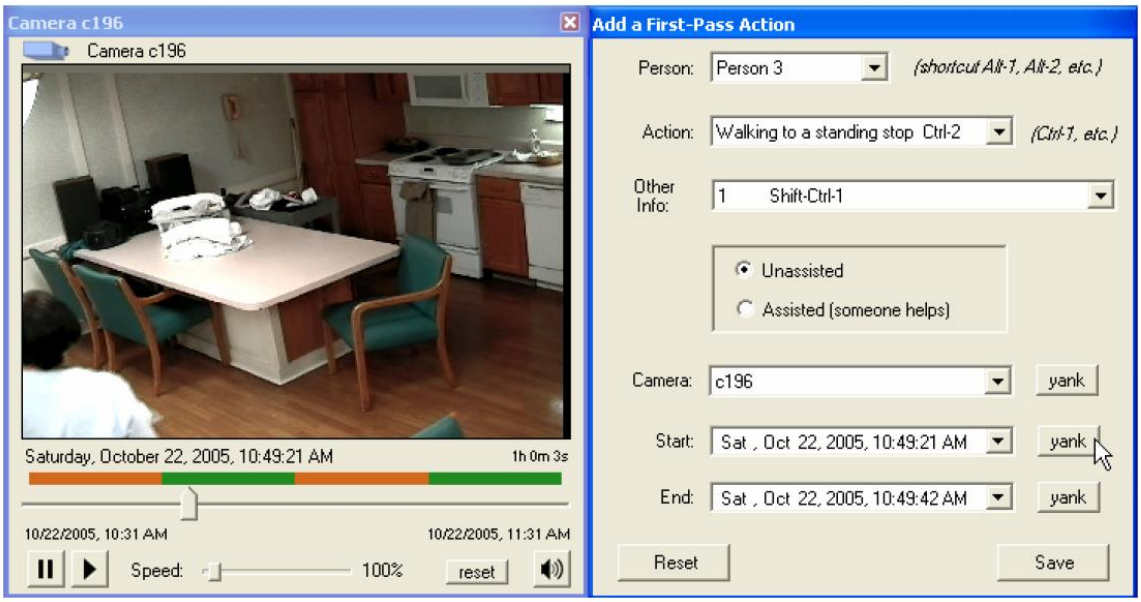

Figure 6.4: The CareMeedia manual coding interface, there showing the interface includes a video player for review and discovery, and the coding form.

time, the coder only tracks one person in order to annotate that person's activities as accurately as possible. Usually, there are multiple activities happening in the same scene. A coder will code one activity at a timeinstead of multi-tasking. A coder can review the video from $1x$ to $5x$ speed. Some simple computer vision filters (e.g. motion extraction) assist the coders to simplify the annotating task. Each position and time is reviewed by at least two coders. We begin the CareMedia manual observation coding at meal time (lunch and dinner) which contains the most activities in public areas.

The annotated data is stored in a database and can be retrieved by time, resident, location, and category. Figure 6.5 shows the event window for a retrieval result. This annotated data plays the same role as notes/documents observed by nursing staff but it is more complete. The manual observation (annotated data) provides detailed information to give medical doctors a more comprehensive view of a resident. Furthermore, this information can actually predict doctors' diagnosis fairly accurately. We will discuss this in more detail in chapter 6.3.1. Figure 6.3 illustrates the role of manual observation in CareMedia nursing health

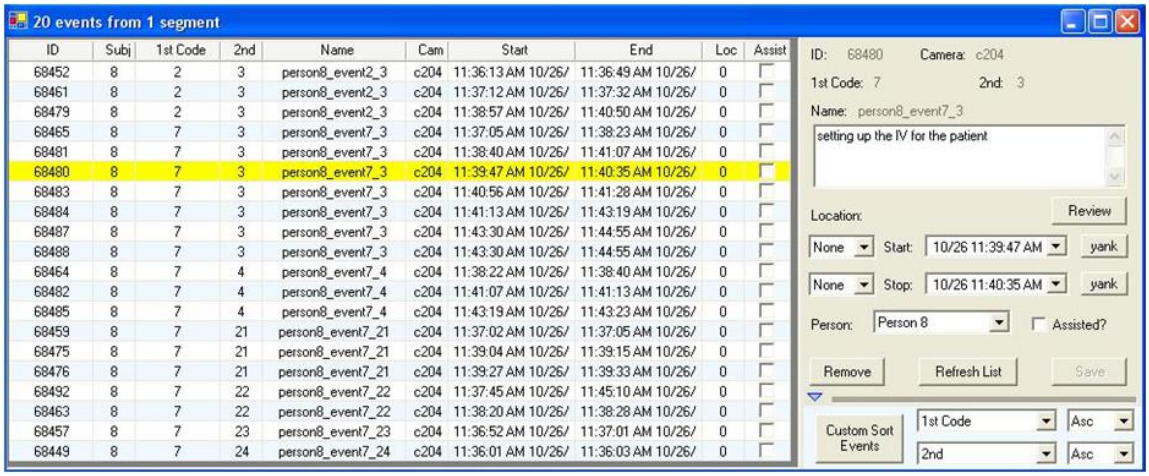

Figure 6.5: CareMedia event window to show annotated activities in the system. The system will show details of each event in the right panel. The event list can be filtered by time, location, resident and behavior type.

care. The manual observations can be an informative source for diagnoses or predict a diagnosis score accurately by machine learning to further improve doctors' judgements.

6.2.2 Automatic observations

In place of manual observations requiring much human effort, computer vision and machine learning provide an alternative way to observe residents' daily lives automatically. Activity recognition and detection are two algorithms which are capable of observing residents' activities automatically from surveillance videos [33]. This video analysis approach plays the same role as experienced coders. The machine applies established activity models to detect interesting activities in the video archive and saves the detection results into a database. This approach can save a tremendous amount of human effort and the process can be faster than coders (since machines can work 24 hours per day). The disadvantage is that the observation accuracy is much worse than manual coding.

Figure 6.6 illustrates automatic observations. Our proposed video analysis methods are based on supervised learning, with the manual annotations providing training examples for training activity models. The "learned by" line in the fig-

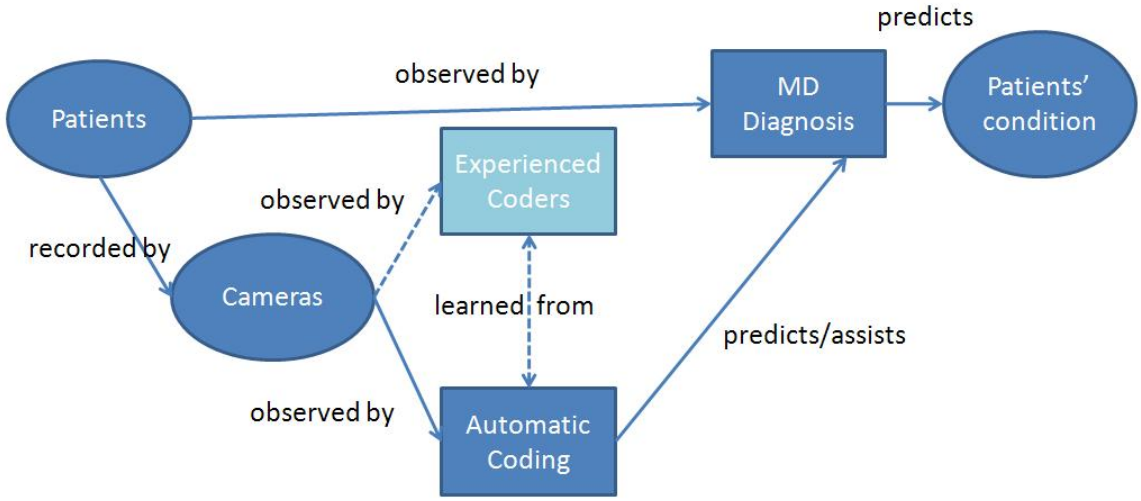

Figure 6.6: CareMeedia long term automatic observation diagram. Automatic video analysis algorithm trained from manual annotations detects and recognize interested activities in the video archive. The automatic coding data provides assist information for medical doctors to survey and helps them to make better diagnosis.

ure indicates that we train activity models from manual annotations. Even though we achieved good video activity analysis accuracy in the pervious chapters (Chapter 3, 5), the analysis performance is still far from perfect. The most important question then becomes: does automatic video analysis have good enough performance to provide informative observations which can be used to assist medical doctors in making a better diagnosis? We will answer this question in our experimental result section (chapter 6.3.2).

6.3 Experimental results

We have to design experiments to answer two questions. The first question is: does manual observation help predict patients' health conditions? The second question is: do current video analysis techniques have adequate performance for understanding the health of patients? Before discussing the experimental designs, we have to understand how to measure patients' health conditions. For senile dementia, medical diagnoses are provided with the aid of instruments that mea-

sure patients' health condition efficiently, such as the **Severe Impairment Battery (SIB)** [69], **Cohen-Mansfield Agitation Inventory-Community (CMAI-C)** [22], **Neuropsychiatric Inventory (NPI-NH)** [23], **Cornell Scale for Depression in Dementia (CSDD)** [5], **Physical Self-Maintenance Scale (PSMS)** [51], and **Cumulative Illness Rating Scale for Geriatrics (CIRS-G)** [61]. Each instrument is designed to evaluate an aspect of a patient's health condition. For example: SIB is designed to test cognitive impairments, PSMS evaluates ability to daily living activities, and CIRS-G is applied to measure medical burden.

In our experimental setting, we focus on predicting PSMS which is the most complete diagnostic instrument in our dataset. Since most residents in the nursing home have some level of senile dementia, some measurements are not completely evaluated on each resident during the observed month. PSMS is the only diagnosis in the database for which two evaluations were completed for 15 residents during the month. Appendix A shows all six categories and the score system of PSMS. Each PSMS activity is scored from 1 to 5. A score of 1 and 2 normally indicates that the patient is capable of doing the activity on their own with very minor help. A score of 3 normally means that the patient requires moderate assistance to perform the activity. A score of 4 and 5 regularly applies if the patient is not functional for the activity. The final PSMS score is the sum of all six activities and represents the ability to perform daily living activities.

Therefore, to answer both questions above, our experiments were designed to evaluate how well the manual observation and automatic observation predict PSMS scores. The manual observation can be treated as an oracle activity analysis which recognizes every activity during the period. The automatic observation is the automatic video analysis result which is predicted by the learned activity models. Given the 30 (15 residents x 2 times) PSMS diagnosis samples we have, it's unrealistic to predict the detailed scoring system of $1-5$. Therefore, we turn this diagnosis prediction to a binary classification problem which is learnable by machines. From PSMS scoring system, it is clear that a score of 3 is the dividing threshold. A resident who gets a score under 3 generally demonstrates his/her ability to finish the activity. A resident with a score above 3 (includes 3) normally is not capable of the activity. Therefore, we transfer the task of predicting PSMS diagnostic scores to predicting binary capability in each PSMS activity. In other

word, a diagnosis with score above 3 (includes) is labeled as positive (incapable to achieve the activity) and a diagnosis with score lower than 3 (capable to execute the activity) is annotated as negative. For the final PSMS score (sum of all six PSMS activities), we set up 18 (3x6) as the threshold.

We now obtain labels for the classification task by turning the PSMS scores to positive and negative labels. The next step is to transform manual observation (oracle video analysis) and automatic observation (automatic video analysis) into feature vectors to be able to train binary classifications. This classification task is to summarize a person's activities over time to predict his/her health condition which fits our second definition of a long term activity analysis. Unfortunately, we did not discover major health condition changes during the observed month from our diagnostic database. Therefore, we only focus on predicting patients' health conditions through summarization observed over time in this case study.

6.3.1 Oracle video analysis

The two PSMS diagnoses were collected in the middle and end of the recording month respectively. Therefore, a descriptor has to be generated to describe the manual observations within the two weeks before the end of the diagnostic evaluation. The descriptor is then the feature vector to train models which predict residents' capability in each PSMS activity. There are 12 codes in the movement activity category and there are 83 codes with 7 superordinate behavior codes in the detailed behavior category. Combining both categories, there are 95 codes. A histogram descriptor is generated by counting the frequency of each code within the 2 weeks. The descriptor is a 95 dimensional vector and each dimension indicates the frequency of one code. This descriptor summarizes the observed activities of a patient during the two weeks.

With labels (converted from PSMS scores) and feature vectors (histogram of manual codings), a SVM classifier with radial basis function is trained to predict the capability to perform PSMS activities. There, we need to first setup a baseline to compare with. The baseline is that we randomly guess the patient's capability to perform the PSMS activities which is 66.67% (measured by average precision). The baseline is higher than 50% because the residents in the nursing home all have

Category	Random	SVM	SVM-FS	Top feature
Toilet	66.67%	92.53%	91.53%	Staff activities: Feeding
Feeding	30.00%	50.00%	59.33%	Staff activities: Feeding
Dressing	73.33%	86.17%	96.08%	Standing Up
Grooming	76.67%	90.75%	90.75%	Standing Up
Ambulation	36.67%	57.33%	57.33%	Staff activities: Feeding
Bathing	83.33%	90.06%	98.33%	Positive: Others
PSMS	66.67%	92.53%	$\mathbf{94.20}\%$	Staff activities: Feeding

Table 6.1: Oracle detectors to predict the capability of PSMS activities. SVM classification has a solid performance and feature selection (SVM-FS) keeps boosting the performance. The top feature indicates the most discriminative feature among 95 coded activities. Manual observation is able to predict daily living capability of a resident 94.20% correctly.

some level of senile dementia. If you just randomly guess he/she isn't capable to execute PSMS activities, the chance you are correct is 66.56%. We call the manual observation as oracle video analysis setting because we assume all the observations coded by experienced coders are correct. This is the same as the situation where we would have a perfect video analysis system. Average precision is applied to measure the performance and leave one out cross validation is employed (take one resident out and train on the other 14 residents). The performance is shown in Table 6.1.

Surprisingly, a SVM classifier can predict the functionality of PSMS to 92.53% correct. Among each PSMS category, all SVM predictions outperform the random guesses significantly. This is a surprising result for two reasons: The first reason is that only approximately 20% of the CareMedia data was coded and the annotations are highly biased to meal times. The second reason is that 3 PSMS activities (Toilet, Dressing, and Bathing) aren't observed in any of the public areas. Grooming is also hard to evaluate given our coding strategy. However, despite the biased annotations and the non-specialized coding scheme, manual observations are still very informative to PSMS diagnoses. This is a solid indication that surveillance video can be an informative source to medical diagnoses.

We further explore feature selection on the proposed histogram observation feature. We apply the F-score to select features. F-score is a simple technique which measures the discrimination of a feature. Given training vectors x_k, $k =$

$1, ..., m$, if the number of positive and negative instances are n_+ and n_-, respectively, the then F-score of the ith feature is defined as:

$$F(i) = \frac{(\overline{x_i^+} - \overline{x_i})^2 + (\overline{x_i^-} - \overline{x_i})^2}{\frac{1}{n_+ - 1}\sum_{k=1}^{n_+}(x_{k,i}^+ - \overline{x_i^+})^2 + \frac{1}{n_- - 1}\sum_{k=1}^{n_-}(x_{k,i}^- - \overline{x_i^-})^2} \tag{6.1}$$

where $\overline{x_i}$, $\overline{x_i^+}$, and $\overline{x_i^-}$ are the average of the ith feature of the whole, positive, and negative data sets respectively; $x_{k,i}^+$ is the ith feature of the kth positive instance and $x_{k,i}^-$ is the ith feature of the kth negative instance. The numerator indicates the discrimination between the positive and negative sets, and the denominator indicates the one within each of the two sets. The larger the F-score is, the more likely this feature is more discriminative. Therefore, we use this score as feature selection criterion. We select features with high F-score and then apply SVM for training/predicting until we find a set of features which maximizes the performance. The feature selection further boosts performance to 94.20% and the most discriminative feature is listed in Table 6.1. From the result, it is obvious that "Activities: Feeding" provides a lot of information to predict PSMS functionalities since the annotation is biased to meal times. However, having the ability to eat during meal time can be interpreted as being more healthy in general and this further supports our hypothesis that observations from video recording can be a great aid to understanding patients' health conditions.

6.3.2 Simulated automatic video analysis

The manual observation is actually the ideal case, where we can assume that the activity analysis is perfect. Although it is impractical, the results shown in Table 6.1 can serve as a theoretical upper bound to indicate how useful activity analysis can be. To get a more realistic estimate (as opposed to the perfect "oracle" video analysis) of the activity analysis utility with the state-of-the-art activity analysis techniques, we repeated the experiments after introducing noise into the perfect activity analysis. The result from Table 3.4 and Table 3.5 show that the current activity recognition system can achieve 45% and 19% MAP in the movement activity and detailed behavior categories respectively. Because mean average precision is a rank-based measure and difficult to simulate, we approx-

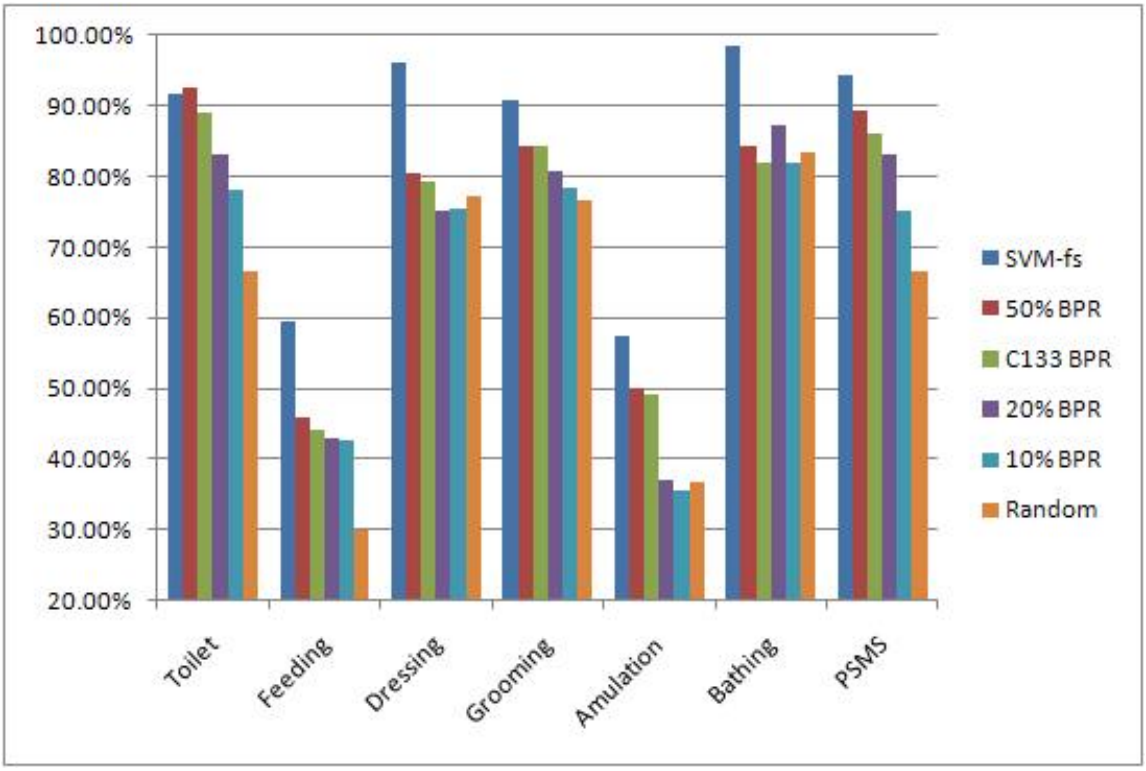

Figure 6.7: Simulated video analysis of predicting PSMS. SVM-fs serves as theoretical upper bound. The results using simulated "noisily" analysis are shown at 50%, 20%, and 10% breakeven precision recall (shown as "50% BPR", "20% BPR", and "10% BPR" respectively). C133 BPR indicates recognizers are simulated by recognition performance by Camera 133 which is the set from which we really built a activity recognition system.

imated this MAP with a breakeven precision-recall point at x where x is desired MAP. Breakeven precision-recall is usually a good approximation for mean average precision. They are equivalent to each other if the precision-recall curve is mirror symmetric to the line of precision=recall. This was easily achieved by randomly switching the labels of positively annotated activities to be (incorrectly) labeled as negative and conversely switching some negatively labeled activities as incorrect positive examples, until we achieve the desired breakeven point where precision is equal to recall. This made the activity labels appear roughly equivalent to a recognizer with MAP of x.

Figure 6.7 shows the performance of predicting the patients' capability to per-

form PSMS activities under different settings. SVM with feature selection on manual observations here serves as a theoretical upper bound. In a more realistic setting, we investigate the performance after introducing recognition noise into the video analysis results. In this case, the prediction performance keep decreasing as more and more noise is added. However, even when the breakeven precision-recall of these activity recognition results is only 10%, the prediction MAP can still be boosted to 13% better than random. Given current video analysis techniques, we extrapolate the performance obtained from Camera 133 (shown in Table 3.4 and Table 3.5) to simulate recognizers on the whole datase. The performance on predicting PSMS capability approaches to 86%. We believe at this level of accuracy automatic systems could provide helpful suggestions for diagnostic assistance. We doesn't show the performance of simulated results with feature selection because feature selection does not improve the noisy data. This experience suggests that although the video analysis provided by the state-of-the-art automatic video analysis algorithms is far from perfect, they still have the potential to augment traditional health care and improve medical diagnoses.

It is worth mentioning that all of the above discussion assume the video analysis is based on activity detection. However, the algorithms we apply to camera 133 is an activity recognition. In practice, activity recognition still outperforms activity detection due to the temporal segmentation issue. However, our experimental results still give a strong indication that weak activity detectors are potentially informative for medical diagnoses. Therefore, this study gives a solid evidence that automatic video analysis has real potential to assist long term health care.

6.4 Summary

In this chapter, we demonstrated the ability to extend video activity analysis to long term activity analysis. We use a case study, CareMedia, to demonstrate a way to analyze long term activities by video activity analysis techniques. In this case study, the long term activity analysis task is to analyze long term health care in nursing home environments. Although there are many aspects in health care, we focus on summarizing patients' behaviors over a period of time to predict their health condition. We successfully demonstrated that the manual observa-

tions from surveillance video are able to predict patients' capabilities of PSMS, a medical diagnosis. Furthermore, the automatic video observations obtained from our proposed video analysis techniques show promising potential to evaluate patient's health condition accurately over time. This long term health care analysis not only successfully validates the idea of the CareMedia project but also demonstrates a way to analyze long term activity from a video surveillance archive. Meanwhile, the experimental results show that even currently inaccurate video analysis techniques can still provide informative observations from video recording and have the capability to predict health conditions in our case study.

Chapter 7

Applications

There are many applications for robust video activity analysis. In this chapter, we demonstrate two applications in two important domains: interactive interface and intelligent surveillance video system. A gestural TV control system demonstrates a natural vision-based interactive interface to control a television set. A customer shopping behavior analysis system provides an intelligent surveillance video system. But before building these systems, we must solve an important problem: robust video activity analysis is computationally expensive.

MoSIFT demonstrates the ability to analyze video activities accurately. However, calculating SIFT and optical flow at multiple scales from every frame in a high-resolution stream is extremely expensive and slow. Fortunately, the increasing availability of large-scale computer clusters is driving efforts to parallelize video applications so that they can be mapped across a distributed infrastructure. The majority of these efforts, such as MapReduce [26] and Dryad [42], focus on efficient batch analysis of large data sets; while such systems accelerate the offline indexing of video content, they do not support continuous processing. A smaller set of systems provide support for the continuous processing of streaming data [1, 7, 21, 87] but most of these focus on queries using relational operators and data types, or are intended for mining applications in which throughput is optimized over latency.

In collaboration with Intel Labs Pittsburgh [41], we successfully parallelized the MoSIFT activity recognition framework on the Sprout [70]. Sprout is a distributed stream processing system designed to enable the creation of interactive

multimedia application. Interaction requires low end-to-end latency, typically well under 1 second [14, 16, 62]. Sprout achieves low latency by exploiting the coarse-grained parallelism inherent in such applications, executing parallel tasks on clusters of commodity multi-core servers. Its programming model facilitates the expression of application parallelism while hiding much of the complexity of parallel and distributed programming.

Therefore, we will first discuss how to implement parallelized MoSIFT activity recognition on the Sprout architecture. Then we will introduce two real world applications: a gestural TV control system and a customer behavior analysis application.

7.1 Parallel MoSIFT activity recognition

We implemented a parallel activity recognition application using MoSIFT features on the Sprout. Figure 7.1 shows the decomposition of the application into the Sprout stages. The implementation uses both coarse-grained parallelism at the stage level, and fine-grained parallelism within stages using OpenMP. This section describes our implementation and the methods used to parallelize its execution, following the processing order shown in Figure 7.1.

7.1.1 Frame pairs and tiling

Since MoSIFT computes optical flow, processing is based on frame pairs. A video data source decomposes the video into a series of overlapping frame pairs, which are input to the main processing stages. Since the MoSIFT interest points are local to regions of an image pair, we exploit intra-frame parallelization using an image tiler stage. The tiler divides each frame into a configurable number of uniformly sized overlapping sub-regions. The tiles are sent to a set of feature extraction stages to be processed in parallel. Overlap of the tiles ensures that interest points near the tile boundaries are correctly identified. The tiler also generates meta-data that includes positions and sizes of the tiles, for merging the results of feature extraction.

This tiling approach is an example of coarse-grained parallelization, since it

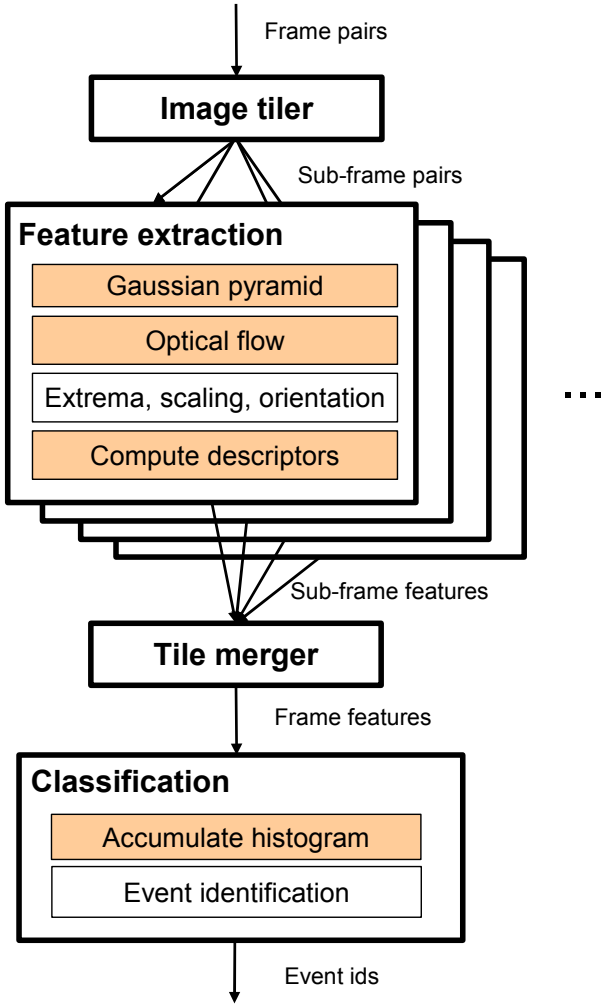

Figure 7.1: Sprout application graph for MoSIFT-based activity recognition. Coarse-grained (intra-frame) parallelism is exploited through tiling. Fine-grained parallelism is used within stages that implement the processing steps shown in shaded boxes.

does not require any changes to the inner workings of the feature extraction stage. The Sprout runtime and APIs make it easy to reconfigure applications to make use of such parallelization. As another example of coarse-grained parallelization, we also run parallel instances of the entire graph of stages in Figure 7.1, using a round-robin data splitter to distribute frame pairs to the parallel instances. This latter technique improves throughput only, while the tiling approach improves both throughput and latency.

7.1.2 Feature extraction

Four major stages are involved in the MoSIFT feature extraction process: Gaussian pyramid, Optical flow, Local extrema (interest point detection), and Compute descriptors. All stages other than the local extrema detection can be fine-grained parallelized.

In the Gaussian pyramid and optical stages, a Gaussian pyramid is applied to each image in the frame pair. These are computed in parallel in two separate threads (one thread for the first image and the other for the second image). The optical flow is then computed between corresponding frames in Gaussian pyramids. We parallelize this set of computations using OpenMP to assign loop invocations to a set of threads. As image size and computation time varies over the octaves, we do not parallelize by octave. Rather, we parallelize by interval, assigning computation for a particular interval index across all octaves to a single thread. This ensures a balanced load among the threads for the optical flow computations.

The local extrema stage detects MoSIFT interest point by detecting local extrema (minima/maxima) of the DoG images across adjacent scales. This step requires few computations and we do not employ parallelism in this stage. The final step of the feature extraction stage is the descriptor computation. Since interest points are independent, descriptors are computed in parallel over the interest points, limited only by the available cores on the processing node.

7.1.3 Tile merger and classification

After the feature descriptors are constructed, each feature extraction stage sends the descriptors to a tile merger stage, which collects the feature descriptors and

adjusts their positions in the whole frame. In the classification stage, features are mapped to codewords in a previously-generated camera-specific codebook. A histogram is generated for the current frame pair, and accumulated into histograms representing different time windows. The histogram is constructed in parallel over the features, up to the number of available cores. Finally, an SVM is used on normalized histograms to identify specific activities.

7.2 Real time gestural TV control system

Vision-based user interfaces enable natural interaction modalities such as gestures. Such interfaces require computationally intensive video processing at low latency. We demonstrate an application that recognizes gestures to control TV operations. Accurate recognition is achieved by MoSIFT, and video processing at low latency is again built by the Sprout. This application demonstrates our robust video analysis techniques which can be used in interactive applications.

Our application involves a situation where the television set is actively observing the viewers all the time. This enables any viewer to control a TV's operations, such as channel selection and volume, without additional devices such as remote controls, motion sensors or special clothing, simply by gesturing to the TV set. We define 6 gestures to control a TV, figure 7.2 shows a "channel up" gesture. The application is an implementation of a low-latency gesture recognition system that processes video from a commodity camera to identify complex gestures in real time and interpret them to control the TV set. While this application uses a commodity webcam, our proposed approach can be applied to video from depth-enhanced cameras that will soon become available. Such sensors offer increased resiliency to background clutter, and initial reports indicate that they are well suited for natural user interfaces [59].

Our application allows any user standing or sitting in front of a TV set to control its operations through gestures. The TV is equipped with a camera that observes the users watching the programs. When a user gives an "attention" signal by raising both arms, the control application then observes this user more carefully for a few seconds to recognize a control command. Examples of control commands can be hand and arm motion upward or outward, as well as crossing

Figure 7.2: User gesturing "Channel Up".

hands/arms. In the current interface, e.g., a left hand moving upwards indicates a channel should be switched up, and a left hand moving outwards signifies that the channel should be switched down. Analogously we use the right hand to control the volume of the audio. Crossing gestures are used to shut off the TV. User tests showed that downward motions cannot be effectively executed by seated users; therefore we avoided downward motions in the current gesture command set.

In this application, we highlight two aspects of our human-activity recognition research. First, we employ MoSIFT to recognize gestures accurately. Although computationally more expensive, this approach significantly outperforms state-of-the-art approaches on standard action recognition KTH data sets. These results validate our belief that MoSIFT is capable to analyze gestures or any further body languages to control devices.

Second, we utilize a cluster-based distributed runtime system that achieves low latency by exploiting the parallelism inherent in video understanding applications to run them in interactive time scales. In particular, although straight-

forward sequential implementations of MoSIFT can process relatively small collections of videos, such as the popular KTH dataset, they cannot process data at the speed required for the real-world applications that are the primary focus of our research. Our application implements the computationally challenging, but highly accurate MoSIFT descriptor on top of the Sprout runtime, and parallelizes execution across a cluster of several 8-core machines, to detect TV control gestures in full-frame-rate video with low latency.

Figure 7.3 illustrates our application data flow. Each video frame from a camera that observes the user is sent to two separate tasks, face detection and MoSIFT detection task. The incoming frame is duplicated (Copy stage) and sent to two different stages which initialize tasks. The face detection task starts from a scale stage (Scaler) which scales the frame to a desired size. The tiling stage (Tiler) is an example of coarse-grained parallelization. The tiler divides each frame into configurable number of uniformly sized overlapping sub-regions. The tiles are sent to a set of stages to be processed in parallel. The tiler also generates meta-data that includes positions and sizes of the tiles, for merging the results. The face detected in the scaled frame is de-scaled via Descaler stage to recover the resolution. The face detection result is then sent to the display stage to display and a classify stage which will further fuse the face detection result with MoSIFT features to detect gestures. The MoSIFT detection task accumulates frame pairs, and then extracts MoSIFT features that encode optical flow in addition to appearance. These features, filtered by the positions of detected faces, are aggregated over a window of frames to generate a histogram of their occurrence frequencies. The histogram is treated as an input vector to a set of support vector machines trained to detect gestures in video streaming. These processes are included in the Classify stage. The gesture detection result is further sent to the TV control stage to perform the associated TV controlling.

7.3 Shopping mall customer behavior analysis

We would like to demonstrate the suitability of our proposed activity detection method to real-world applications. Customer shopping behavior analysis is very important to retailers. Information about the popularity of a product is very valu-

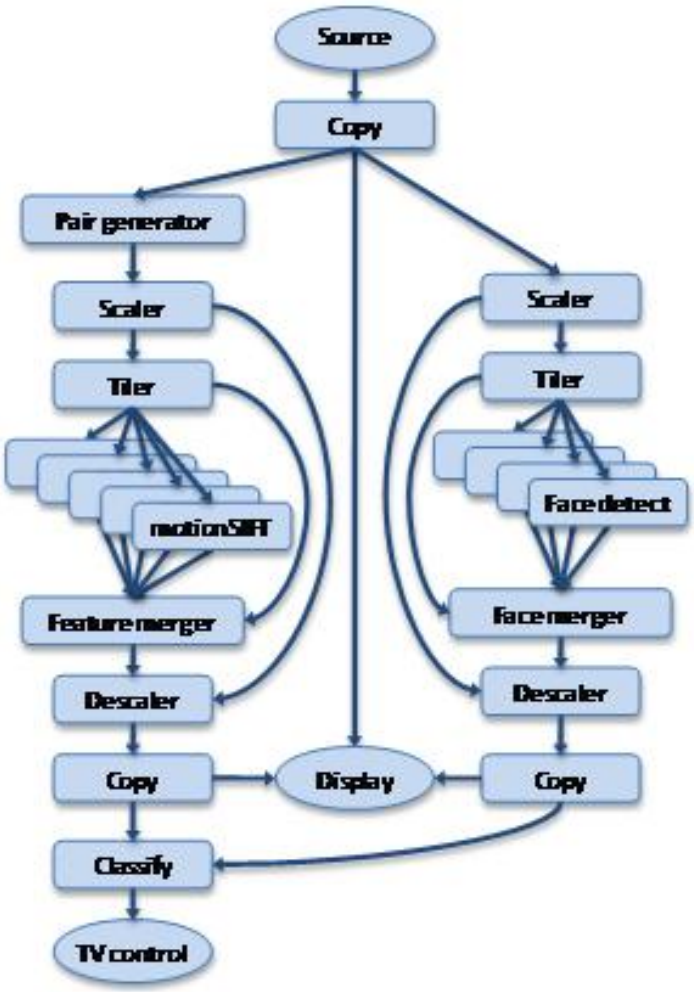

Figure 7.3: Application flow of the video gestural TV control system. The application includes face detection to specify face location, MoSIFT activity recognition to identify gestures, and TV control system to control a TV set. The system is constructed by Sprout and runs frull-frame-rate with low latency.

able to retailers and manufactures. Currently, many online retailers, e.g. Amazon.com, can simply apply machine learning techniques to understand customer's shopping behaviors through logs and click paths. However, similar analysis is very challenging for traditional stores because it is hard to monitor customers' behavior in the store.

Figure 7.4: A touching example in a shopping mall surveillance video. The red bounding box indicates a touching activity.

However, almost every store has a surveillance video system which records activities in the store. Originally, the surveillance systems were built for security desires. This valuable recording actually provides a dataset for customer behavior analysis. Our proposed application detects the ”touching” activity in the video. A touching activity is an action where a customer touches a product on a shelf. Touching can be either just purely touching or taking a product from a shelf. By detecting touching activities, we can calculate the fraction of customers who are interested in a product. Customers touch a product when they either are interested in that product or purchase that product. Both are valuable behaviors from customers. We applied the application on the NEC Shopping-Mall dataset. The NEC Shopping-Mall dataset is a surveillance video data collection from a supermarket in Japan. It has 2 calibrated cameras and contains 2 hours of recording. The recording was at 640x480 resolution and 30 fps MPEG-1 format. We preliminary evaluated the first hour for the touching activity detection.

We applied our activity detection algorithm with a people detection algorithm. The people detection [30] first detects people in the video and provides a rectan-

gular bounding box to apply the activity detection. Figure 7.4 shows that our system detects a touching activity in a crowded shopping mall. The performance to correctly detect a touching activity is at 69% precision and 61% recall. This performance gives a solid tool to analyze customers' touching behaviors in the store and, furthermore, the system can also be supported by the Sprout to run in real time.

7.4 Summary

To demonstrate the feasibility of video activity analysis, we successfully built two applications in two important domains, interactive interface and intelligent surveillance system. Furthermore, with help from Intel Labs Pittsburgh, we successfully parallelized the MoSIFT activity analysis framework on the Sprout architecture. This technique enables us to build MoSIFT applications to run at full frame rate with low latency. The interactive interface application we built is a system which recognizes human gestures to a television set. The intelligent surveillance application is a system which analyzes customers' shopping behaviors by detecting touching activities in a shopping store. These two applications show the great potential to extend video analysis and long term video analysis techniques to various domains. The applications also demonstrate that video activity analysis is sufficiently mature for real-world applications. Although video activity analysis is still a very tough computer vision and machine learning task, adopting current techniques to build practical applications is now possible.

Chapter 8

Conclusion

Long term activity analysis is an emerging research area in multimedia communities. In this thesis, we specifically focus on analyzing activities from surveillance video achieves. In order to analyze activities over a long period of time, there are several fundamental problems to address. First, a solid video feature must describe motions explicitly. Second, a robust activity recognition framework must identify the interesting activities. Third, a solid activity detection technique should specify when the interesting activity happens. Finally, long term activity analysis must be framed on a machine learning task. We consider our study on the CareMedia long term health care analysis as a case study of a long term activity analysis. Specifically, we study long term video activity analysis in nursing homes where the analysis can improve quality of care and quality of life of nursing home residents.

The motivation of this research comes from two phenomena that we observed. First, a large amount of surveillance video is recorded every day without processing. Second, observations over time provide a unique view to analyze data. Traditionally, video recording is mainly for security concerns. It is only used to review as evidence. However, many activities can actually be detected from surveillance videos to either prevent harm or understand human behaviors. Furthermore, surveillance video keeps recording day after day which is a valuable information source to understand human behavior over time. A long term behavior analysis is valuable, e.g. customer shopping behavior model, patients' behavioral changes, and traffic loads over time. All these observations inspire us to study long term

activity analysis of surveillance video archives.

In this work, we first study the two essential components for video analysis, activity recognition and detection through a powerful video feature descriptor, MoSIFT. We then perform a case study on the CareMedia data to demonstrate a way to analyze long term activity to help the nursing home health care.

8.1 Contributions

The first contribution of this study is to develop a framework of MoSIFT activity recognition. MoSIFT is a descriptor which explicitly describes both appearance and motion of a region of interest at multiple scales from a video. The activity recognition framework consists of interest point extraction, video codebook construction/mapping, bag-of-word feature representation, and modeling. The constraint-based clustering approach, bigram model and soft-weighting scheme are introduced to enhance the bag-of-word representation and further improve recognition performance. In developing this framework, we learnt several important concepts to build a robust activity recognition:

- Explicitly describing motions is critical in video feature descriptors.
- Instead of detecting interest points in temporal space with complex criteria, it is more important to detect what people can observe directly from a video.
- Dense descriptors are efficient and robust to build accurate activity models.
- The bag-of-word feature is an efficient and robust approach to represent interest points.
- Encoding relationships into the bag-of-word feature can substantially improves the recognition performance.
- The chi-square kernel of SVM performs strongly on modeling histogram features.

The second contribution comes from building an activity detection strategy. A brute-force search strategy is achieved by sliding a fixed length window over a video to generate candidate windows. A cascade SVM classifier is built to identify interesting activities among all the candidate windows. The false positive rate

is decreased by the good property of the cascade architecture and concatenating positive prediction strategy. From building this activity detection framework, we learned:

- Temporal segmentation is a subjective task and is not practical.
- The brute-force search strategy always generates too many negative examples and results in high false positive rates.
- The cascade architecture efficiently reduces false positive rates but maintains a high detection rate.
- The cascade architecture consumes negative examples very fast.

The third contribution comes from a successful case study to analyze long term activity from surveillance video in the nursing home health care domain. A long term activity analysis is domain dependent and there is no general way to solve this problem. The case study we proposed in the CareMedia project is to detect nursing home residents' daily lives over time to better estimate their health conditions. We demonstrate that the observations in surveillance video are informative by predicting patients' diagnoses from manual annotations. Furthermore, we successfully simulate automatic video analysis results and demonstrate that inaccurate video analysis can still assist medical doctors to make better diagnoses. This work as we know is the first to validate that video surveillance can assist health care by observing patients over a long period of time. It also demonstrates that multimedia techniques are now able to analyze information accurately if reasonable task is designed. By applying our method to long term health care analysis, we learned:

- Long term activity analysis is very domain specific. It requires domain knowledge to understand what information is needed.
- It is important to design a machine learnable approach to analyze the long term activity.
- Since automatic activity analysis is still not very accurate, it is important to first evaluate the ideal condition. For example, are the interesting activates sufficient for analyzing the desired long term pattern?
- The ideal condition can be achieved by manual observations.

- Simulations can provide a solid estimate of the automatic video analysis performance.
- Current video analysis techniques are bebining to provide helpful information but more fundamental computer vision and machine learning research is still needed.
- Sensors can definitely be a great auxiliary source to visual activity analysis and the long term activity analysis.

The fourth and last contribution is to demonstrate two video analysis applications. We successfully parallelize MoSIFT activity recognition by the Sprout architecture to achieve real time activity analysis. This technique enables us to build real-world applications. We demonstrate the proposed activity analysis techniques in two aspects: an interactive interface and an intelligent retail store surveillance system. The success in building real-world applications gives us the confidence that the proposed methods can be applied to many emerging areas, e.g. content-based video retrieval, traffic load analysis, tracking, day care surveillance systems etc. Given the exponential growth of video content, our proposed techniques can provide a tool to access video content efficiently. We learned several lessons when we build the applications:

- Coarse-grained and fine-grained parallelism are needed to improve the latency in video processing.
- Video activity analysis can be integrated with other techniques, e.g. face detections or sensors.
- A large number of human annotations are still required to train a robust activity model.

8.2 Future Work

There are many future research opportunities in long term activity analysis and the more general research area of video activity analysis. We categorize future research into four directions: low level video features, video activity analysis, long term activity analysis, and video content understanding.

MoSIFT is extended from SIFT and is proved to be a robust low level feature to describe video content. However, MoSIFT also inherits the weakness of SIFT. Interest points detected by MoSIFT emphasize high contrast points around corners or edges. Sometimes, it is not enough to describes activities. Also, camera motions cause motions all over a video which causes our algorithm to report bad results. However, camera motions are unavoidable in real-world videos. Due to the properties of MoSIFT, MoSIT is not sensitive to motions which are moving away from cameras. All these problems require further research toward making MoSIFT more robust.

In the area of video analysis, many interesting problems remaining for future work. First, the bag-of-word feature representation does not capture structure information. Although we proposed several methods to connect interest points, capturing global structure is still an on-going research direction. Our proposed recognition framework has very solid performance in many different domains. On the other hand, the proposed activity detection method can still be improved. To improve the proposed activity detection method, the most urgent topic is to segment the video more accurately to limit the search space. It may not be able to detect activity segments. However, predicting possible locations instead of the brute-search strategy could significantly decrease false positive rates.

Long term activity analysis requires much future research. The highest priority problem is to build a protocol which gives a guide line for transforming a domain specific long term analysis task to a machine learnable task. This is a challenging problem. It requires designing an application to analyze the domain dependent information need, constructing a system to observe the necessary information, developing a feature which represents the long term observations, and finally building a model to fill the information needs. Each step requires a language to facilitate communication between users and systems. Mixed-initiative learning [39] may be a good approach to construct communication between users and systems. Furthermore, senors provide more accurate information than video recording. Combining sensors with vision-based long term activity analysis is a emerging topic to explore.

Finally, we want to extend these video analysis techniques from surveillance video domain to the general video domain. Concept-based video content retrieval

is a promising direction in the video retrieval field. Here, MoSIFT is a solid and robust feature to detect semantic concepts. However, tremendous human effort would be required to annotate data in order to train a concept detector. Automatically associating images/video and text is a promising way to obtain robust annotations from the internet. This could open a new research domain for researchers to explore.

Appendix A

The PSMS coding manual

Table A.1: A full description of Physical Self-Maintenance Scale (PSMS).

Category	Description	Score
Toilet	Ability to care for self at toilet; ability to control bowels and bladder	1 = Cares for self at toilet completely, no incontinence 2 = Needs to be reminded or needs help in cleaning self or has rare accidents 3 = Soiling or wetting while asleep more than once a week 4 = Soiling or wetting while awake more than once a week 5 = No control of bowels or bladder
Feeding	Ability to feed self	1 = Eats without assistance 2 = Eats with minor assistance at meal time and/or with special preparation of food or help in cleaning up after meals 3 = Feeds self with moderate assistance 4 = Requires extensive assistance for all meals 5 = Does not feed self at all and resists efforts of others to feed him/her

Dressing	Ability to dress self	1 = Dresses, undresses, and selects clothing from own wardrobe 2 = Dresses and undresses self with minor assistance 3 = Needs moderate assistance in dressing or selection of clothes 4 = Needs major assistance in dressing but cooperates with efforts of others to help 5 = Completely unable to dress self and resists efforts
Grooming	Ability to groom self	1 = Always neatly dressed, well-groomed, without assistance 2 = Grooms self and adequately with occasional minor assistance, e.g. shaving 3 = Needs moderate and regular assistance or supervision in grooming 4 = Needs total grooming care but can remain well-groomed after help from others 5 = Actively negates all efforts of others to maintain grooming
Ambulation	Ability to ambulate within residence or outside residence	1 = Goes about grounds or city 2 = Ambulates within residence or about one block distance 3 = Ambulates with assistance 4 = Sits unsupported in chair or wheelchair but cannot propel self without help 5 = Bedridden more than half the time
Bathing	Ability to bathe or wash self	1 = Bathes self (tub, shower, sponge bath) without help 2 = Bathes self with help in getting in and out of tub

		3 = Washes face and hands only but cannot bathe rest of body 4 = Does not wash self but is cooperative with those who bathe him/her 5 = Does not try to wash self and resists efforts to keep him/her clean
Total	Sum of above 6 categories	range from 6-30

Appendix B

The CareMedia coding manual

Code	Activity
2001	Walking through
2002	Walking to a standing stop
2003	Standing up (the act of)
2004	Sitting down (the act of)
2005	Object placed on table
2006	Object removed from table
2007	Wheelchair movement
2008	Enters
2009	Exits
2010	Attempts to exit
2011	Communicates with staff
2011	Knocks on window

Table B.1: The coding manual of the movement activity category. The code is the key to save in the database. There are 12 activities in movement activity category in the coding manual.

Table B.2: The coding manual of the detail behavior category. The code is the key to save in the database. Major activity indicates superordinate behavior descriptions. Minor activity means subordinate behavior descriptions. There are 83 codes in this category by 7 superordinate behavior codes.

Code	Major activity	Minor activity
100	Pose and/or Motor Action	Assisted Action
101	Pose and/or Motor Action	Sleeping/Napping
102	Pose and/or Motor Action	Prone
103	Pose and/or Motor Action	Supine
104	Pose and/or Motor Action	Stooped
105	Pose and/or Motor Action	Facial dyskinesia
106	Pose and/or Motor Action	Tremors
107	Pose and/or Motor Action	Unsteady gait
108	Pose and/or Motor Action	Other motor behaviors
200	Positive	Smiles
201	Positive	Makes eye contact with person, object or activity
202	Positive	Socially appropriate touch, hug, kiss, holding hands
203	Positive	Dancing
204	Positive	Clapping pleasantly (e.g., to music)
205	Positive	Conversing pleasantly with others
206	Positive	Singing
207	Positive	Helping staff with their chores
208	Positive	Easily directed by staff in daily activities
209	Positive	Positive or affectionate verbal comments
210	Positive	Petting a real or stuffed animal or doll
211	Positive	Feeding or attempting to feed self
212	Positive	Other
300	Physically Aggressive	Spitting
301	Physically Aggressive	Grabbing

302	Physically Aggressive	Banging
303	Physically Aggressive	Pinching or squeezing
304	Physically Aggressive	Punching
305	Physically Aggressive	Elbowing
306	Physically Aggressive	Slapping
307	Physically Aggressive	Tackling
308	Physically Aggressive	Using object as weapon
309	Physically Aggressive	Taking from others
310	Physically Aggressive	Kicking
311	Physically Aggressive	Scratching
312	Physically Aggressive	Throwing
313	Physically Aggressive	Knocking over
314	Physically Aggressive	Pushing
315	Physically Aggressive	Pulling or tugging
316	Physically Aggressive	Biting
317	Physically Aggressive	Hurting self
318	Physically Aggressive	Obscene gestures
319	Physically Aggressive	Other
400	Physically Non-aggressive	Fidgeting/restless
401	Physically Non-aggressive	Pacing
402	Physically Non-aggressive	Wandering (lost)
403	Physically Non-aggressive	Exit seeking
404	Physically Non-aggressive	Picking
405	Physically Non-aggressive	Hoarding or hiding objects
406	Physically Non-aggressive	Unusual motor behaviors
407	Physically Non-aggressive	Eating or mouthing objects
408	Physically Non-aggressive	Interfering with others
409	Physically Non-aggressive	Urinating
410	Physically Non-aggressive	Defacating
411	Physically Non-aggressive	Eating
412	Physically Non-aggressive	Drinking

413	Physically Non-aggressive	Other
500	Verbally Aggressive	Scream/yell
501	Verbally Aggressive	Threatening or hostile comments
502	Verbally Aggressive	Argumentative
503	Verbally Aggressive	Name calling
504	Verbally Aggressive	Cursing
505	Verbally Aggressive	Other
600	Verbally Non-aggressive	Repeats self without obvious purpose
601	Verbally Non-aggressive	Nagging, pleading or calling for help
602	Verbally Non-aggressive	Refuses care, activities, food or medications
603	Verbally Non-aggressive	Bossy or demanding
604	Verbally Non-aggressive	Whiny or repetitive complaints
605	Verbally Non-aggressive	Talks to self
606	Verbally Non-aggressive	Sneezing
607	Verbally Non-aggressive	Coughing
608	Verbally Non-aggressive	Other
700	Staff Activities	Talking
701	Staff Activities	Feeding
702	Staff Activities	Getting food from cart
703	Staff Activities	Organizing, processing or dispensing medication
704	Staff Activities	Assisting a resident or another & Staff member
705	Staff Activities	Busing trays
706	Staff Activities	Vacuuming
707	Staff Activities	Mopping
708	Staff Activities	Writing or documenting care activities
709	Staff Activities	Redirecting a resident & Verbally or & Physically
710	Staff Activities	Other activity, non-patient related
711	Staff Activities	Other activity involving a patient

Appendix C

Experiment parameters

Experiment	Dataset	Codebook size	cost, gamma	Description
Table 3.1	KTH	900	8, 0.5	Leave-one-out cross validation
Table 3.2	Hollywood	1000	8, 1	Evaluate on test set
Table 3.3	Gatwick	2000	7, 4	5-folder cross validation by 5 days
Table 3.4	CareMedia	1000	8, 2	5-folder cross validation
Table 3.5	CareMedia	1000	8, 1	5-folder cross validation
Figure 4.1.3	KTH	600	8, 1	constraints with 2x2x5 window size
Table 4.1	KTH	900	8, 0.1	300 bigrams with 5x5x60 kernel size
Table 4.2	Gatwick	2000	1, 4	600 bigrams with 5x5x60 kernel size
Table 4.3	KTH	900	8, 0.5	4 closer clusters are soft-weighted
Table 4.4	Sound and Vision	2000	8, 2	4 closer clusters are soft-weighted
Table 5.2	Gatwick	2000	7, 4	cascade classifier

Table C.1: Parameters used in our experiments. Cost and gamma indicates two parameters in SVM kernel

Bibliography

[1] D. J. Abadi, Y. Ahmad, M. Balazinska, U. Çetintemel, M. Cherniack, J. Hwang, W. Lindner, A. S. Maskey, A. Rasin, E. Ryvkina, N. Tatbul, Y. Xing, and S. Zdonik. The design of the Borealis stream processing engine. In *Proc. Innovative Data Systems Research*, 2005. 7

[2] A. Adami, M. Pavel, T. Hayes, and C. Singer. Detection of movement in bed using unobtrusive load cell sensors. In *IEEE Transactions on Information Technology in Biomedicine*, 2009. 2.7

[3] A. Agarwal and B. Triggs. Hyperfeatures - multilevel local coding for visual recognition. In *ECCV*, 2006. 4.3.1

[4] K. Akita. Image sequence analysis of real world human motion. *Recognition*, 17(1), 1984. 2.1

[5] G. Alexopoulos, R. Abrams, R. Young, and C. Shamoian. Cornell scale for depression in dementia. In *Biol Psychiatry*, 1988. 1.5, 6.3

[6] S. J. Allin and E. Eckel. Machine perception for occupational therapy: Toward prediction of post-stroke functional scores in the home. In *Proceedings of the 29th Rehabilitation Engineering and Assistive Technology Society of North America (RESNA) Conference*, 2006. 6

[7] L. Amini, H. Andrade, R. Bhagwan, F. Eskesen, R. King, P. Selo, Y. Park, and C. Venkatramani. SPC: A distributed, scalable platform for data mining. In *Proc. Workshop on Data Mining Standards, Services, and Platforms*, 2006. 7

[8] B. Basu, M. Bilenko, and A. Banerjess. Probabilistic semi-supervised clustering with constraints. *Semi-Supervised Learning, MIT Press*, 2006. 4.1.2

[9] R. Bekkerman and J. Allan. Using bigrams in text categorization. *CIIR Tech-*

nical Report IR-408, 2004. 4.2

[10] A. Bharucha, H. Wactlar, S. Stevens, B. Pollock, M. Dew, D. Chen, and C. Atkeson. Caremedia: Automated video and sensor analysis for geriatric care. In *Proceedings of the Fifth Annual WPIC Research Day, University of Pittsburgh School of Medicine*, 2005. 6.1.1

[11] A. Bobick and J. Davis. The recognition of human movement using temporal templates. *IEEE Trans. PAMI*, 2001. 2.2

[12] J. Boger, P. Poupart, J. Hoey, C. Boutilier, G. Fernie, and A. Mihailidis. A decision-theoretic approach to task assistance for persons with dementia. In *IJCAI*, 2005. 2.4

[13] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Proc. Computational Learning Theory*, 1992. 3.3.3

[14] J. Brady. A theory of productivity in the creative process. *IEEE Computer Graphics and Applications*, 6(5), May 1986. 7

[15] C. Bregler. Learning and recognizing human dynamics in video sequences. In *CVPR*, 1997. 2.1

[16] S. K. Card, G. G. Robertson, and J. D. Mackinlay. The information visualizer, an information workspace. In *Proc. SIGCHI*, 1991. 7

[17] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`. 1.4, 5.2

[18] Y. J. C. D. Y. R. Chang, Y. People identification wth limited labels in privacy-protected video. In *International Conference on Multimedia and Expo (ICME'06)*, 2006. 6.2.1

[19] D. Chen, H. Wactlar, R. Malkin, and J. Yang. Detecting social interaction of elderly in a nursing home environment. In *ACM Transactions on Multimedia Computing, Communication and Application*, 2006. 6

[20] M.-y. Chen and A. Hauptmann. MoSIFT: Recognizing huamn actions in surveillance videos. Technical Report CMU-CS-09-161, Carnegie Mellon University, 2009. 5.1

[21] M. Cherniack, H. Balakrishnan, M. Balazinska, D. Carney, U. Çetintemel, Y. Xing, and S. Zdonik. Scalable distributed stream processing. In *Proc. Innovative Data Systems Research*, 2003. 7

[22] J. Cohen-Mansfield, M. Marx, and A. Rosenthal. A description of agitation in a nursing home. In *Journal of Gerontology*, 1989. 1.5, 6.3

[23] J. Cummings. Neuropsychiatric inventory. In *Nursing Home*, 1996. 1.5, 6.3

[24] N. Dala and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005. 2.2

[25] N. David, D. Doermann, L. David, and D. D. Mining tool for surveillance video. In *Proc. Storage and Retrieval Methods and Applications for Multimedia*, 2004. 2.4

[26] J. Dean and S. Ghemawat. MapReduce: simplified data processing on large clusters. *CACM*, 51(1), 2008. 7

[27] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *IEEE Workshop on PETS*, 2005. 2.4, 2.3, 3.1, 3.4.1, 3.4.1

[28] A. Fathi and G. Mori. Action recognition by learning mid-level motion features. In *CVPR*, 2008. 1.6.1

[29] A. Gilbert, J. Illingworth, and R. Bowden. Scale invariant action recognition using compound features mined from dense spatio-temporal corners. In *ECCV*, 2008. 1.6.1, 6

[30] M. Han, W. Xu, H. Tao, and Y. Gong. An algorithm for multple object trajectory tracking. In *CVPR*, 2004. 7.3

[31] C. Harris and M. Stephens. A combined corner and edge detector. In *Proc. Alvey Vision Conference*, 1988. 2.3, 3.1

[32] A. Hauptmann, M. Christel, and R. Yan. Video retrieval based on semantic concepts. In *Proceedings of the IEEE 96*, 2008. 2.5

[33] A. Hauptmann, H. Wactlar, J. Yang, Y. Qi, R. Yan, and J. Gao. Automated analysis of nursing home observations. In *IEEE Pervasive Computing, Special Issue on Pervasive Computing for Successful Aging*, 2004. 6.2.2

[34] A. Hauptmann, R. Yan, and W. Lin. How many high-level concepts will fill the semantic gap in news video retrieval? In *CIVR*, 2007. 1.6.4, 2.5

[35] A. Hauptmann, R. Yan, W.-H. Lin, M. Christel, and H. Wactlar. Can high level concepts fill the semantic gap in video retrieval? a case study with broadcast news. In *IEEE Transactions on Multimedia*, 2007. 2.5

[36] T. Hayes, S. Hagler, D. Austin, J. Kaye, and M. Pavel. Unobtrusive assessment of walking speed in the home using inexpensive pir sensors. In *31th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2009. 6.1.2

[37] T. Hayes, M. Pavel, and J. Kaye. An approach for deriving continuous health assessment using in-home sensors. In *Festival of International Conferences on Caregiving, Disability, Aging and Technology*, 2007. 6.1.2

[38] D. Hogg. Model-based vision: a program to see a walking person. *Image and Vision Computing*, 1(1), 1983. 2.1

[39] Y. Huang and T. Mitchell. Framework for mixed-initiative clustering. In *NESCAI*, 2007. 8.2

[40] A. Inoue, S. Hao, T. Saito, K. Shinoda, I. Kim, and C. Lee. Titgt at trecvid 2009 workshop. In *Proc. TRECVID Workshop*, 2009. 2.5

[41] Intel Labs Pittsburgh. `http://www.pittsburgh.intel-research.net/`. 7

[42] M. Isard, M. Budiu, Y. Yu, A. Birrell, and D. Fetterly. Dryad: distributed data-parallel programs from sequential building blocks. In *European Conference on Computer Systems*, 2007. 7

[43] H. Jhuang, T. Serre, L. Wolf, and T. Poggio. A biologically inspired system for action recognition. In *ICCV*, 2007. 1.6.1

[44] H. Jhuang, T. Serre, L. Wolf, and T. Poggio. A biologically inspired system for action recognition. In *ICCV*, 2007. 6

[45] Y. Ke, R. Sukthankar, and M. Hebert. Efficient visual event detection using volumetric features. In *ICCV*, 2005. 2.4, 2.3, 3.4.1, 3.4.1

[46] Y. Ke, R. Sukthankar, and M. Hebert. Event detection in crowded videos. In

ICCV, 2007. 2.3

[47] A. Kläser, M. Marszałek, and C. Schmid. A spatio-temporal descriptor based on 3D-gradients. In *BMVC*, 2008. 1.6.1

[48] J. Kleinberg. Bursty and hierarchical structure in streams. In *KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2002. 6

[49] I. Laptev and T. Lindeberg. Space-time interest points. In *ICCV*, 2003. 2.3, 3.1

[50] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*, 2008. 1.6, 1.6.1, 1.5, 3.2, 3.4.1, 3.4.2, 3.4.2, 3.4.3, 3.4.3

[51] M. Lawton and E. Brody. Assessment of older people: Self-maintaining and instrumental activities of daily living. In *Gerontologist*, 1969. 1.5, 6.3

[52] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognitizing natural scene categories. In *CVPR*, 2006. 4.3

[53] J. Liu, S. Ali, and M. Shah. Recognizing human actions using multiple features. In *CVPR*, 2008. 6

[54] J. Liu and M. Shah. Learning human actions via information maximization. In *CVPR*, 2008. 1.6.1, 6

[55] D. Lowe. Distinctive image features form scale-invariant keypoints. *IJCV*, 60(2), 2004. 2.3, 3, 3.1, 3.2, 3.3, 3.4

[56] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. *Proc. the 7th International Joint Conference on Artificial Intelligence*, 1981. 3.1.2

[57] A. MAdami, T. Hayes, M. Pavel, and C. Singer. Detection and classification of movements in bed using load cells. In *27th Annual International Conference of the IEEE Engineering In Medicine And Biology Society (EMBS)*, 2005. 6.1.2

[58] Y. Michael, E. McGregor, J. Allen, and S. Fickas. Observing outdoor activity using global positioning system-enabled cell phones. In *International Con-*

ference on Smart Homes and Health Telematics (ICOST), 2008. 2.7

[59] Microsoft, Project Natal in detail. `http://www.xbox.com/en-GB/news-features/news/Project-Natal-in-detail-050609.htm`. 7.2

[60] K. Mikolajczyk and H. Uemura. Action recognition with motion-appearance vocabulary forest. In *CVPR*, 2008. 1.6.1, 6

[61] M. Miller, C. Paradis, P. Houck, S. Mazumdar, J. Stack, A. Rifai, B. Mulsant, and C. Reynolds. Rating chronic medical illness burden in geropsychiatric practice and research: application of the cumulative illness rating scale. In *Psychiatry Res.*, 1992. 1.5, 6.3

[62] R. B. Miller. Response time in man-computer conversational transactions. In *Proc. AFIPS*, 1968. 7

[63] G. Moak and S. Borson. Mental health services in long-term care. In *American Journal of Geriatric Psychiatry*, 2000. 6.1

[64] J. C. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. In *BMVC*, 2006. 1.6.1, 3.4.1, 3.4.1

[65] National insititude of standards and technology. `http://www.nist.gov/index.html`. 1.6.3

[66] E. Nowak, F. Jurie, and B. Triggs. Sampling strategies for bag-of-features image classification. In *ECCV*, 2006. 4.3

[67] S. Nowozin, G. Bakir, and K. Tsuda. Discriminative subsequence mining for action classification. In *ICCV*, 2007. 1.6.1

[68] U. G. A. Office. Nursing homes: Prevalence of serious quality problems remains unacceptably high, despite some decline. *Washington, D.C.: U.S. General Accounting Office*, 2003. 6.1.1

[69] M. Panisset, M. Roudier, J. Saxton, and F. Boller. Bursty and hierarchical structure in streams. In *Archives of Neurology*, 1994. 1.5, 6.3

[70] P. Pillai, L. Mummert, S. Schlosser, R. Sukthankar, and C. Helfrich. SLIPStream: scalable low-latency interactive perception on streaming data. In

Proc. NOSSDAV, 2009. 1.3, 7

[71] R. Polana and R. Nelson. Low level recognition of human motion (or how to get your man without finding his body parts). In *Proc. IEEE Computer Society Workshop on Motion of Non-Rigid and Articulated Objects*, 1994. 2.2

[72] M. Rodriguez, J. Ahmed, and M. Shah. ActionMACH: a spatio-temporal maximum average correlation height filter for action recognition. In *CVPR*, 2008. 1.6.1, 6

[73] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. In *IEEE Trans. PAMI*, 1998. 5

[74] S. Savarese, A. Pozo, J. Niebles, and L. F.-F. Spatial-temporal correlations for unsupervised action classification. In *Proc. IEEE Workshop on Motion and Video Computing*, 2008. 4.2.1

[75] S. Savarese, J. Winn, and A. Griminisi. Discriminative object class models of apperance and shape by correlations. In *CVPR*, 2006. 4.2.1

[76] K. Schindler and L. Van Gool. Action snippets: How many frames does human action recognition require? In *CVPR*, 2008. 1.6.1, 3.2

[77] H. Schneiderman and T. Kanade. A statistical model for 3d object detection applied to faces and cars. In *CVPR*, 2000. 5

[78] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local SVM approach. In *ICPR*, 2004. 1.6, 1.6.1, 1.4, 3.4.1, 3.4.1

[79] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *ICCV*, 2003. 4.3

[80] D. Smith. Detecting and browsing events in unstructured text. In *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, 2002. 6

[81] C. Snoek, K. Sande, O. Rooij, B. Huurnink, J. Uijlings, M. Liempt, M. Bugalho, I. Trancoso, F. Yan, M. Tahir, K. Mikolajczyk, J. Kittler, M. Rijke, J. Geusebroek, T. Gevers, M. Worring, A. Smeulders, and D. Koelma. The mediamill trecvid 2009 semantic video search engine. In *Proc. TRECVID Workshop*, 2009. 2.5

[82] S. Stevens, D. Chen, H. Wactlar, A. Hauptmann, M. Christel, and A. Bharucha. Automatic collection, analysis, access and archiving of psycho/social behavior by individuals and groups. In *Capture, Archival and Retrieval of Personal Experiences (CARPE'06)*, 2006. 1.6

[83] X. Sun, M.-Y. Chen, , and A. Hauptmann. Action recognition via local descriptors and holistic features. In *CVPR*, 2009. 1.6.1

[84] K. Sung and T. Poggio. Example-based learning for view-based human face detection. In *IEEE Trans. PAMI*, 1998. 5

[85] TRECVID 2008. `http://www-nlpir.nist.gov/projects/tv2008/tv2008.html`. 1.6, 1.6.3, 3.4.3, 5.3

[86] TRECVID 2009. `http://www-nlpir.nist.gov/projects/tv2009/tv2009.html`. 1.3, 1.6, 1.6.4, 3.4.3

[87] D. S. Turaga, B. Foo, O. Verscheure, and R. Yan. Configuring topologies of distributed semantic concept classifiers for continuous multimedia stream processing. In *ACM Multimedia*, 2008. 7

[88] D. Unay. Augmenting clinical observations with visual fatures from longitudinal mri data for improved demntia diagnosis. In *ACM International Conference on Multimedia Information Retrieval*, 2010. 2.7

[89] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR*, 2001. 5, 5.2

[90] H. Wactlar, A. Bharucha, S. Stevens, A. Hauptmann, and M. Christel. A system of video information capture, indexing and retrieval for interpreting human activity. In *Proc. IEEE International Symposium on Image and Signal Processing and Analysis*, 2003. 1.3, 1.6, 6

[91] L. Wang and D. Suter. Learning and matching of dynamic shape manifolds for human action recognition. In *IEEE Transactions on Image Processing*, 2007. 6

[92] G. Willems, T. Tuytelaars, and L. Van Gool. An efficient dense and scale-invariant spatio-temporal interest point detector. In *ECCV*, 2008. 1.6.1

[93] S.-F. Wong and R. Cipolla. Extracting spatiotemporal interest points using global information. In *ICCV*, 2007. 1.6.1, 3.4.1, 3.4.1

[94] J. Yamato, J. Ohya, , and K. Ishii. Recognizing human action in time-sequential images using hiden markov model. In *CVPR*, 1992. 2.1

[95] R. Yan and A. Hauptmann. Automatically labeling data using using multi-class active learning. In *ICCV*, 2003. 6.2.1

[96] M. Yang, F. Lv, W. Xu, and Y. Gong. Human action detection by boosting efficient motion features. In *IEEE Workshop on Video-oriented Object and Event Classification in Conjunction with ICCV*, 2009. 2.5, 2.6

[97] X. Yang, Y. Xu, R. Zhang, E. Chen, Q. Yan, B. Xiao, Z. Yu, Z. Ning Li, N. Huang, C. Zhang, X. Chen, A. Liu, Z. Chu, K. Guo, and J. Huang. Shanghai jiao tong university participation in high-level feature extraction and surveillance event detection at trecvid 2009. In *TRECVID workshop*, 2009. 2.6

[98] K. Yokoi, T. Watanabe, and S. Ito. Toshiba at trecvid 2009: Surveillance event detection task. In *TRECVID workshop*, 2009. 2.6

[99] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *IJCV*, 73(2), 2007. 3.3.3, 4.3

[100] G. Zhu, M. Yang, K. Yu, W. Xu, and Y. Gong. Detecting video events based on action recognition in complex scenes using spatio-temporal descriptor. In *ACM Multimedia*, 2009. 2.6