HOLISTIC REPRESENTATIONS FOR ACTIVITIES AND CROWD BEHAVIORS

by

BERKAN SOLMAZ
B.S. Middle East Technical University, 2005
M.S. Middle East Technical University, 2008

A dissertation submitted in partial fulfilment of the requirements
for the degree of Doctor of Philosophy
in the Department of Electrical Engineering and Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Summer Term
2013

Major Professor: Mubarak Shah

ABSTRACT

In this dissertation, we address the problem of analyzing the activities of people in a variety of scenarios, this is commonly encountered in vision applications. The overarching goal is to devise new representations for the activities, in settings where individuals or a number of people may take a part in specific activities. Different types of activities can be performed by either an individual at the fine level or by several people constituting a crowd at the coarse level. We take into account the domain specific information for modeling these activities. The summary of the proposed solutions is presented in the following.

The holistic description of videos is appealing for visual detection and classification tasks for several reasons including capturing the spatial relations between the scene components, simplicity, and performance [1, 2, 3]. First, we present a holistic (global) frequency spectrum based descriptor for representing the atomic actions performed by individuals such as: bench pressing, diving, hand waving, boxing, playing guitar, mixing, jumping, horse riding, hula hooping etc. We model and learn these individual actions for classifying complex user uploaded videos. Our method bypasses the detection of interest points, the extraction of local video descriptors and the quantization of local descriptors into a code book; it represents each video sequence as a single feature vector. This holistic feature vector is computed by applying a bank of 3-D spatio-temporal filters on the frequency spectrum of a video sequence; hence it integrates the information about the motion and scene structure. We tested our approach on two of the most challenging datasets, UCF50 [4] and HMDB51 [5], and obtained promising results which demonstrates the robustness and the discriminative power of our holistic video descriptor for classifying videos of various realistic actions.

In the above approach, a holistic feature vector of a video clip is acquired by dividing the video into spatio-temporal blocks then concatenating the features of the individual blocks together. However, such a holistic representation blindly incorporates all the video regions regardless of

their contribution in classification. Next, we present an approach which improves the performance of the holistic descriptors for activity recognition. In our novel method, we improve the holistic descriptors by discovering the discriminative video blocks. We measure the discriminativity of a block by examining its response to a pre-learned support vector machine model. In particular, a block is considered discriminative if it responds positively for positive training samples, and negatively for negative training samples. We pose the problem of finding the optimal blocks as a problem of selecting a sparse set of blocks, which maximizes the total classifier discriminativity. Through a detailed set of experiments on benchmark datasets [6, 7, 8, 9, 5, 10], we show that our method discovers the useful regions in the videos and eliminates the ones which are confusing for classification, which results in significant performance improvement over the state-of-the-art.

In contrast to the scenes where an individual performs a primitive action, there may be scenes with several people, where crowd behaviors may take place. For these types of scenes the traditional approaches for recognition will not work due to severe occlusion and computational requirements. The number of videos is limited and the scenes are complicated, hence learning these behaviors is not feasible. For this problem, we present a novel approach, based on the optical flow in a video sequence, for identifying five specific and common crowd behaviors in visual scenes. In the algorithm, the scene is overlaid by a grid of particles, initializing a dynamical system which is derived from the optical flow. Numerical integration of the optical flow provides particle trajectories that represent the motion in the scene. Linearization of the dynamical system allows a simple and practical analysis and classification of the behavior through the Jacobian matrix. Essentially, the eigenvalues of this matrix are used to determine the dynamic stability of points in the flow and each type of stability corresponds to one of the five crowd behaviors. The identified crowd behaviors are (1) bottlenecks: where many pedestrians/vehicles from various points in the scene are entering through one narrow passage, (2) fountainheads: where many pedestrians/vehicles are emerging from a narrow passage only to separate in many directions, (3) lanes: where many pedestrians/vehicles are moving at the same speeds in the same direction, (4) arches or rings: where the

collective motion is curved or circular, and (5) blocking: where there is a opposing motion and desired movement of groups of pedestrians is somehow prohibited. The implementation requires identifying a region of interest in the scene, and checking the eigenvalues of the Jacobian matrix in that region to determine the type of flow, that corresponds to various well-defined crowd behaviors. The eigenvalues are only considered in these regions of interest, consistent with the linear approximation and the implied behaviors. Since changes in eigenvalues can mean changes in stability, corresponding to changes in behavior, we can repeat the algorithm over clips of long video sequences to locate changes in behavior. This method was tested on over real videos representing crowd and traffic scenes.

To my family.

ACKNOWLEDGMENTS

I would like to thank Dr. Shah for his guidance and support. His constant motivation and faith in my ability was a very important factor for the completion of this work. I am grateful to Dr. Brian Moore, for his guidance during my research. I would also like to thank my friends and colleagues: Shayan Modiri Assari, Dr. Ramin Mehran, Dr. Omar Oreifej, Dr. Kishore Reddy, Dr. Vladimir Reilly, Dr. Imran Saleemi, Yang Yang, and others for their support. I am also grateful to Dr. Niels da Vitoria Lobo, Dr. Brian Moore, Dr. Sumit K. Jha, and Dr. Marcel Ilie for serving on my committee.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

One of the most fundamental goals of computer vision is to make computers perceive the objects present in the scene and to discover the ongoing actions and events. The human vision system is capable of performing these computationally complicated tasks in various domains and environments. The computer vision researchers have been designing and developing tools which can detect and recognize the objects and analyze their activities to be used in many applications involving the understanding of human activities.

Analyzing people and the activities they participate in is a very important problem in computer vision research. The variations in scene settings such as the number of objects or people, motion, scene structure, occlusion and frame resolution make this problem very challenging. We can divide the approaches for activity analysis into two broad groups; local spatio-temporal interest point based approaches and the holistic(global) approaches. The local approaches [2, 11, 12, 13, 14] describe each video (i.e. a sequence of image frames) by the computed local descriptors around the spatio-temporal interest points. These approaches are popular for vision applications due to their high performance and ability to deal with occlusion. On the other hand, they require the detection of interest points or quantization. Their limitation is the loss of geometrical and temporal information. On the other hand, the holistic approaches such as [1, 2, 3, 15] have recently received a lot of attention for visual detection and classification tasks since they have a simpler structure which inherently captures the spatial and temporal relations between the regions and they do not require the interest point detection or segmentation of foreground and background.

The majority of the previously designed methods for activity recognition are generic and they do not take into account the domain specific knowledge. They may not be directly applicable to the scenarios where the types of events may differ. For example, in a crowd video the movement of people is restricted and we expect to see higher level of activities when compared to a video with single person performing an action. This dissertation develops algorithms and holistic repre-

sentations that provide frameworks to model and label the activities involving both individuals and crowds in the videos to solve a wide range of common problems in the field of computer vision. The traditional pipeline of computer vision research, utilizing object detection, tracking, and behavior analysis may not be most feasible solution for analyzing the actions of people which still remains an open problem due to the inherent complexity and the diversity in the scenes.

1.1 Recognizing Actions of Individuals

The first problem this dissertation addresses is the recognition of simple actions of individuals, which has applications such as human-computer interaction, monitoring of patients and elder people, video retrieval and classification. Due to the massive number of videos uploaded online each day, several approaches have been proposed in the literature [16, 12, 13, 2, 17, 18] for the recognition of individual actions in diverse real-world videos. Computing descriptors for videos is one of the crucial tasks for action recognition on the grounds that the similarity of two videos can be measured by comparing their corresponding descriptors. In this dissertation, we present a holistic video descriptor (GIST3D) for the classification of web videos. Our holistic descriptor is computed by applying a bank of 3-D spatio-temporal filters on the spatio-temporal frequency spectrum of a video sequence; hence it integrates the information about the motion and scene structure. We tested our approach on three public datasets; KTH [6], UCF50 [4] and HMDB51 [5] and also on TRECVID 2011 event collection [19].

1.2 Discriminative Blocks of Holistic Descriptors

Second, we present a novel method which further improves the performance of the holistic approaches to be used for visual activity recognition. A holistic representation for a video clip can be obtained by dividing the video into spatio-temporal blocks and computing feature vectors for individual blocks or by applying a bank of filters on the videos, and finally concatenating the

features of the individual blocks together. However, holistic representations incorporate all of the video regions without taking into account their actual contribution in the classification. Regarding this problem, our new method improves the performances of holistic descriptors for the application of activity recognition by selecting the highly discriminative blocks of videos automatically and then extracting the descriptors on the selected blocks.

1.3 Recognizing Crowd Behaviors

When a large number of people gather together, individuals cannot move freely due to the lack of room for movement and the people in the scene may be involved in specific crowd behaviors. The increase in the number of objects or people in the scene results in severe occlusion and clutter which makes the detection and tracking of individuals and the recognition of primitive actions impractical; according to our observations even human visual system faces considerable difficulties in identifying the actions. Because of these challenges the traditional approaches would fail in identifying the behaviors in crowds. Automated detection of crowd behaviors has numerous applications related to video surveillance, such as prediction of congestion, which may help avoid unnecessary crowding or clogging, and discovery of abnormal behaviors or flow, which may help avoid tragic incidents. One goal of this particular work is to devise a novel holistic approach that identifies five common and specific crowd behaviors. These behaviors are called: bottlenecks, fountainheads, lanes, arches, and blocking. In this work, we will present a method that combines the low-level local motion features, computed by optical flow, with the high-level information of the scene, obtained by analyzing the trajectories and the regions of interest in the scene. Relying on the use of Lagrangian particle dynamics model [20, 21] of crowd scenes, the crowds are treated as collections of mutually interacting particles, hence our method is well-suited for small and large crowds [22]. Our method performs well in various types of crowd scenes as it does not involve object detection or tracking, which may be unreliable for crowd scenes. Also, our method does not

require training, as needed in most current approaches for behavior recognition. Our approach is not restricted to isolated activities, but is able to identify the coherent behaviors in the scene.

1.4 Summary of Contributions

The local video descriptors developed in computer vision research [16, 12, 13, 2] mainly utilize a bag-of-features representation discarding the spatial and temporal distribution of the local descriptors. The temporal information is important in recognizing an action. For example, temporal distribution helps distinguishing between actions such as opening and closing or pushing and pulling since they have similar types of motion but in reverse temporal orders. Similarly, the scene and context information plays an important role in classifying unconstrained web videos with a large number of action classes since the scenes and the ongoing actions are related. One of the contributions in this dissertation is to introduce a new holistic motion and scene descriptor for the classification of individual actions in realistic videos. Preserving the useful spatial and temporal information and without the need for interest point detection, background subtraction or tracking, we will represent each video with a single feature vector, and obtain the highest classification accuracies on two of the most complex datasets, UCF50 and HMDB51. Moreover, the combination of these local and holistic descriptors resulted in a further improvement in the results.

For further improvement of holistic descriptors, we will present a new approach which identifies the discriminative blocks and computes descriptors on the selected blocks. The selection of these blocks is done based on the contribution of each block on the inter-class and average discriminativity measures. Our method significantly improves the performance of descriptors such as histograms of oriented gradients (HOG3D) [2], GIST3D [23] and Action Bank [15] over all benchmark datasets. Furthermore, the computation and memory requirements are less since most of the regions are not used for feature extraction. In addition, we tested the use of HOG3D on the spatio-temporal frequency spectrum and observed an additional performance improvement over the

space-time gradient based descriptor. This presented descriptor has advantages such as translation and intensity invariance.

For high-density crowd scenes, we presented an approach to identify five specific crowd behaviors. When compared to other methods of crowd analysis, our method is the only one which is able to detect and identify particular crowd behaviors. None of the other studies connected flow fields with crowd behaviors. In summary, the four contributions of our approach are: (1) considering a linearization of the dynamical system, which is defined by the optical flow, we use several properties of the Jacobian matrix to explain the type of flows in regions of interest. (2) Using local analysis based on the Jacobian matrix and particle advection we are able to detect specific holistic behaviors i.e. bottlenecks, fountainheads, lanes, arches, and blocking. (3) Our method is simple, yet effective, and does not require object detection, tracking or any training to detect these behaviors. (4) Developing a modular framework, we can identify multiple crowd behaviors in one scene.

1.5 Organization of the Dissertation

The dissertation is structured as follows: Chapter 2 reviews existing literature, and discusses different approaches and applications for describing individual actions and analyzing crowds in videos in many problem domains. In Chapter 3, we present a holistic scene and motion descriptor to model actions of individuals. Chapter 4 describes our novel method for improving the classification performance of holistic descriptors and reports results on the benchmark datasets. Chapter 5 presents our framework to identify specific crowd behaviors through stability analysis for dynamical systems, which may happen when there is a high density crowd in a surveillance scenario. We will also present quantitative results on the videos collected from online sources. Finally, the dissertation is concluded in Chapter 6 with a summary of the contributions and the description of future work.

CHAPTER 2: LITERATURE REVIEW

In this chapter we will review a number of relevant methodologies in the literature related to visual recognition tasks such as analysis of individual actions and crowds in the videos. We will describe the advantages and drawbacks of these approaches in the following two sections.

2.1 Analysis of Individual Actions

Recognizing actions of individuals is one of the most popular topics in computer vision. In the literature, several approaches have been proposed for the recognition of individual actions in simple scene settings and also in diverse real-world videos. The survey papers [24, 25] provide a detailed overview of the present approaches, challenges and the available datasets.

Several datasets were presented in the literature. KTH [6] is the most popular benchmark dataset for action recognition and includes 6 action classes. This dataset was captured in a controlled environment and lacked the realistic scenarios such as variations in scale, view point, and frame rate. Laptev et al. [11], addressing the demand for datasets of large number of realistic action classes, introduced Hollywood dataset with eight action classes, collected from the movies. Liu et al. [17] introduced the UCF YouTube dataset consisting of 11 categories of actions collected from YouTube and personal videos. UCF Sports Actions dataset [8] includes videos of 10 categories of sports actions. UCF50 [9] extended the 11 action categories of the UCF YouTube dataset to a total of 50 action categories. As an extension to UCF50, later, UCF101 dataset [10] was announced which consists of videos of 101 action classes. Kuehne et al. [5] presented the very challenging HMDB51 dataset of 51 action classes with video clips collected from a wide range of sources. The variations in the recording settings and inter-personal differences make these datasets very complex. Provided these datasets, several approaches for action recognition that fall into two broad groups were proposed: holistic methods and local spatio-temporal interest point based methods.

Holistic methods represent the actions based on global information about the action and scene. Klaser et al. [2] represented videos based on histograms of oriented 3-D spatio-temporal gradients. Sadanand et al. [15] used a bank of action detectors, where each action detector is a template which is used to filter the video clip and generate a correlation volume. Then, max-pooling is employed using three levels in the octree. Oliva et al. [3] presented a holistic descriptor for images which is computed by the application of a set of filters in frequency domain. Holistic models perform well for the complex scenes.

Spatio-temporal interest point-based methods represent the scene and the performed actions as a combination of local descriptors, which are computed in a neighborhood of interest points. The neighborhood can be selected as an image patch or as a spatio-temporal volume, called cuboids, in a video. The spatio-temporal interest point based methods have received a lot of attention in the vision community due to their robustness to scale and viewpoint invariance. Laptev et al. [16] introduced the Space−Time Interest Point detector, a three-dimensional (3-D) variant of the Harris corner detector [26], which identified the points with high variations in intensity and motion. They used a bag-of-features representation on histogram of oriented gradients (HOG) and histogram of optical flow (HOF) descriptors for recognizing natural human actions [11]. Observing the sparseness of the detected STIP interest points, Dollar [12] proposed an alternative feature detector, which computes the response after the application of separable Gaussian filters in space and a quadrature pair of Gabor filters in the time domain for each pixel, followed by the computation of local maxima. Depending on the response, they simply compute gradients or optical flow on cuboids, then flatten them and finally apply principal component analysis (PCA) for dimension reduction. Scovanner et al. [13] proposed 3-D SIFT, an extension of the SIFT descriptor to spatio-temporal data. The local descriptor approaches are less sensitive to noise or occlusion; however, they require the detection of sufficient and relevant interest points and lack the capability of modeling the holistic geometrical or temporal information. Furthermore, these approaches, often utilize the bag-of-features model, requiring the quantization of large amount of data. Even though the

interest points and the features are computed locally, each sequence is represented by a histogram, which does not carry any spatial or temporal information.

In the early literature, there were also methods modeling the silhouettes [27, 28] and motion [29, 30]. Johansson et al. [31] captured the motion of body joints using landmarks such as the light displays attached to the human body and showed that a few landmarks are adequate to represent the actions. Campbell et al. [32] learned the movements of a ballet dancer using trajectories of attached markers which form a high-dimensional phase space of axes torso location, joint angles and attitude. These methods required the localization of the human body through alignment, background subtraction or tracking. They performed well in a controlled environment; however, on datasets of low-quality web videos, conditions such as the presence of clutter in the background, occlusion and viewpoint changes made the use of these methods impractical.

Recently, trajectory-based methods were proposed for action recognition. Wu [33] captured ensemble motions of a scene on a set of dense Lagrangian particle trajectories, which were computed by numerical integration of optical flow over time. Then actions were described by the use chaotic features extracted on object motion trajectories which were obtained after low rank minimization and clustering of all trajectories. Similarly, Wang et al. [34] represented the videos by computing motion boundary histograms, HOG, HOF and trajectory descriptors along the dense foreground motion trajectories. Then, they utilized the bag-of-features model for the final representation of the videos (The bag-of-features model starts with the computation of descriptors around the interest points, then the descriptors are grouped into K clusters using k-means algorithm. The center of each cluster represents a codeword in a codebook of size K. Finally, the descriptors computed for a video sequence are quantized using the nearest codeword for generating the final representation of histogram of descriptors). Both methods [33, 34] only extracted features on the dynamic trajectories belonging to the foreground and did not capture the context information such as the objects present in the background or scene properties, which may be helpful for recognizing actions. Moreover, computation and analysis of trajectories may be computationally expensive.

There are also works aimed at capturing the relationships between actions and the scene. Ikizler-Cinbis et al. [18] extracted multiple features on the human, objects and scene, and utilized a multiple-instance learning framework for human action recognition on YouTube videos. However, their approach required motion compensation for foreground estimation, and also the detection and tracking of the human in the scene. Liu et al. [17] extracted a combination of motion and static features and utilized the PageRank algorithm to prune the static features using motion cues as an alternative way to motion compensation. The hybrid use of motion and static features improved the performance of their approach.

A little work has been done for representing the global information in videos for classifying individual actions. In this thesis, we propose a holistic descriptor which captures the frequency spectral components related to the motion and scene elements in a video.

The methods mentioned above concentrated on actions of individuals. Next, we describe the methods related to analysis of crowd scenes, where several people present perform activities.

2.2 Analysis of Crowds

The majority of the methods used in various video surveillance systems to monitor areas (stations, streets, malls, etc.) still lack the capabilities to analyze crowd scenes. Limited efforts have addressed the problems of high density crowd scene analysis due to complexity. Furthermore, most studies are aimed at abnormal behavior detection [35, 36, 37, 33], detecting/tracking individuals in crowds [38, 39, 40, 41, 42], counting people in crowds [43, 44, 45], identifying different regions of motion and segmentation [46, 47, 48, 49], or crowd detection [50], rather than identifying collective crowd activities or behaviors. For a more comprehensive review of related work on behavior analysis for crowd scenes, see the survey article [51].

The conventional approach for scene understanding and activity analysis involves object detection, tracking and behavior recognition. This approach requires the use of low-level mo-

tion [52, 53] and appearance features [54, 53], or object trajectories [55, 52]. The methods following this approach perform well in scenes with a low object density, but fail in real-world situations where the scenes contain high to moderate density crowds. In addition, tracking is a hurdle, as methods are often not suitable for multiple target tracking, due to their computational expense or unreliability. In a recent paper [56], they analyze the motion trajectories of multiple objects and learn a Spatio-Temporal Driving Force model for segmentation of group motion patterns, but all these approaches require training.

It is hard to perform tracking reliably in a complicated crowd scene. Therefore, the researchers proposed a holistic approach for activity analysis and scene understanding. This approach avoids tracking as it uses the features directly rather than computing the trajectories for representing activities. This approach requires the use of features such as multi-resolution histograms [36], spatio-temporal cuboids [35, 57], and appearance or motion descriptors [58, 59], etc. In another work [37], a representation based on dynamic textures, in which appearance and dynamics are modeled jointly, is used for detecting anomalies in crowd scenes. Although, these types of holistic approaches are suitable for recognizing actions and detecting and segmenting activities, they also require training, hence the manual labeling of activities.

There are presented methods which use a Lagrangian particle dynamics model of crowd scenes, [20, 21] and treats crowds as collections of mutually interacting particles. These methods are well-suited for small and large crowds [22]. Similarly, [40, 60] uses Lagrangian particle dynamics, based on optical flow, for analyzing the flow in crowd scenes. Mehran et al. [60] utilizes the social force model of Helbing et al. [61] for particles overlaid on the scene and uses the learned normal behavior in the scene to detect any abnormal behavior, but our method is concerned with detecting and identifying particular crowd behaviors. The method of [40] uses the flow of the crowd to aid in tracking individuals, but the method is specific to dense crowds with uniform motion, and our method is less restrictive on the type of scene for applications, requiring only a characteristic flow. Finally, [62, 63] present methods for learning motion patterns in crowd scenes.

In other related work, description of orientation fields by phase portraits [64] and detection of critical points [65] are proposed, and [66] introduced a fluid-dynamic model for simulating the movement of pedestrians and showed the phenomenon of lane formation which may occur in dense real crowds. [67] analyzed the spread of particles near the singularities regarding the problem of oceanic diffusion.

None of these studies in the literature connect flow fields with crowd behaviors. Our method, presented in this dissertation, aims to locate specific instances of crowd behaviors and does not require learning the typical flow. To the best of our knowledge, it is the first attempt in computer vision to identify specific crowd behaviors, i.e. bottlenecks, fountainheads, lanes, arches/rings, and blocking.

In this chapter, we summarized various present approaches for recognizing the activities of individuals and the crowd behaviors. The objective of the work proposed in this dissertation is to fill the mentioned voids in the literature.

CHAPTER 3: A HOLISTIC DESCRIPTOR FOR RECOGNIZING ACTIONS OF INDIVIDUALS

There is a huge demand in computer vision research for a reliable representation of the videos for popular applications such as detecting a certain action in an amateur video or video gaming. A video descriptor represents the visual features of the contents in a video by describing the characteristics such as motion, texture or shape.

In this chapter, we present a holistic video descriptor for classification of actions in videos. We represent each video sequence as a single feature vector that is computed by applying a bank of 3-D spatio-temporal filters on the frequency spectrum of the video sequence. The designed 3-D filter bank captures both scene and motion information in videos. Hence, our descriptor does not require the detection of interest points, the extraction of local video descriptors or the quantization of descriptors into a code book. We tested our method on three datasets, KTH [6], UCF50 [9], HMDB51 [5] and TRECVID 2011 event collection [19], and obtained promising results which demonstrate the robustness and the discriminative power of our holistic video descriptor for classifying videos of various actions. In addition, the combination of our holistic descriptor and a local descriptor resulted in the highest classification accuracies on UCF50 and HMDB51 datasets.

This chapter is organized as follows. Section 3.1 gives an insight into the basics of our approach. In Section 3.2, we describe our approach and the implementation details, followed by the quantitative results in Section 3.3. Finally, we conclude our work in Section 3.4.

3.1 Gist of a Video

The "gist" proposed by Torralba et al. [68] is a holistic scene descriptor connected with power spectrum features, and it has the state-of-the-art performance for scene classification. However, it is not suitable for action recognition as it does not capture the motion information.

Figure 3.1: Example action classes from the **(a)** KTH, **(b)** UCF50 and **(c)** HMDB51 datasets

We believe the computation of frequency spectral components of videos may provide useful scene and motion information for action classification. Here we propose a holistic descriptor for videos, to be used for the action classification of challenging datasets such as UCF50 and HMDB51 with a large number of action classes, some of which are illustrated in Figure 3.1. Our descriptor is generated by applying a bank of 3-D spatio-temporal filters on the frequency spectrum of a sequence.

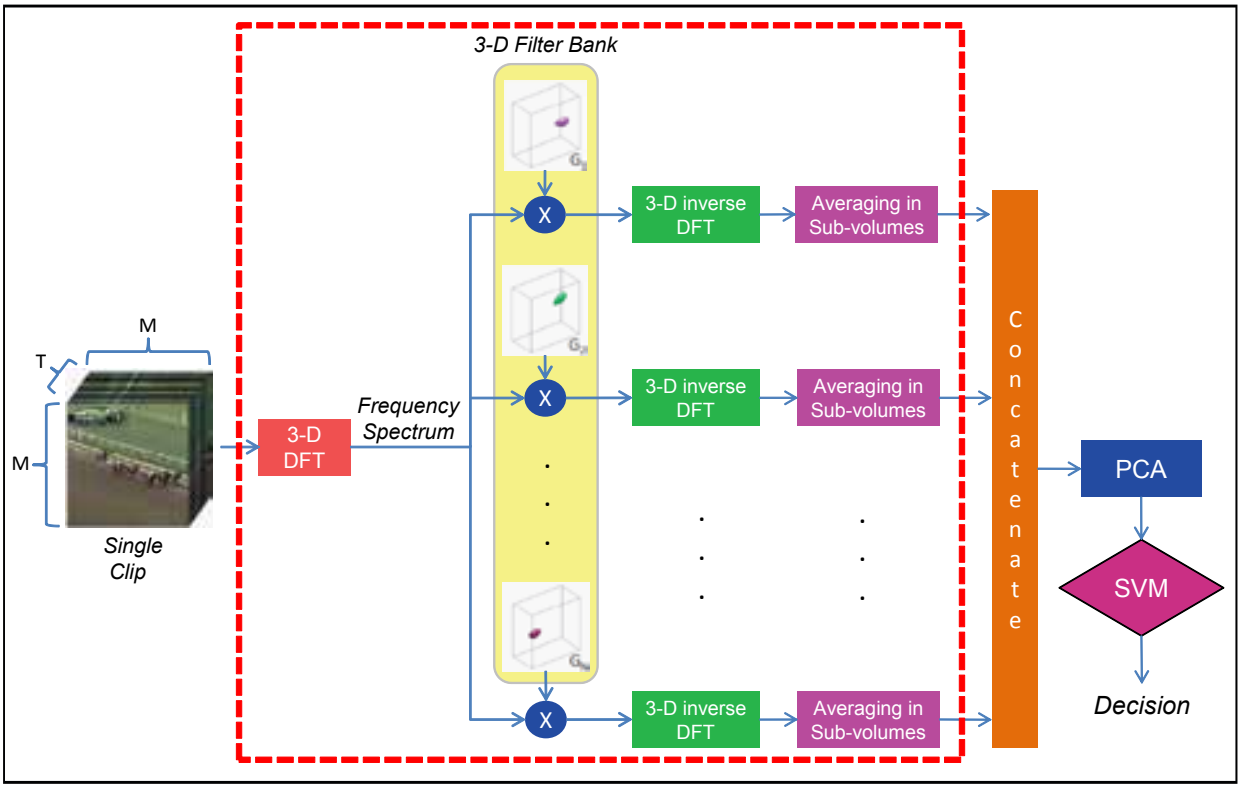

Figure 3.2: Overview of our approach for a single clip: given a single clip, we compute the 3-D Discrete Fourier Transform. Applying each filter of the 3-D filter bank separately to the frequency spectrum, we quantize the output in fixed sub-volumes. Next, we concatenate the outputs and perform dimension reduction by Principal Component Analysis and classification by the use of a support vector machine.

The bandpass nature of these filters alleviates the need for motion compensation. Furthermore, as opposed to the approaches which apply bag-of-features model, our approach preserves the spatial and temporal information, as we perform quantization in fixed spatio-temporal sub-volumes after application of each filter on the frequency spectrum and taking the inverse Fourier transform. As the filter responses for all filters on all sub-volumes are concatenated, the ordering and the length of each feature vector are identical for all represented video clips. The framework of our approach is shown in Figure 3.2. For long sequences, we divide the video into clips and compute our de-

scriptor vectors for each clip, then we concatenate the vectors for the final representation of the video. This is depicted in Figure 3.3.

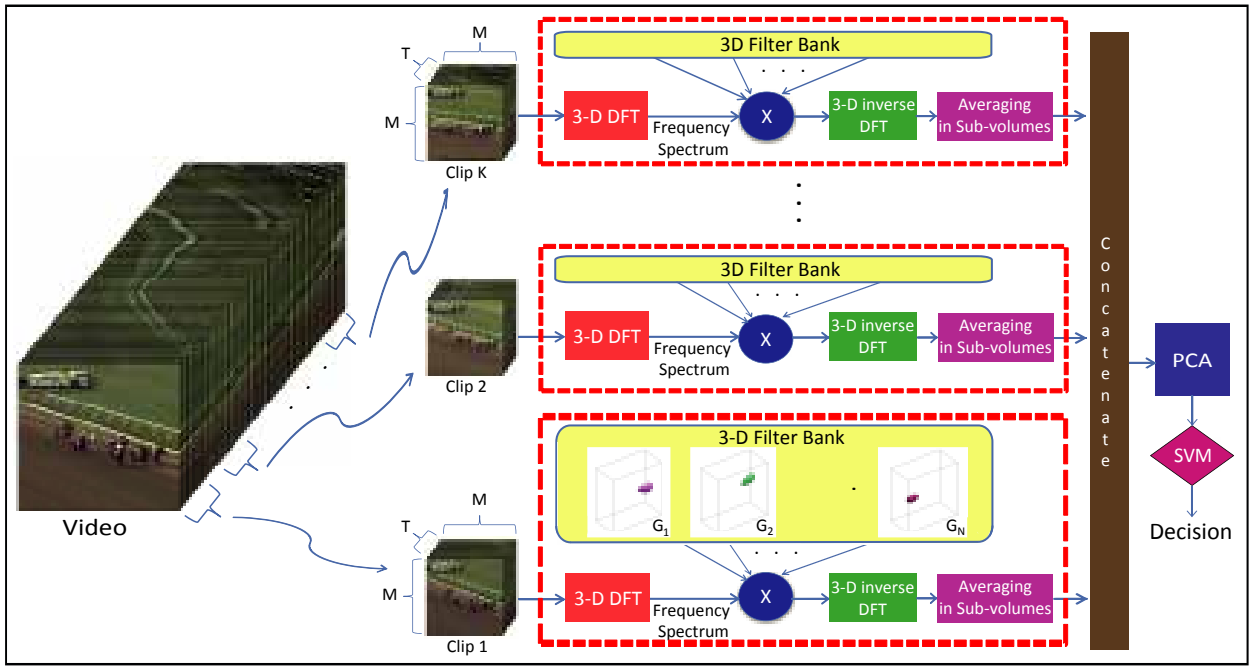

Figure 3.3: Overview of our approach for a video: Given a video of K clips, we compute the descriptor vectors for each clip. Then, we concatenate the vectors and perform dimension reduction by Principal Component Analysis and classification by the use of a support vector machine.

Videos which involve similar actions tend to have similar scene structure and motion. We use the regularities in the appearance or motion to pinpoint the type of actions involved in the videos. The frequency spectrum computed for a video clip could capture both scene and motion information effectively [68, 3, 69], as it represents the signal as a sum of many individual frequency components. In a video clip, the frequency spectrum can be estimated by computing the 3-D discrete Fourier transform (DFT).

The motion is an important element which can be representative of the type of performed action in a scene. It can be explained in a straightforward way by considering the problem in

the Fourier domain [69]. The frequency spectrum of a two-dimensional pattern translating on an image plane lies on a plane, the orientation of which depends on the velocity of the pattern. Given a 2-D image $f_0(x, y)$, we can create a volume, space$-$time image sequence, by translating $f_0(x, y)$ with a velocity $\bar{u} = [u_1 u_2]$ over time. This volume is then expressed as

$$f(x, y, t) = f_0(x - u_1 t, y - u_2 t). \tag{3.1}$$

The three-dimensional discrete Fourier transform of $f(x, y, t)$ over space and time is computed as

$$F(f_x, f_y, f_t) = \frac{1}{MNT} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \sum_{t=0}^{T-1} f(x, y, t) \quad e^{-j2\Pi(\frac{xf_x}{M} + \frac{yf_y}{N} + \frac{tf_t}{T})}, \tag{3.2}$$

where M, N and T are the width, height and length of the clip, and x, y and t are the spatial positions and time of each point in the created volume. Here, the 3-D DFT of the volume will have the same size as the volume itself.

After substituting Eq. 3.1 in Eq. 3.2 and rearranging the terms, the Fourier transform formula would be

$$F(f_x, f_y, f_t) = \frac{1}{MNT} \sum_{t=0}^{T-1} \left[\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f_0(x - u_1 t, y - u_2 t) e^{-j2\Pi(\frac{xf_x}{M} + \frac{yf_y}{N})} \right] e^{-j2\Pi \frac{tf_t}{T}}. \tag{3.3}$$

The inner term in Eq. 3.3 is actually the 2-D discrete Fourier transform of $f_0(x - u_1 t, y - u_2 t)$. Hence the equation may be simplified to

$$F(f_x, f_y, f_t) = \frac{1}{T} \sum_{t=0}^{T-1} F_0(f_x, f_y) e^{-j2\Pi(\frac{u_1 tf_x}{M} + \frac{u_2 tf_y}{N})} e^{-j2\Pi \frac{tf_t}{T}}, \tag{3.4}$$

$$= F_0(f_x, f_y) \frac{1}{T} \sum_{t=0}^{T-1} e^{-j2\Pi t(\frac{u_1 f_x}{M} + \frac{u_2 f_y}{N})} e^{-j2\Pi \frac{tf_t}{T}}, \tag{3.5}$$

where $F_0(f_x, f_y)$ represents the 2-D discrete Fourier transform of $f_0(x, y)$.

The discrete Fourier transform of the complex exponential term is a Dirac delta function, hence the frequency spectrum of the volume in Eq. 3.1 will be simplified to

$$F(f_x, f_y, f_t) = F_0(f_x, f_y)\,\delta\left(\frac{u_1 f_x T}{M} + \frac{u_2 f_y T}{N} + f_t\right), \tag{3.6}$$

where δ is the Dirac delta function. Thus $F(f_x, f_y, f_t)$ will have non-zero values on a plane passing through the origin, as the delta function will be non-zero only when $\left(\frac{u_1 f_x T}{M} + \frac{u_2 f_y T}{N} + f_t\right) = 0$, as shown in Figure 3.4(a−c). This derivation shows that analyzing the Fourier transform of a signal, the motion in a sequence can be estimated by finding the plane which contains the power. Furthermore, multiple objects with different motion will generate frequency components in multiple planes as depicted in Figure 3.4(d−r).

Since the motion can occur in different directions and frequencies, in our work we use 3-D Gabor filters of different orientations and center frequencies to effectively capture the motion information in a video clip. By filtering the frequency spectrum with a certain oriented filter and taking the inverse Fourier transform, the motion and scene components which are normal to the orientation of the filter are pronounced, as illustrated in the example in Figure 3.5.

3.2 Implementation

A flowchart describing the implementation of our method is shown in Figure 3.3. Our goal is to represent each video sequence by a single holistic descriptor and perform the classification of actions in videos.

For the current implementation, we extract K uniformly sampled clips of a fixed length from each given video. As the second step, we compute the 3-D DFT and obtain the frequency spectrum of each clip as given by Eq. 3.2. In order to capture the components at various intervals of the frequency spectrum of a clip, we apply a bank of narrow band 3-D Gabor filters with different orientations and scales.

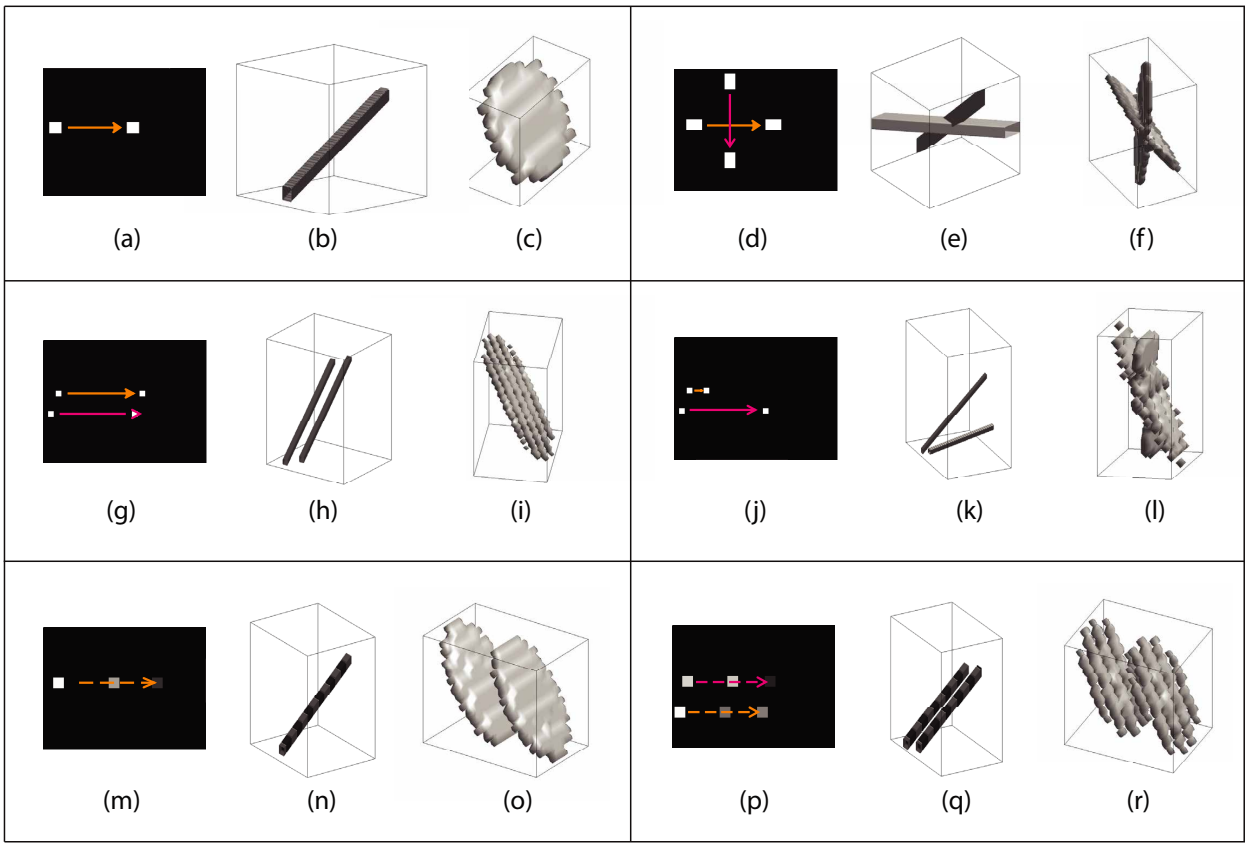

Figure 3.4: Orientation of frequency spectrums: the translating object **(a)** generates a space−time volume **(b)**, and a frequency spectrum of non-zero values on a plane **(c)**. Similarly, motion in different orientations **(d)** results in the volume in **e** and frequency spectrum **(f)** with two planes. Uni-directional motion results in a single plane in the frequency spectum **(g−i)**. Motion with different velocities **(j)** corresponds to two planes in the frequency spectum **(l)**. A translating object with a sinusoidal intensity over time **(m−n)** resulted in two identical planes in frequency spectrum with a separation based on the frequency of the object **(o)**. For multiple objects introducing more gradients **(p, g)**, the planes are still present but appear partially **(i, r)**.

The transfer function of each 3-D filter, tuned to a spatial frequency f_{r_0} along the direction specified by the polar and the azimuthal orientation angles θ_0 and ϕ_0 in a spherical coordinate

system, can be expressed by

$$G(f_r, \theta, \phi) = exp\left\{-\frac{(f_r - f_{r_0})^2}{2\sigma_r^2} - \frac{(\theta - \theta_0)^2}{2\sigma_\theta^2} - \frac{(\phi - \phi_0)^2}{2\sigma_\phi^2}\right\}, \tag{3.7}$$

where $f_r = \sqrt{f_x^2 + f_y^2 + f_t^2}$, $\theta = arctan\left(\frac{f_y}{f_x}\right)$ and $\phi = arccos\left(\frac{f_z}{\sqrt{f_x^2+f_y^2+f_t^2}}\right)$. The parameters σ_r, σ_θ and σ_ϕ are the radial and angular bandwidths, respectively, defining the elongation of the filter in the spatio-temporal frequency domain.

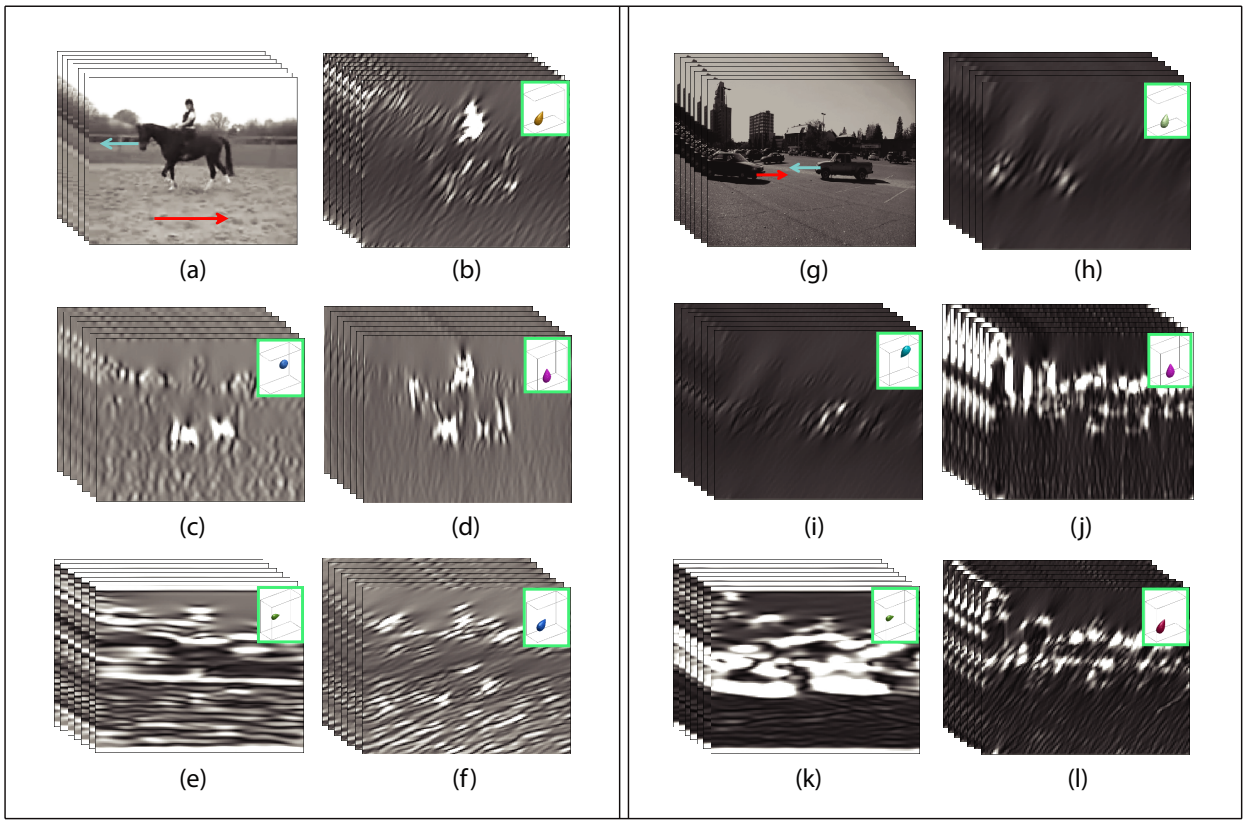

Figure 3.5: Effect of filtering the frequency spectrum: using different orientations of 3-D filters on the frequency spectrum for the sample clips **(a, g)**, the components with different motion **(b, c, h, i)**, vertical scene components **(d, j)**, horizontal scene components **(e, k)**, and diagonal scene components **(f, l)** are highlighted. The *red* and *cyan* arrows show the direction of motion in the two videos. The applied filters are shown in *green* bounding boxes.

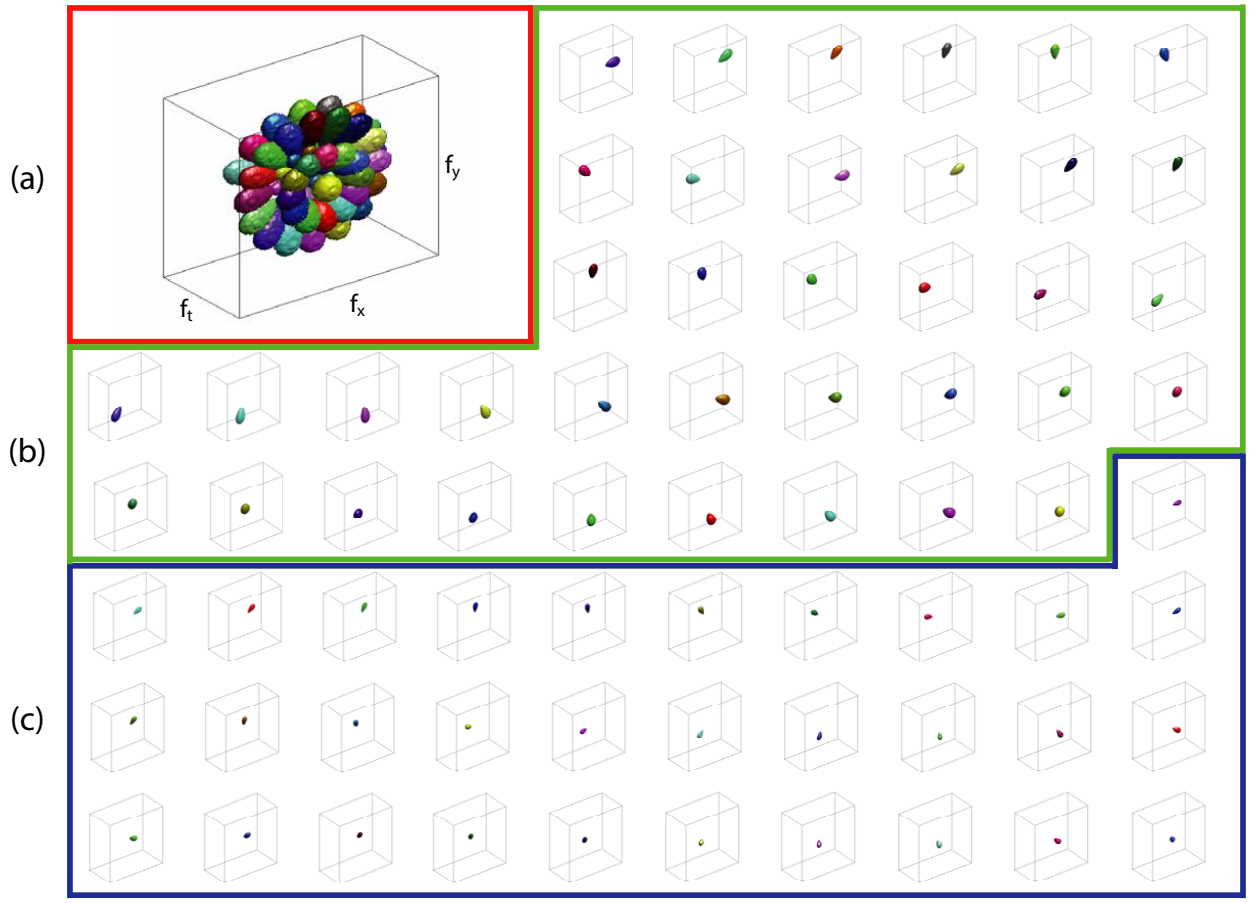

Figure 3.6: Visualization of the filters in 3-D: all filters from the first scale **(b)** and the second scale **(c)** are shown together in **(a)**. (For visualization, we specified a cutoff at 3 dB on the filters)

The combination of our filters is selected to cover only half of the total volume of the frequency spectrum due to the symmetrical nature of the discrete Fourier transform. 3-D plots of these filters are shown in Figure 3.6. Applying each generated 3-D filter on the frequency spectrum of the clip, we compute the output

$$\Gamma_i(f_x, f_y, f_t) = F(f_x, f_y, f_t)\ [G_i(f_x, f_y, f_t)], \tag{3.8}$$

where $\Gamma_i(f_x, f_y, f_t)$ is the output when the i^{th} filter is applied. Then we take the inverse 3-D DFT

$$H_i(x, y, t) = \sum_{f_x=0}^{M-1} \sum_{f_y=0}^{N-1} \sum_{f_t=0}^{T-1} \Gamma_i(f_x, f_y, f_t) e^{j2\Pi(\frac{xf_x}{M} + \frac{yf_y}{N} + \frac{tf_t}{T})}. \tag{3.9}$$

By quantizing the output volume in fixed sub-volumes and taking the sum of each sub-volume and performing the same computation for each filter in our filter bank, we obtain a long feature vector which represents a single clip. This feature vector has the advantage of preserving the spatial information as the response of each filter on each sub-volume contributes to an element in the concatenated feature vector. The last step is to apply PCA, a popular method for dimensionality reduction, in order to generate our holistic video descriptor.

3.3 Experimental Results

To test the performance of our approach, we used publicly available datasets: KTH, UCF50, and HMDB51. UCF50 and HMDB51 are the two of the most challenging datasets with large number of classes, which are collections of thousands of low-quality web videos with camera motion, different viewing directions, large interclass variations, cluttered backgrounds, occlusion and varying illumination conditions. We also conducted experiments on TRECVID'11 event collection.

Instead of computing the corresponding feature vectors on all clips of a video, we uniformly sampled K clips of a given video, as shown in Figure 3.3. This reduced the computation and memory requirements for the final descriptor generation (Alternatively, the key clips can be picked automatically by detecting the shot boundaries). We performed an experiment for the sampling of different numbers of clips for generating our descriptor and observed that sampling more than three clips did not result in a further improvement in performance as depicted in Figure 3.7. Hence, for all experiments, we picked three key clips of 64 frames from each video and downsampled the frames of clips to a fixed size (128×128) for computational efficiency. Next, we computed the 3-D DFT to compute the frequency spectrum of each clip and then applied the generated filter bank.

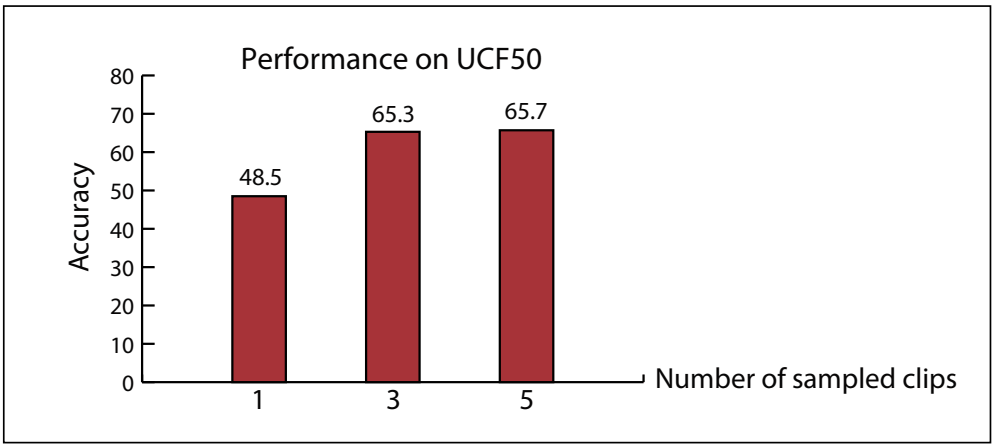

Figure 3.7: Effect of number of sampled clips on the classification performance

Our generated filter bank, described in Section 3.2, consisted of 68 3-D Gabor filters, which corresponded to 2 scales and 37 and 31 orientations for the first and second scales, respectively, in the spatio-temporal frequency domain. The filters are shown in Figure 3.6. The selection of filters was designed experimentally to capture the frequency components effectively as shown in Figure 3.8; the cumulative power spectrum of 500 videos of different actions was computed and our filter bank captures more than 99% of the total power. There was no need for another set of filters, which captures very high frequencies with negligible power. However, increasing the number of filters in the pass band makes the filters become narrower and the descriptor to have a finer response, with a penalty of higher computational requirements. As an experiment, we tested a three-scale filter set with 64 narrower filters per scale and obtained an additional 2.5% performance improvement on UCF50. Considering the computation time trade-off, we did not use this configuration for the reported results.

In our experiments, the central frequencies for the two scales of filters were set to 38.8 and 19 with radial bandwidths 14.2 and 8.6, respectively. The angular bandwidths σ_θ and σ_ϕ were set to 0.2 and 0.1, respectively. Each of the filters we computed had $128 \times 128 \times 64$ as the size

of the frequency spectrum of clips. After the application of the filters, we computed the average response of filters on 512 uniformly spaced $16 \times 16 \times 8$ sub-volumes, to quantize and generate the holistic feature vector for the clip. The length of the feature vector in our experiments was 104,448, as there are 68 filters, 512 sub-volumes and 3 key clips. We reduced the dimensionality of the feature vectors to 2,000 using PCA [70]. To achieve even higher performance, we tested combining our descriptor with a state-of-the-art local descriptor [71], Space−Time Interest Points (STIP). We computed dense STIP features, and generated 1,000 and 2,000-dimensional codebooks to represent each sequence as a histogram.

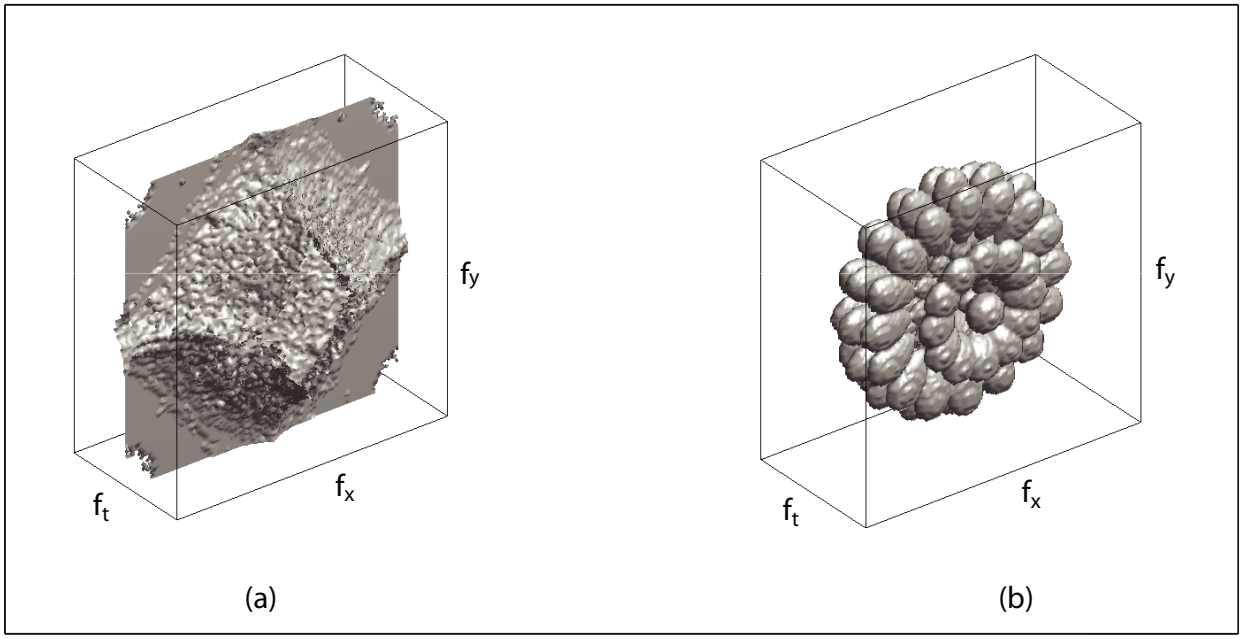

Figure 3.8: The cumulative power spectrum of 500 sample videos **(a)** is captured effectively by the selection of our 3-D filter bank **(b)**.

For classification, we trained a multi-class support vector machine (SVM) [72] using the linear kernel for our descriptor and histogram intersection kernel for STIP. We performed cross validation by leaving one group out for testing and training the classifier on the rest of the dataset

and performing the same experiment for all groups on UCF50. For HMDB51, we performed cross validation on the three splits of the dataset. We did not include any clips of a video in the test set if any other clip of the same video was used in the training set.

The discriminative power of our descriptor can be seen clearly in the example in Figure 3.9. This basic experiment was done using four sequences from a public dataset. For each of the four sequences, we computed the descriptors. Each entry in the matrix in Figure 3.9(c) is the normalized Euclidean distance between the computed descriptors of the four sequences. As seen in the matrix, the descriptor distances between the jumping actions in two different scenes is comparably lower than the other distances, which shows that our descriptor can generalize over intra-class variations. The distances are high when different actions are performed in different scenes, such as the ones labeled by blue arrows in Figure 3.9(a).

To illustrate the advantage of the 3-D holistic descriptor, we compare our descriptor with the popular descriptors: GIST [68] (on UCF50 dataset) and STIP [16, 71] (on KTH, UCF50 and HMDB51 datasets) which involve the computation of histograms of oriented gradients (HOG) and histograms of optical flow (HOF). For comparison, we also list the performance of a low-level descriptor based on color and gray values [5] (on UCF50 and HMDB51 datasets), and the biologically motivated C2 features [73, 5] (on KTH and HMDB51 datasets). Figure 3.10 shows the comparison of performance over three datasets.

3.3.1 KTH Dataset

The KTH dataset includes videos captured in a controlled setting of six action classes with 25 subjects for each class. As depicted in Table 3.1, our descriptor has a classification accuracy of 92.0%, which is comparable to the state of the art. Figure 3.11 shows the confusion table for this dataset. This experiment shows that our descriptor is able to discriminate between the actions with different motions appearing in similar scenes.

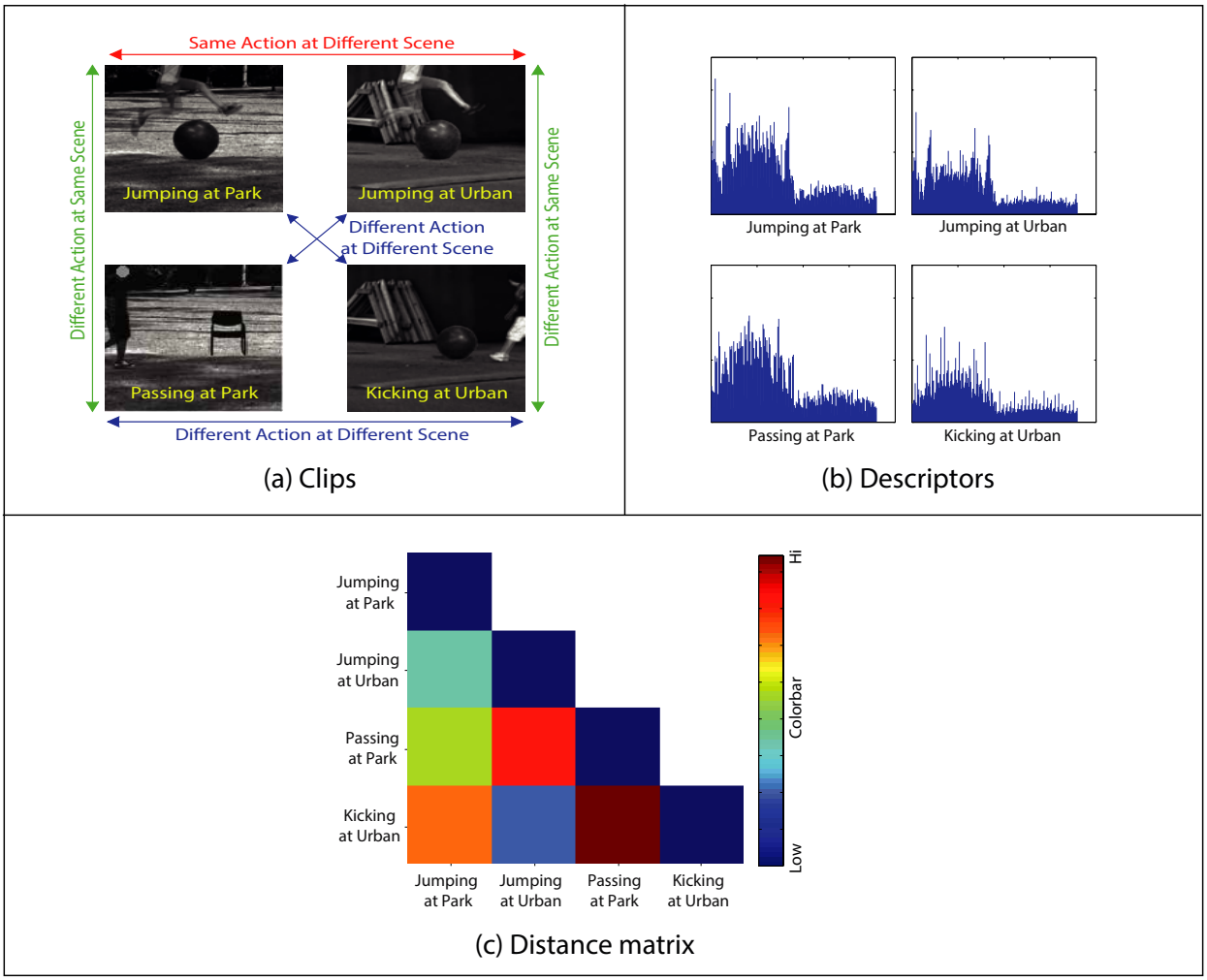

Figure 3.9: Descriptor distances for example clips: for the clips with the similarities and differences mentioned in **(a)**, the distances of the computed descriptors **(b)** are shown as a color-coded matrix in **(c)**. The descriptors with similar actions and scene have lower distances.

3.3.2 UCF50 Dataset

This dataset includes unconstrained web videos of 50 action classes with more than 100 videos for each class. As depicted in Table 3.2, our descriptor, GIST3D, has an accuracy of 65.3% over 50 action classes, which outperforms GIST and STIP. For evaluating the performance of the GIST descriptor, we have used various numbers (3, 20, 40) of sampled frames for each video

and performed classification after concatenating the computed descriptors for each sampled frame. The accuracy increased up to 42.4% when 40 frames were used. Figures 3.14, 3.15 and 3.16 shows the confusion tables for GIST, STIP and our descriptor, GIST3D. Using the combination of STIP and GIST3D by late fusion resulted in a classification accuracy of 73.7%, which is another 8% improvement in the performance. Figure 3.17 depicts the confusion table for the combined classification.

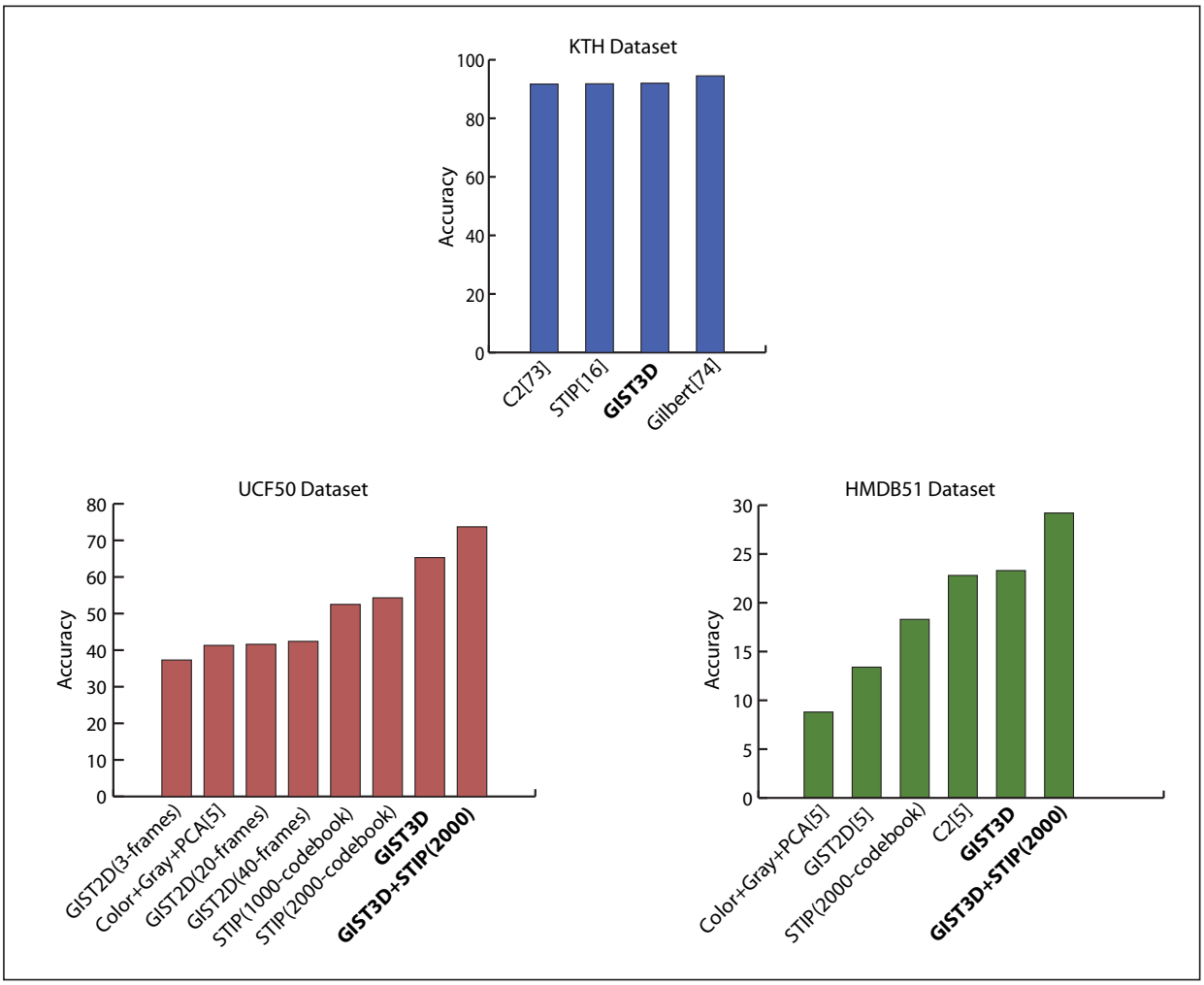

Figure 3.10: Average classification accuracies over KTH, UCF50 and HMDB51 datasets

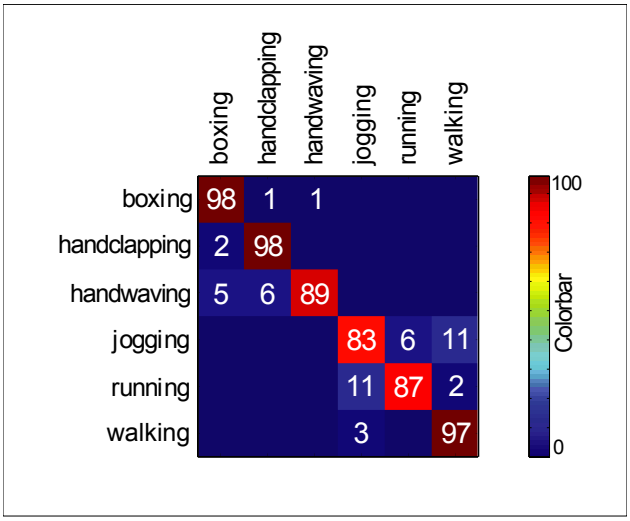

Figure 3.11: Confusion Table for KTH using our descriptor, GIST3D

Table 3.1: Classification Results of 6 action classes of the KTH dataset

Descriptor	Accuracy
C2 [73]	91.7
STIP [16]	91.8
Gilbert [74]	94.5
GIST3D	**92.0**

For comparison of our descriptor to STIP, we also analyzed the average similarities of descriptors among action classes of UCF50. We computed the Euclidean similarity for our descriptors and histogram intersection as the similarity measure for STIP. Our descriptor has higher intra-class similarity and lower inter-class similarity than STIP as shown in Figure 3.12. This clearly explains why our holistic descriptor (GIST3D) performs superior than STIP.

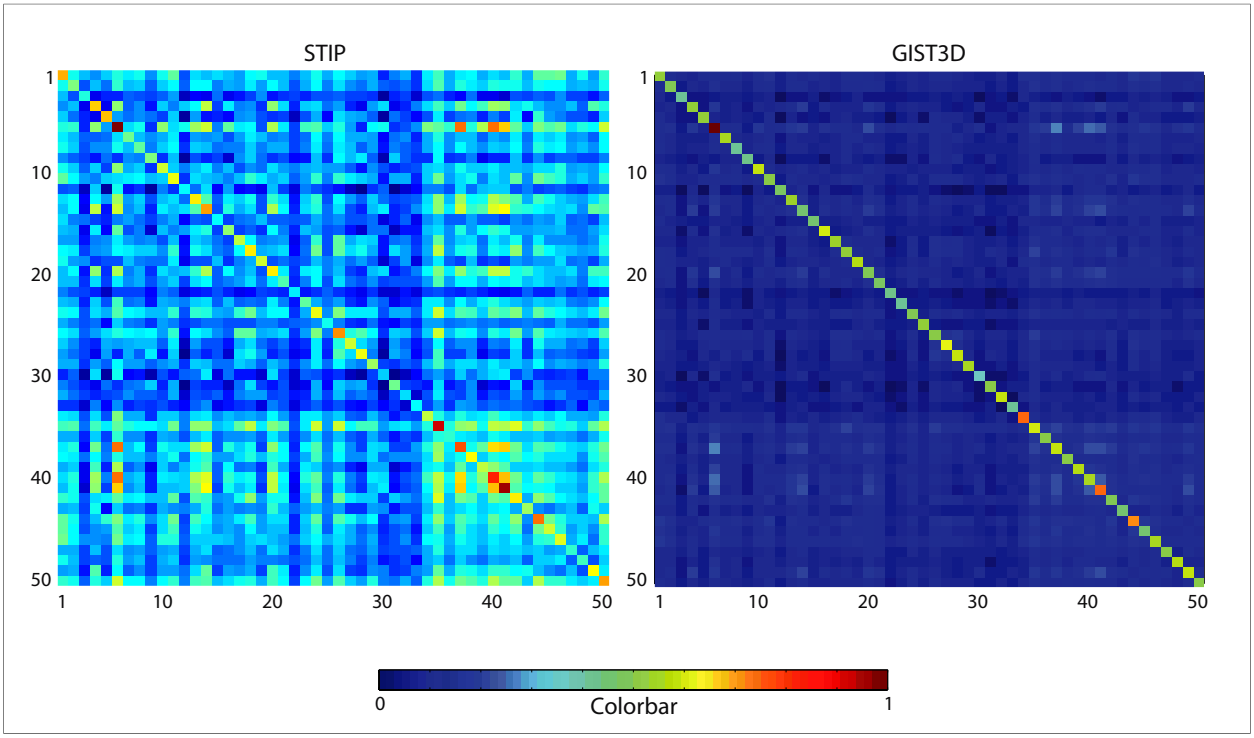

Figure 3.12: Descriptor similarity matrices for STIP **(a)** and GIST3D **(b)** computed among 50 action classes of UCF50 dataset

3.3.3 HMDB51 Dataset

The HMDB51 dataset includes videos of 51 action classes with more than 101 videos for each class. As depicted in Table 3.3, our descriptor, GIST3D, has a classification accuracy of 23.3% over 51 action classes, which outperforms STIP by 5%. The late fusion classifier of these two descriptors resulted in a 6% improvement in the performance over using just our descriptor. Figures 3.18, 3.19 and 3.20 show the confusion tables for STIP, GIST3D and the fused classifier. The actions in the video sequences of HMDB51 are not isolated; multiple actions may be present in a single video sequence despite a given single class label for the sequence. There is also large intra-class scene variation. Therefore, classifying actions on this dataset is more challenging and the performances of the mentioned methods are lower.

Table 3.2: Classification Results of 50 action classes of the UCF50 dataset

Descriptor	Accuracy
GIST(3-frames)	37.3
Color+Gray+PCA [5]	41.3
GIST(20-frames)	41.6
GIST(40-frames)	42.4
STIP(HOG/HOF)(1000-dim codebook)	52.5
STIP(HOG/HOF)(2000-dim codebook)	54.3
GIST3D	**65.3**
GIST3D+STIP(2000-dim codebook)	**73.7**

Table 3.3: Classification Results of 51 action classes of the HMDB51 dataset

Descriptor	Accuracy
Color+Gray+PCA [5]	8.8
GIST [5]	13.4
STIP(HOG/HOF)(2000-dim codebook) [71]	18.3
C2(Motion+Shape) [5]	22.8
GIST3D	**23.3**
GIST3D+STIP(2000-dim codebook)	**29.2**

As another experiment, we compared our descriptor to STIP by analyzing the average similarities of descriptors among 51 action classes. We computed the Euclidean similarity for our descriptors and histogram intersection as the similarity measure for STIP and observed that our descriptor has higher intra-class similarity and lower inter-class similarity than STIP as shown in Figure 3.13.

3.3.4 TRECVID Dataset

As an additional experiment, we tested our approach on TRECVID 2011 event collection [19], which has 15 event categories: boarding trick, flash mob, feeding animal, landing fish,

wedding, woodworking project, birthday party, changing tire, vehicle unstuck, grooming animal, making sandwich, parade, parkour, repairing appliance,and sewing project. This collection contains more than 2,000 event videos with a high degree of diversity in content, environmental settings, frame rate and resolution. There are also 62 action concepts manually defined and annotated in the collection which results in over 8,300 video clips. We performed both concept and event level classification experiments using 10-fold cross validation. We trained a linear multi-class SVM as the classifier. Figure 3.21 shows the confusion matrix for 15 events on this dataset. Table 3.4 depicts the classification performance of our descriptor using various settings. Our descriptor performed better than the local descriptor, STIP. As another experiment we tested the performance of our descriptor for concept level detection. The performance was evaluated in terms of mean average precision-recall for each of the 62 concepts as shown in Figure 3.22. It is important to note that TRECVID 2011 dataset is one of the most extensive collections of diverse and complex videos.

Table 3.4: Results on Concept and Event Level Classification on TRECVID dataset

Descriptor	Experiment	Accuracy
GIST3D	62 Concepts	36.0
STIP [16]	62 Concepts	31.0
GIST3D+STIP [16]	62 Concepts	40.6
GIST3D	15 Events	37.0

3.3.5 Discussion

By comparing our classification accuracy with the tested descriptors and analyzing the Tables 3.2 and 3.3, we found out that GIST, being a scene descriptor, suffered from the lack of captured motion information. For example, as observed in Figure 3.15, GIST cannot differentiate between the actions of Playing Guitar and Playing Violin which happen in similar indoor scenes.

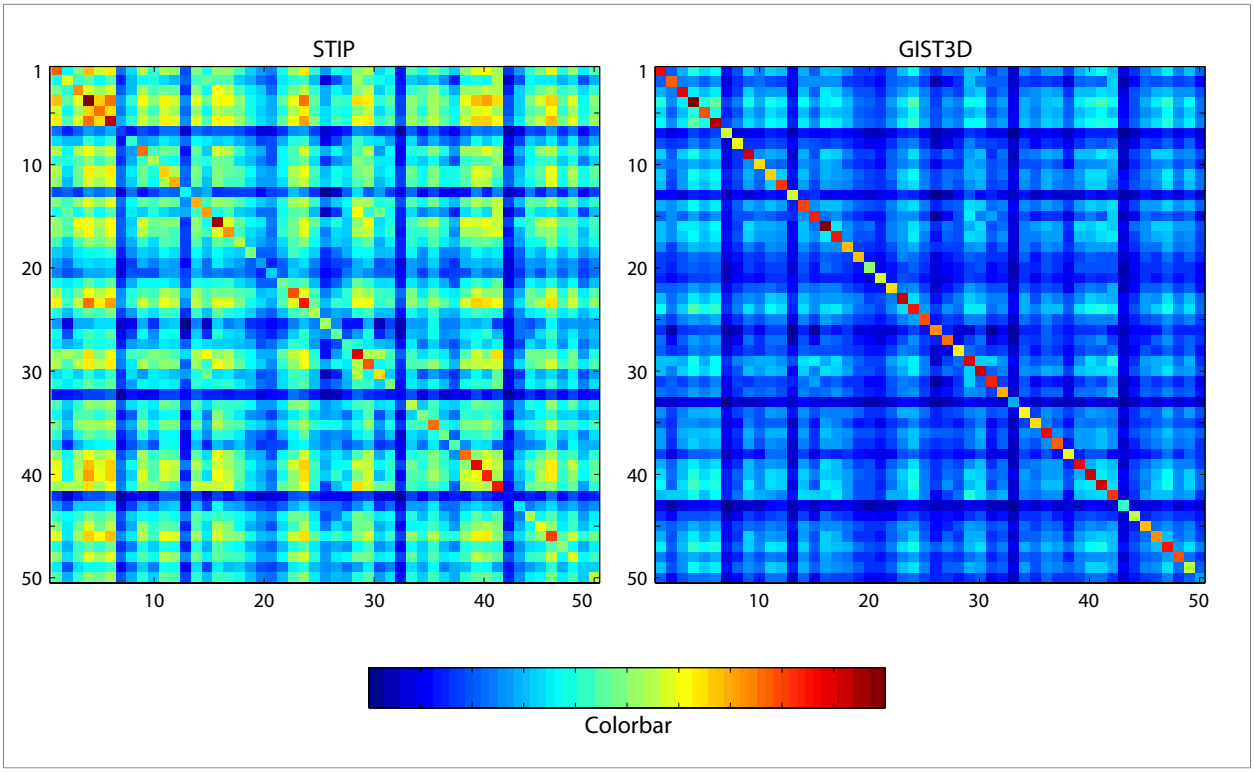

Figure 3.13: Descriptor similarity matrices for STIP **(a)** and GIST3D **(b)** computed among 51 action classes of HMDB51 dataset

The motion is discriminative for these videos and our descriptor (GIST3D) is able to differentiate these actions, as shown in Figure 3.16. Conversely, STIP suffered from locality, as it did not capture the holistic scene structure and did not carry spatio-temporal information due to the global histogram representation. For example, walking with dog and horse riding actions have similar translating motion, and STIP is not as discriminative as our descriptor for these two actions, as depicted in Figure 3.14. The horse riding videos mostly have rural scenes with periodic vertical and horizontal components such as fences, whereas the walking with dog videos contain urban or park scenes. Our descriptor encodes the useful scene information and is able to discriminate between these two actions.

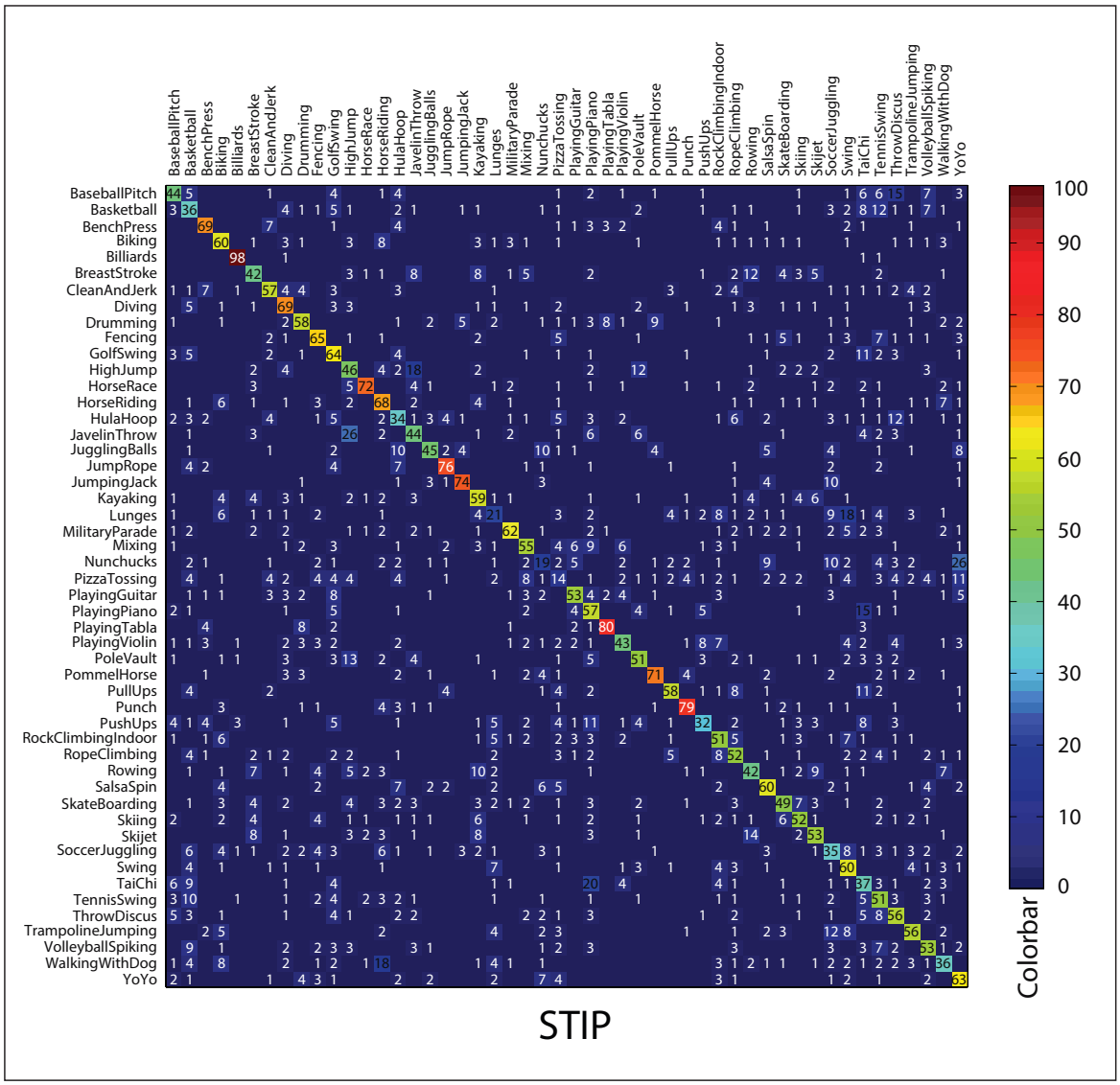

Figure 3.14: Confusion table for STIP over 50 action classes of UCF50 dataset (Average accuracy is 54.3%.)

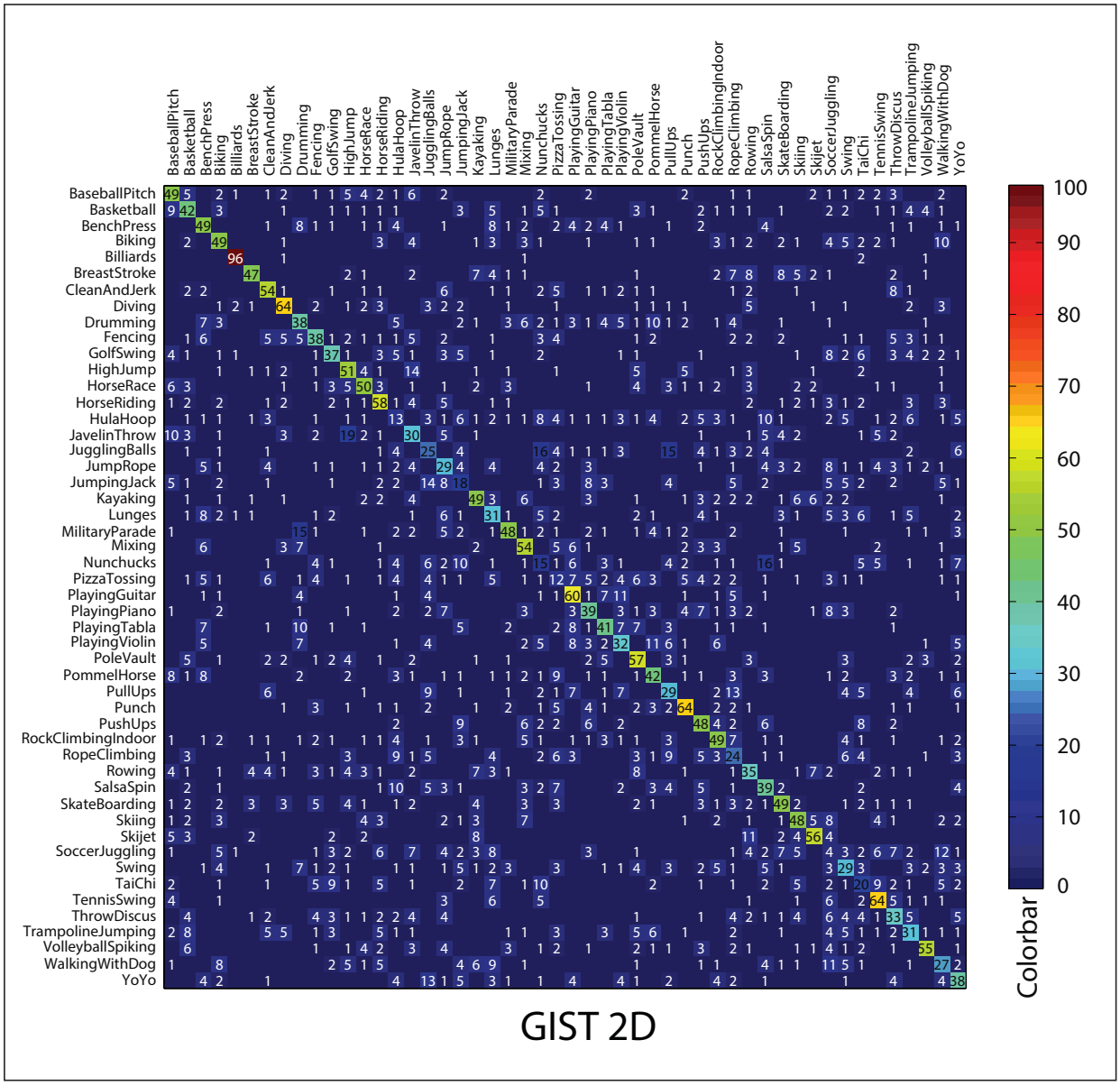

Figure 3.15: Confusion table for GIST 2D (40 images per video) over 50 action classes of UCF50 dataset (Average accuracy is 42.4%.)

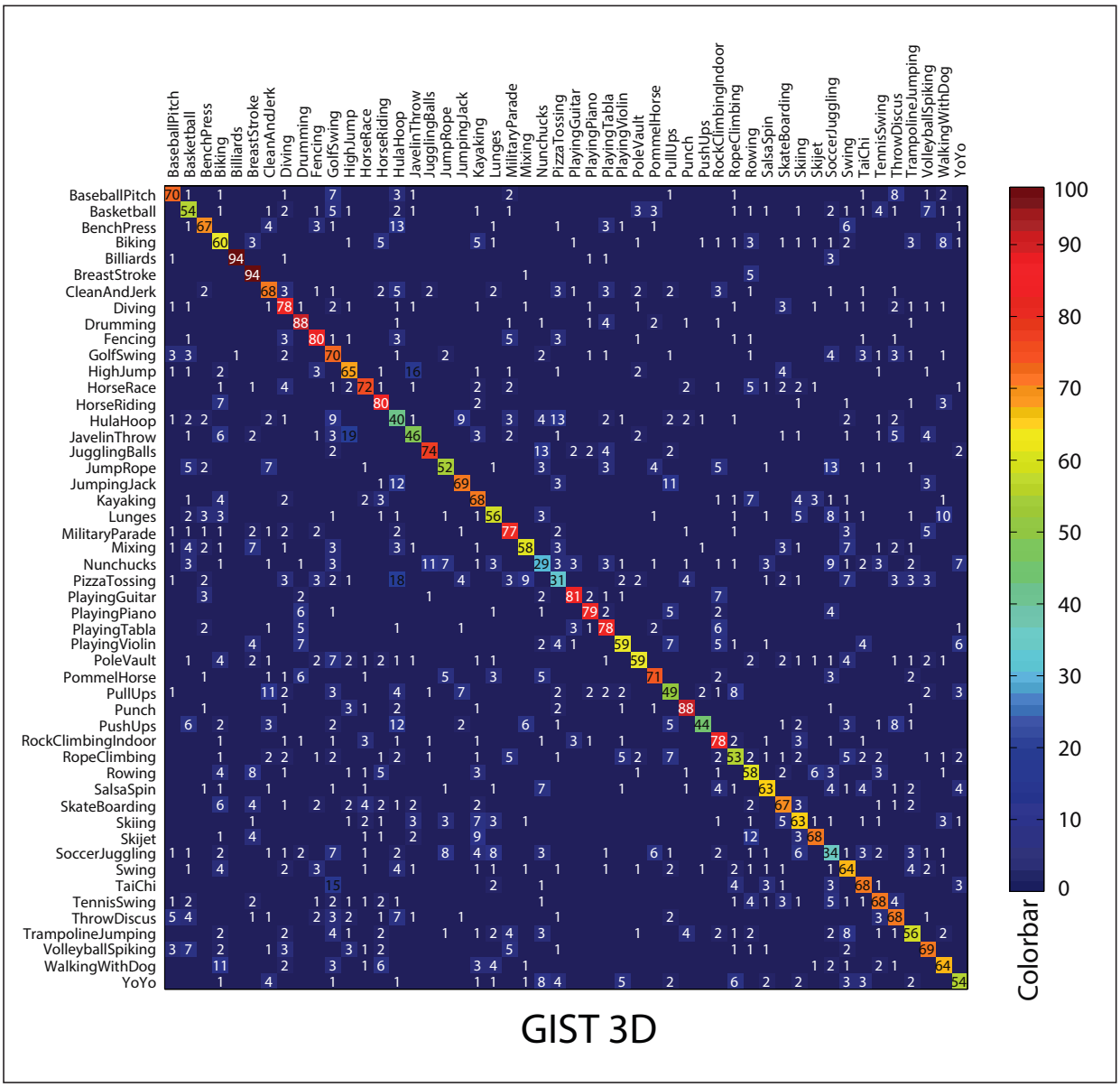

Figure 3.16: Confusion table for GIST3D over 50 action classes of UCF50 dataset (Average accuracy is 65.3%.)

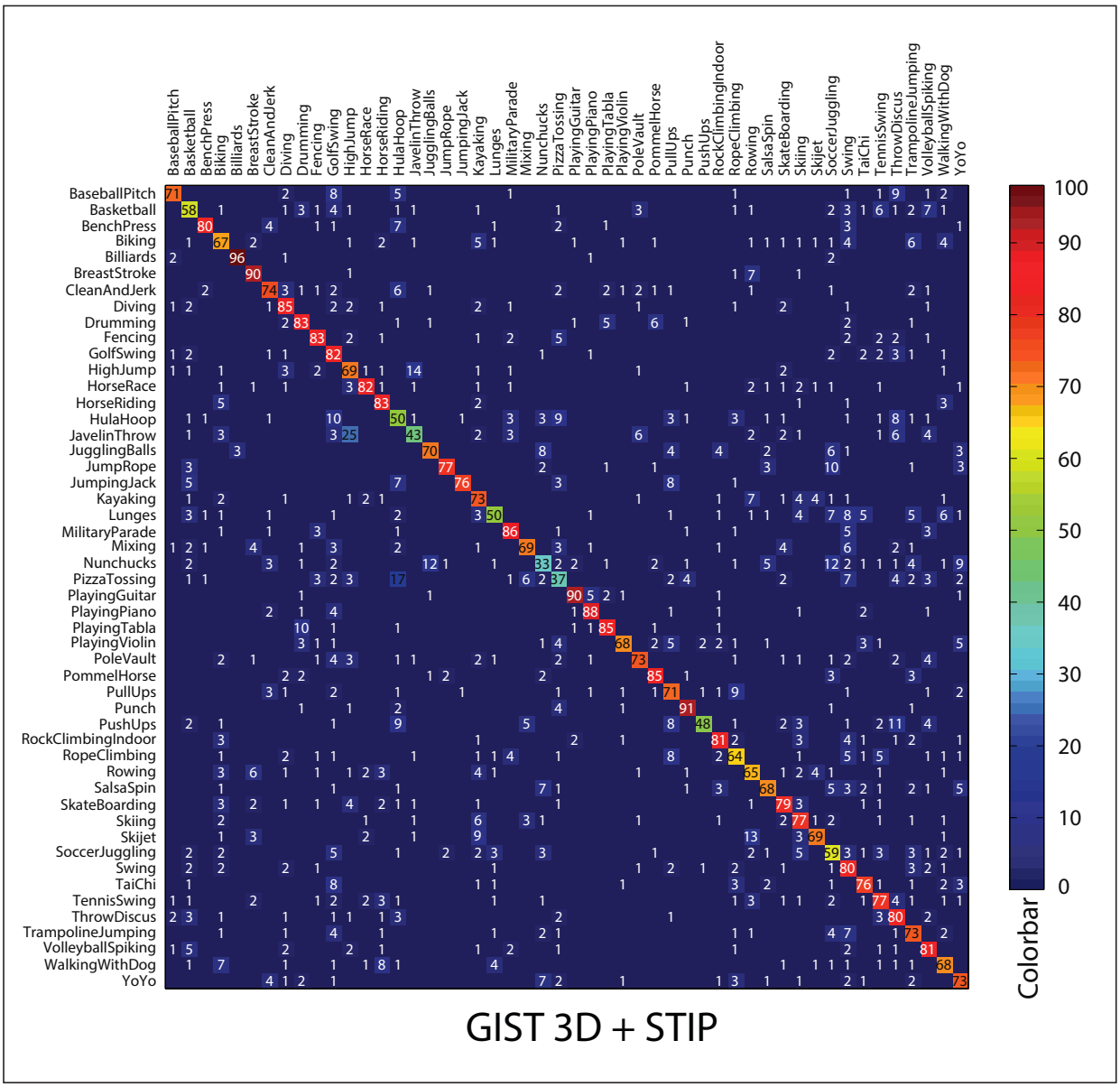

Figure 3.17: Confusion table for the combined descriptor over 50 action classes of UCF50 dataset (Average accuracy is 73.7%)

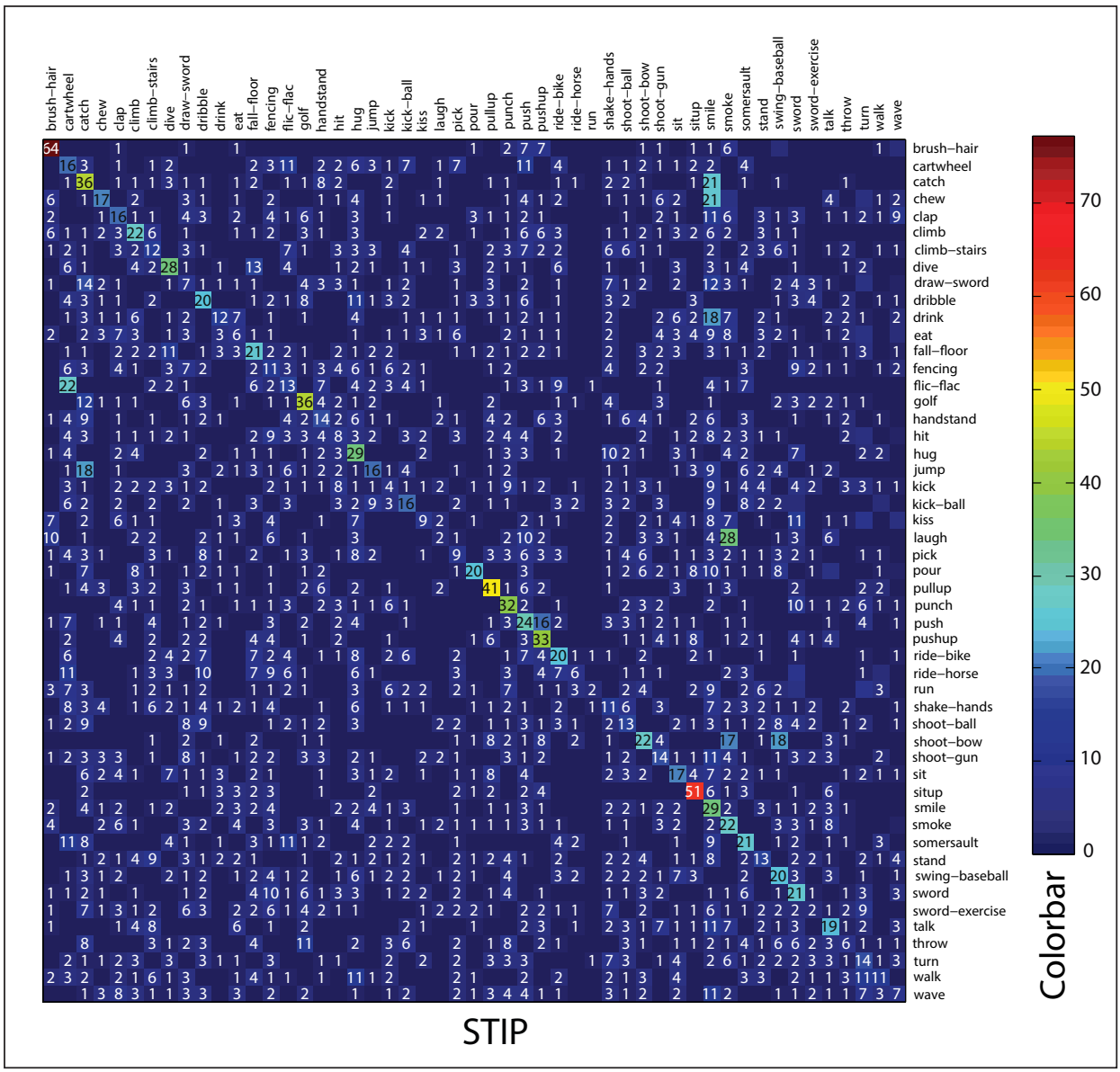

Figure 3.18: Confusion table for STIP over HMDB51 dataset. (Average accuracy for STIP is 18.3%)

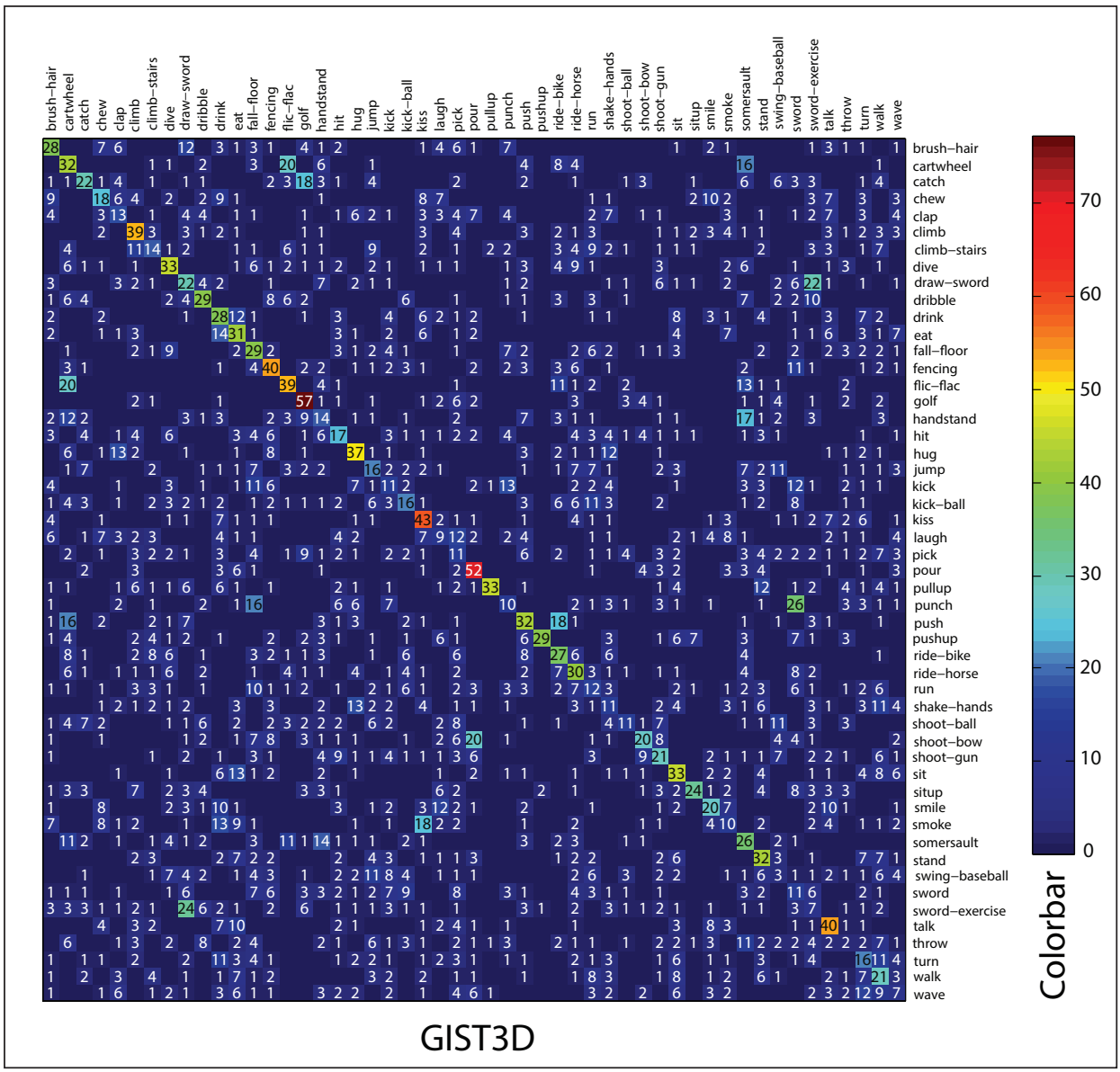

Figure 3.19: Confusion table for GIST3D over HMDB51 dataset. (Average accuracy for GIST3D is 23.3%)

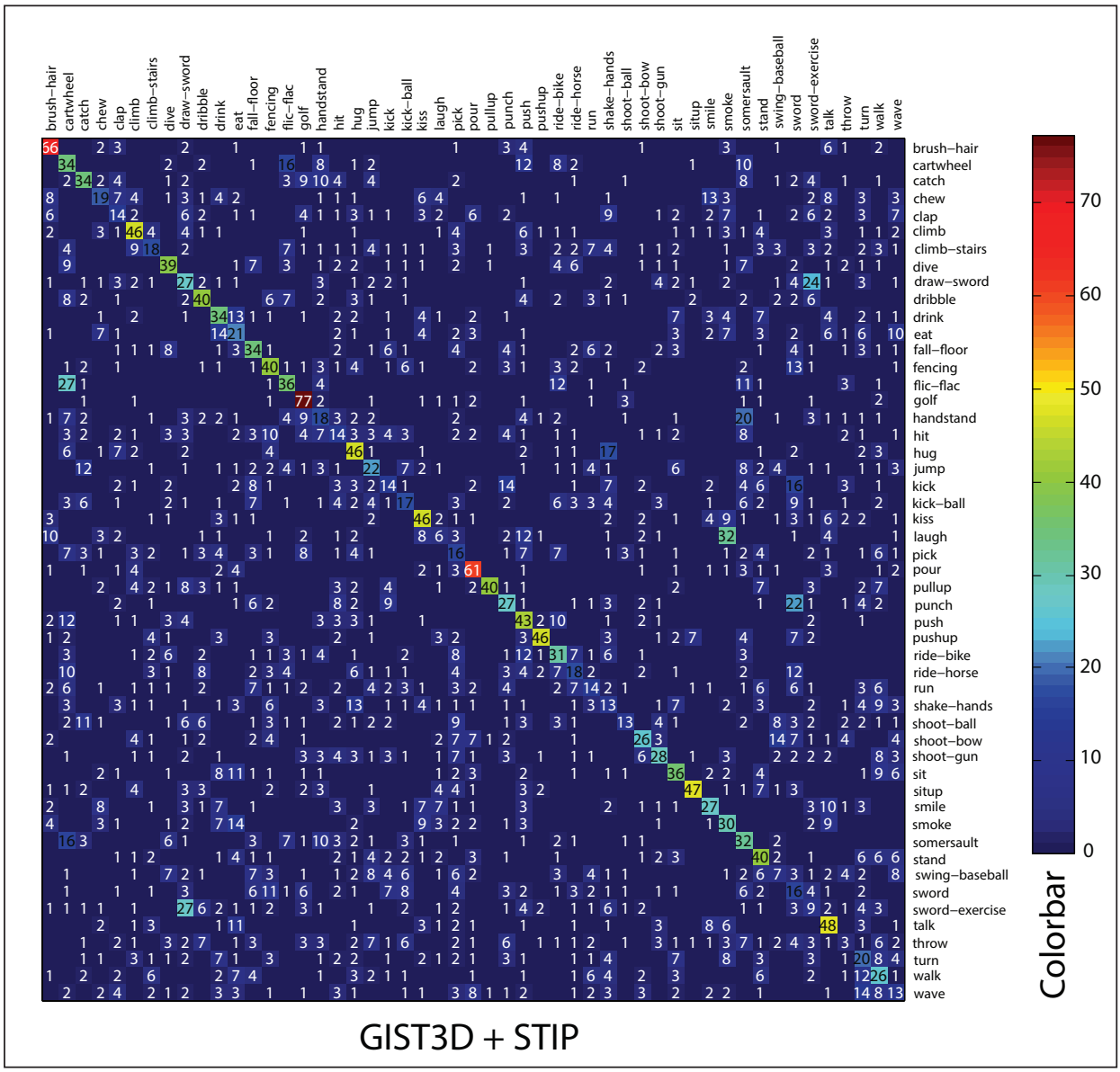

Figure 3.20: Confusion table for the combined classifier over HMDB51 dataset. (Average accuracy for the combined classifier is 29.2%)

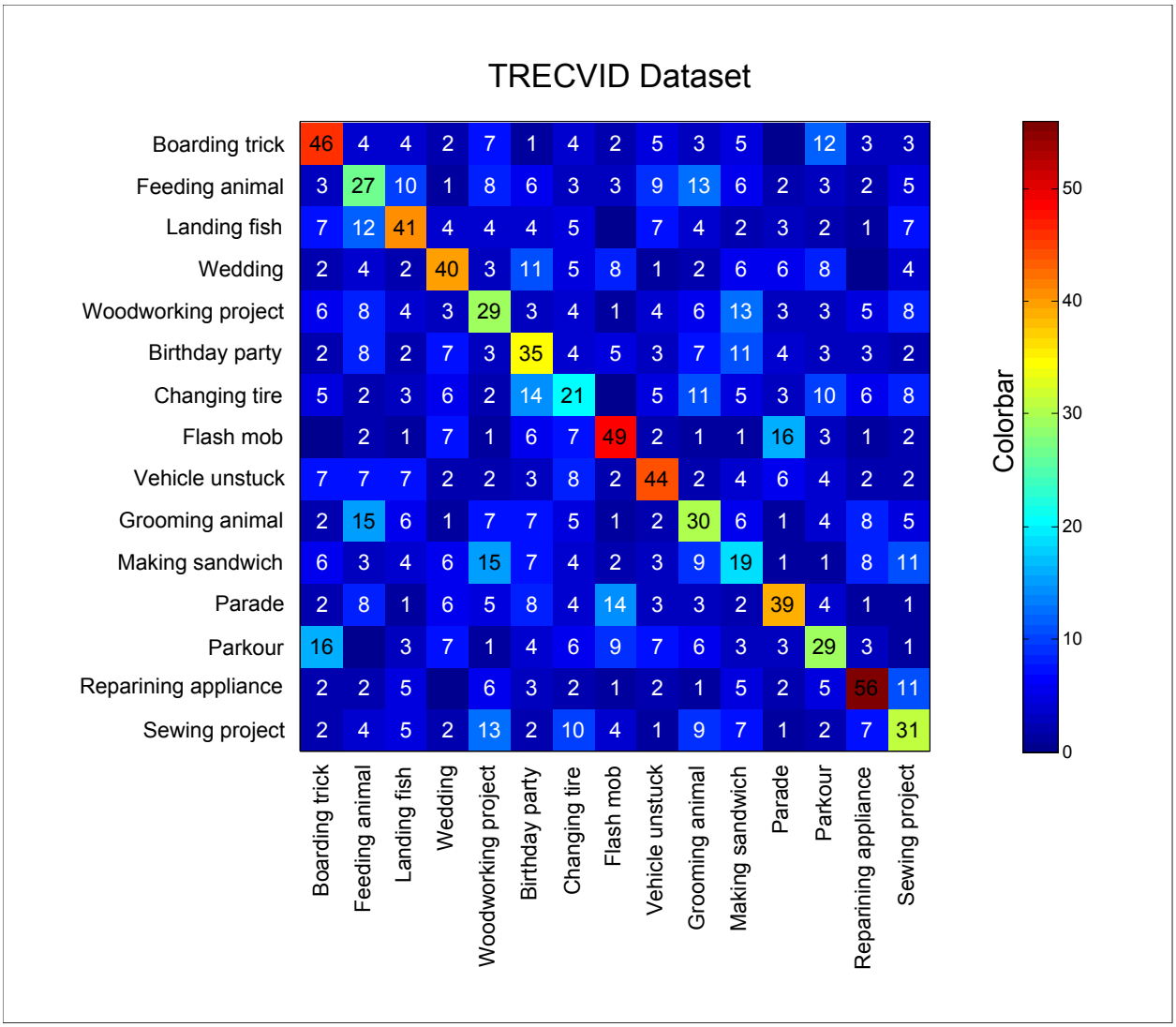

Figure 3.21: Confusion table for our classifier over 15 event classes of TRECVID dataset. (Average accuracy is 37.06%)

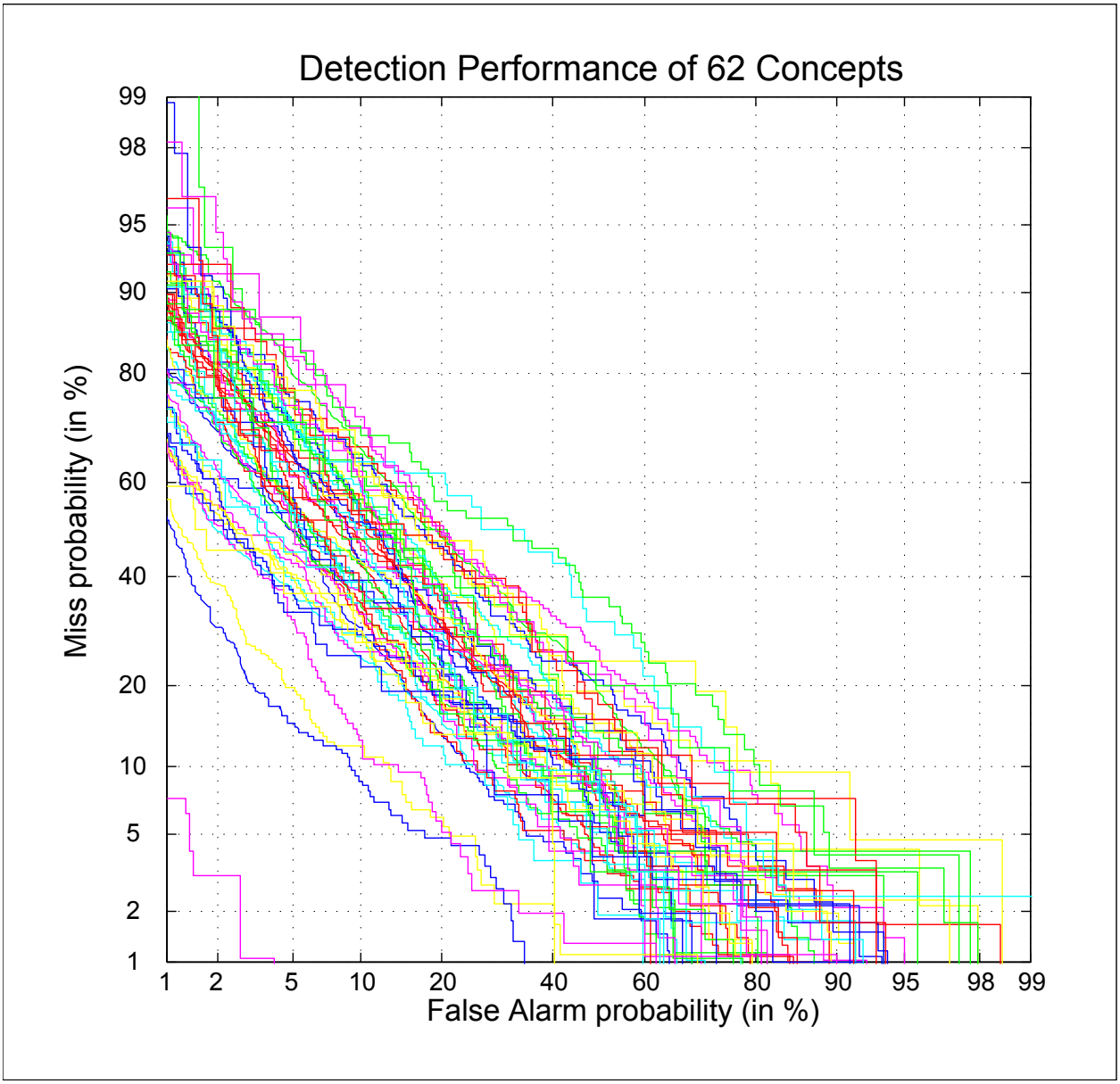

Figure 3.22: Mean average precision-recall of 62 concept detectors on TRECVID dataset.

3.4 Summary

In this chapter, we presented a holistic scene and motion descriptor to classify realistic videos of different actions. The local video descriptors may require the detection of interest points, background subtraction or tracking, whereas our descriptor, GIST3D, bypasses these steps and represents each video with a single feature vector. In contrast to the local descriptors which utilize a bag-of-features framework and then use a non-linear SVM, our descriptor does not have a histogram representation and uses a linear SVM. Preserving also the useful spatial and temporal information, our descriptor had a better performance than the state-of-the-art local descriptor, STIP, utilizing a bag-of-features representation which discards the spatial distribution of the local features. In addition, by combining our descriptor with STIP, we achieved the highest classification accuracies on the challenging datasets, UCF50 and HMDB51. Our descriptor performed comparable to the state-of-the-art descriptors on the KTH dataset, as there was no useful scene information in this dataset. The experiments showed that both scene structure and motion information are important in classifying realistic videos.

CHAPTER 4: DETERMINING DISCRIMINATIVE BLOCKS OF HOLISTIC DESCRIPTORS

In the previous chapter, we proposed a novel holistic descriptor for action recognition, which represents the actions based on the global information of the action and scene. For a video clip, the holistic descriptor can be obtained by concatenating individual descriptors computed on spatio-temporal blocks of the video or by the application of a set of filters globally on the video. As illustrated in Figure 4.1, this representation indiscriminately unites all the information of these descriptor blocks disregarding their beneficence in classification performance. However, some of the blocks may be redundant because they do not capture any useful information or they may be confusing for the classifier.

In this chapter, we present a novel method which improves the performance of descriptors by analyzing the discriminativity of video blocks for the action recognition problem. We measure the discriminativity of a block by examining its response to a pre-learned support vector machine model. In particular, a block is considered highly discriminative if it responds positively for the positive training samples, and negatively for the negative training samples. The goal is to find the optimal blocks by selecting a subset of blocks which maximizes the total classifier discriminativity. Finally, the descriptors are computed only on the selected blocks. We performed experiments on several benchmark datasets [6, 7, 8, 9, 5, 10] to show that our method is able to discover the useful regions in the videos in order to eliminate the ones which are confusing for the classifier. Hence, we obtain a significant performance improvement using a variety of holistic descriptors on the benchmark datasets. Some example actions from the benchmark datasets are illustrated in Figure 4.2.

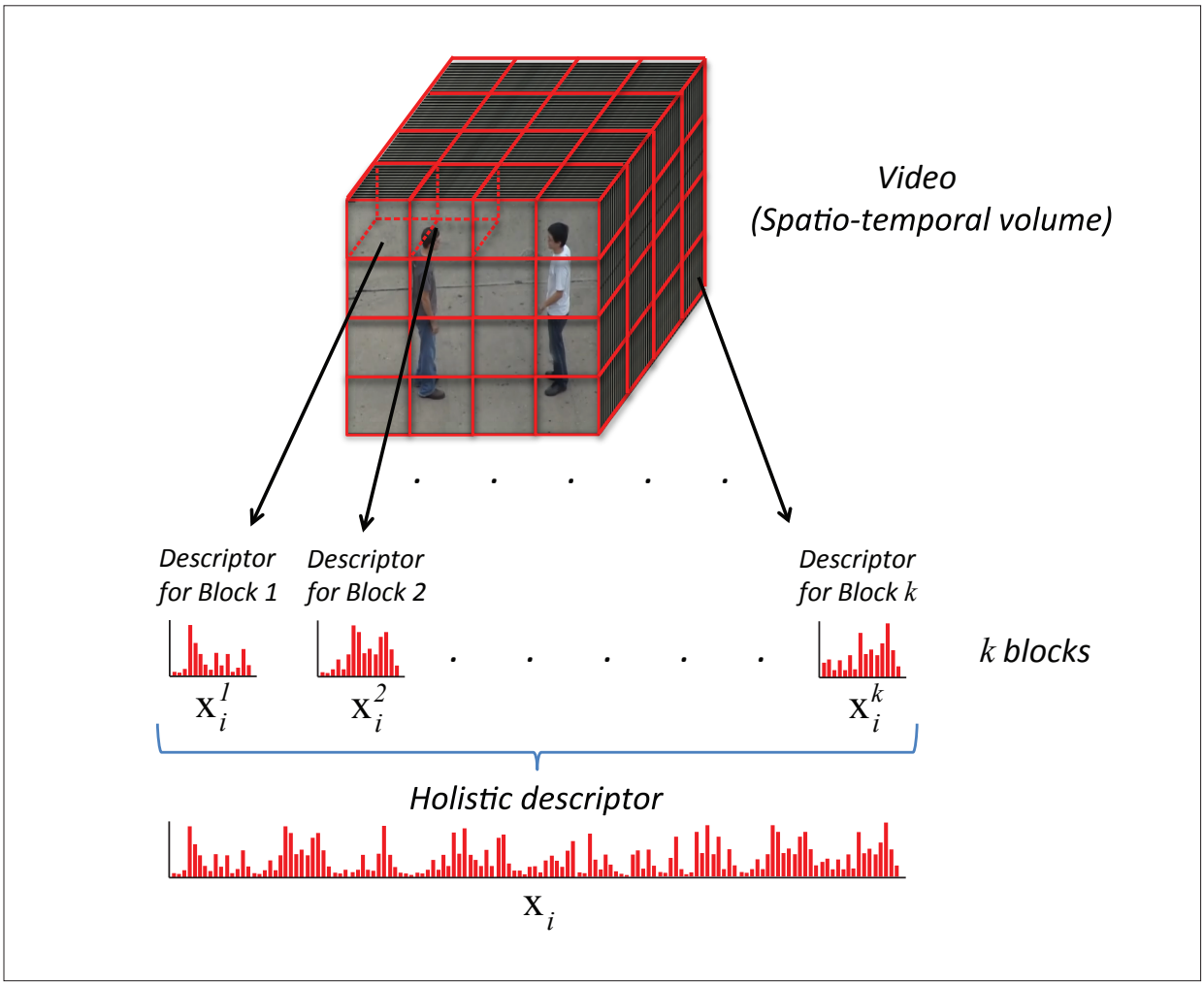

Figure 4.1: The structure of a holistic descriptor $\mathbf{x}_i$, which is a composition of descriptors of k blocks represented by $\mathbf{x}_i^1, \mathbf{x}_i^2, \ldots, \mathbf{x}_i^k$.

4.1 Discriminativity of Blocks

Consider a set of N training descriptors $\mathcal{X} = \{\mathbf{x}_i\}|_{i=1}^N$, $\mathbf{x}_i \in \mathbb{R}^d$, and corresponding N labels $\mathcal{Y} = \{y_i\}|_{i=1}^N$, $y_i \in \mathbb{Z}^1$. In our problem, $\mathbf{x}_i$ is a holistic descriptor for a video with a specific activity, and it is composed by concatenating k blocks $\mathbf{x}_i = [\mathbf{x}_i^1, \mathbf{x}_i^2, \ldots, \mathbf{x}_i^k]^T$. An example is shown in Figure 4.1.

Figure 4.2: Example action classes from the benchmark datasets; a KTH [6], b UT-Interaction [7], c UCF Sports [8], d-e UCF50-101 [9, 10] and f HMDB51 [5] datasets.

For classification, a SVM is trained using $\mathcal{X}$ and $\mathcal{Y}$. In the binary classification case with $y_i \in [-1, 1]$, a linear SVM classification score is computed as

$$s_i = \mathbf{w} \cdot \mathbf{x}_i + \beta, \tag{4.1}$$

where $\mathbf{w}$ is the weight vector of the linear SVM. Similar to the method described in [75], where the detection score is divided among image blocks, we can divide the score among the spatio-temporal blocks, and obtain per-block responses

$$s_i^j = \mathbf{w}^j \cdot \mathbf{x}_i^j + \beta_j, \tag{4.2}$$

where $\mathbf{w}^j$ is the portion of $\mathbf{w}$ corresponding to block j, and β_j is computed similar to [75] and the classification score is the sum of per-block responses

$$s_i = \sum_{j=1}^{k} s_i^j. \tag{4.3}$$

In this chapter, our goal is to find the most discriminative set of blocks and eliminate the blocks which are confusing for classification. To measure the discriminativity of a block, note that for a positive training sample, the per-block score s_i^j is expected to be positive. Similarity, for a negative sample, the score is expected to be negative. Additionally, the magnitude of the score indicates the decision confidence. Therefore, we can express the block discriminativity over the training set as

$$d^j = \sum_{i}^{N} s_i^j y_i. \tag{4.4}$$

Figure 4.3 illustrates the computation of discriminativity for k blocks. The block scores are multiplied with the corresponding label ($y_i \in [-1, 1]$) for each sample in the training set and then summed to obtain the per block discriminativity measure d^j. Note that d^j is highest if all training samples respond correctly within block d^j, and it is penalized whenever a sample responds incorrectly.

For activity recognition, we deal with a multi-class SVM. Typically, a multi-class SVM with c classes has $n = c(c-1)/2$ binary SVM models representing all possible pairs of classes. For instance, for 3 classes, we have three models which are 1 vs 2, 1 vs 3, and 2 vs 3. In order to decide the class of a sample x_i, it is tested against all model pairs, then the class with the maximum votes is selected. To measure the discriminativity in the multi-class scenario, we can consider a matrix D of size k blocks $\times$ n models, with entries d_m^j representing the discriminativity of block j against the binary model m. In this case, the total discriminativity of a block depends on all pairs of models.

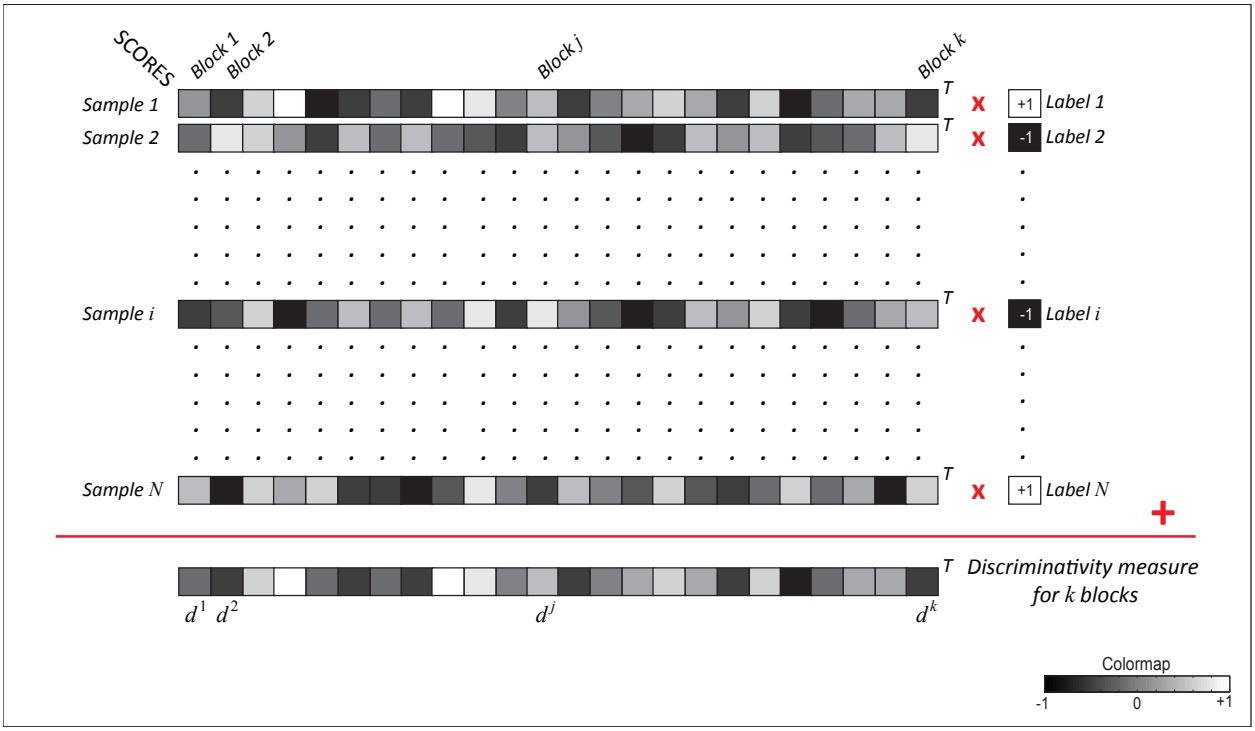

Figure 4.3: The computation of discriminativity of k blocks: For each block, the corresponding scores for the training samples are multiplied with the corresponding labels and then summed to obtain the discriminativity.

The multi-class discriminativity of blocks of a holistic descriptor can be defined based on the average discriminativity measures of blocks over all pairs of models, which is going to be the rows sums of the matrix D, given by

$$d = D\,\mathbf{e_{nx1}}. \tag{4.5}$$

where $\mathbf{e_{nx1}} \in \mathbb{R}^n$ is vector with all entries equal to 1.

The classification performance can be improved by sorting the multi-class discriminativity of blocks and picking a portion of the most discriminative blocks. Figure 4.4 shows the performance of holistic descriptor, GIST3D, when various numbers of blocks are selected considering their multi-class discriminativity on KTH, UCF50 and HMDB51 datasets. When all 1,536 blocks

are selected, the baseline results presented in Chapter 3 are obtained. The performance increases after removing some confusing blocks till a point where the performance decreases sharply in the extreme case where a very high number of blocks are removed.

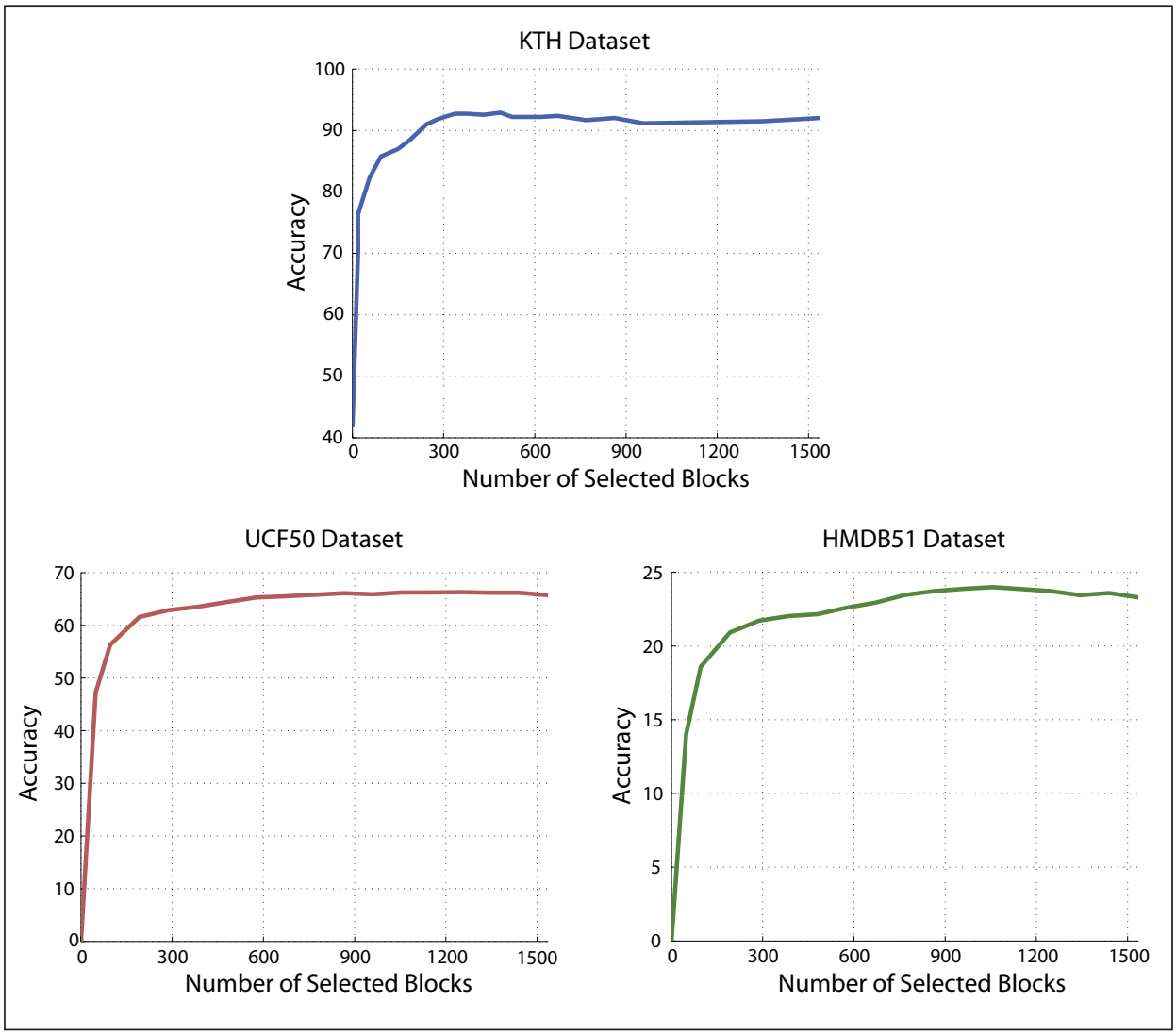

Figure 4.4: Classification performance of GIST3D on the KTH ($blue$), UCF50 (red) and HMDB51 ($green$) datasets by selecting various numbers of blocks based on their discriminativity.

4.2 Optimal Set of Blocks

In the previous section, a portion of the descriptors blocks are selected regarding the global discriminativity for a performance improvement. However, the numbers of selected blocks need to be specified for different datasets. Furthermore, for the multi-class classification scenario, this problem is far from trivial since a certain block which is discriminative for a certain pair of classes could be rather confusing for another pair. In order to address this problem, we introduce a random variable $\mathbf{z} \in \mathbb{R}^k$ of length equal to the number of blocks. $\mathbf{z}$ can be considered as a weight vector for the blocks. The problem of finding the most discriminative set of blocks can be formulated as a minimization problem

$$\mathbf{z}_{disc} = \arg\min_{z} -||D^T\mathbf{z}||_2^2 + \lambda||D^T\mathbf{z} - D^T\mathbf{e_{kx1}}||_2^2 \quad \text{s.t. } ||\mathbf{z}||_0 < \sigma, \tag{4.6}$$

where $\mathbf{e_{kx1}}$ is a k dimensional vector with all entries equal to 1. The first term in equation 4.6 represents the global discriminativity which encourages $\mathbf{z}$ to have higher weights for the blocks which maximizes the total discriminativity, without considering the model of each pair of classes separately. The second term represents the class discriminativity which encourages the obtained solution to maintain the disciminativity of each individual model. Meanwhile, the sparsity constraint guarantees that the selected blocks are sparse, thus eliminating the blocks with minimum discriminativity (the confusing blocks).

Equation (4.6) cannot be directly solved using standard ℓ_1 optimization tools such as [76, 77] since they are typically designed to solve a formulation which does not involve the first term (total discriminativity). Fortunately, it is not difficult to observe that if the matrix D has all positive entries, then the second term (class discriminativity) will also encourage the total discriminativity to be maximized. In our experiments, we noticed that it is very rare that D includes negative

entries. This is because the number of training samples which responds correctly in a certain block is often more than the number of samples which responds incorrectly, resulting in a positive and high discriminativity measure (d_m^j) for the discriminative blocks, while the confusing blocks have low but still mostly positive d_m^j. Therefore, we can safely drop the first term from equation (4.6) and solve it using a standard ℓ_1 optimization method. In all of our experiments, we use the gradient projection method [76], which we found superior in performance and speed. After solving equation (4.6), we drop the blocks corresponding to the non-zero entries in the recovered $\mathbf{z}_{disc}$, and use the remaining discriminative blocks to train a new multi-class SVM. Figures 4.5 and 4.6 visualizes the detected discriminative blocks for UT-Interaction dataset as a vector and in space-time as a 3-D volume. Using these blocks as a mask, Figure 4.7 depicts the overlaid discriminative blocks of this dataset over example videos of 6 action classes.

Note that our method can be contrasted to a dimensionality reduction approach, since the final selected blocks is typically not more 20% of the original blocks. In this context, a traditional dimensionality reduction method (such as PCA [70]) is often performed independently from classification. In contrast, our method is supervised and exploits the same model used for classification in order to detect and eliminate the portions of the descriptors which are confusing for classification.

In a general perspective, it is important to note that, fundamentally, a regular SVM assigns weights to the training samples in order to form a model consisting of the support vectors. However, a SVM cannot weight or eliminate certain blocks of the features (i.e. select certain dimensions). Our method can be thought of as an approach that allows a SVM to not only select the support vectors but also select certain blocks from the features in order to build the best classification model.

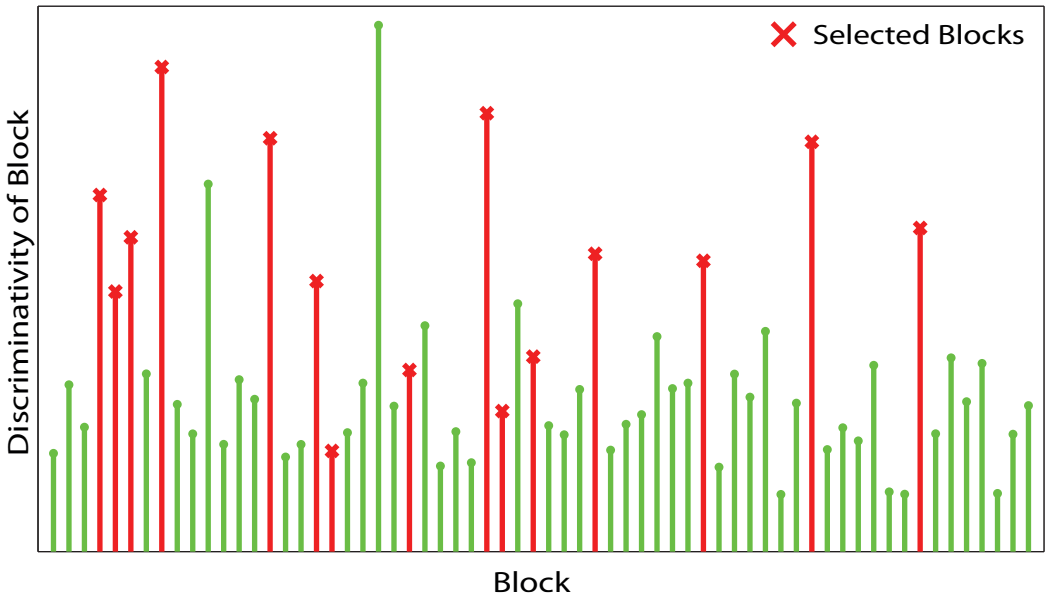

Figure 4.5: Discriminativity of blocks for UT-Interaction dataset: 15 blocks shown in *red* are selected out of a total of 64 blocks. The remaining blocks shown in *green* are not used for the final descriptor generation. The discriminative blocks are selected by the optimization of Equation 4.6. (Please note that the configuration of the selected blocks does not necessarily include all blocks which are individually highly discriminative but it maximizes the total classifier discriminativity)

4.3 Experimental Results

We extensively experimented on the proposed ideas using six benchmark datasets including: UT-Interaction [7], KTH Action [6], UCF Sports Action [8], UCF50 [9], UCF101 [10], and HMDB51 [5]. Example action classes are shown in Figure 4.2. Based on our method, the discriminative blocks can be discovered for any block-based feature representation (i.e. a feature composed of a group of meaningful blocks). Please note that our method detects a set of blocks which are discriminative over whole dataset, hence in some scenarios some of the blocks which are highly discriminative individually may still not be selected if they are not improving the overall classification performance.

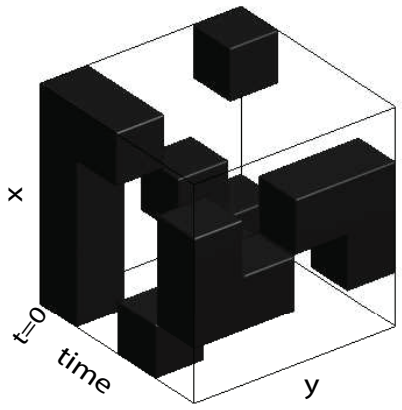

Figure 4.6: Detected discriminative blocks in 3-D for UT-Interaction dataset: The masked spatio-temporal blocks are shown in *black*. The final descriptors are computed only on the selected blocks.

In our experiments, we used three types of descriptors. First descriptor is the three dimensional histograms of oriented gradients (HOG3D) [2]. In that, for each video clip the frames are stacked over time and the obtained spatio-temporal volume is divided into spatio-temporal blocks, and HOG3D is computed for each block. Consequently, the histograms of the blocks are concatenated for generating a holistic descriptor. In the experiments, for HOG3D, we downsampled spatio-temporal volume of each video clip to $128 \times\ 128 \times 128$ by using linear interpolation. We divide the videos into 8 cells with size $32 \times 32 \times 32$ and each cell into 8 blocks with size $16 \times 16 \times 16$. For each spatio-temporal volume we compute the gradients along horizontal, vertical and temporal dimensions separately using the central difference mask $[1 \quad 0 \quad -1]/2$ and quantized the three dimensional gradient orientations into 12-bins for both azimuthal and elevation angles. In most of the cases, the the two orientation angles do not exactly correspond to one of the 12-bins, therefore we split the gradient among the 4 neighbor bins where weight of each bin is assigned based on its distance to the gradient orientation. This setting resulted in a 144-dimensional histogram for each block and hence a 9,216-dimensional descriptor for each video.

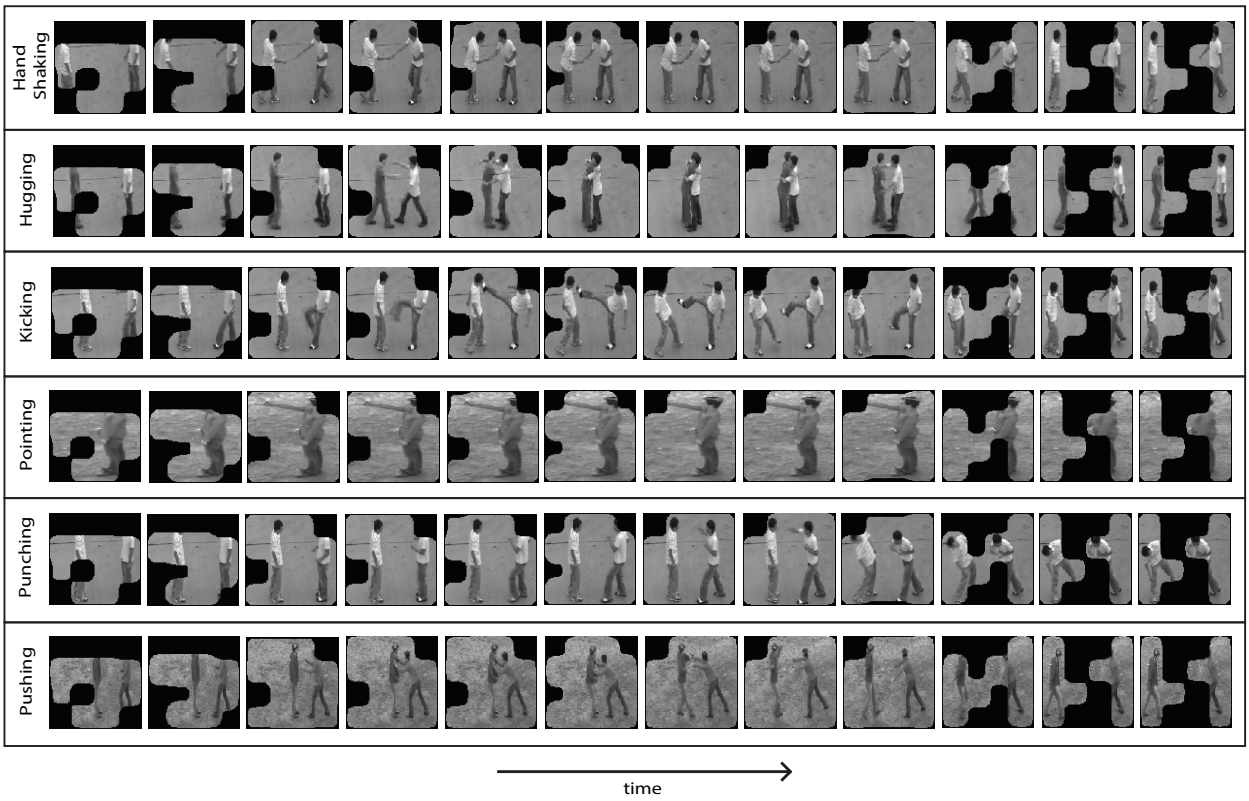

Figure 4.7: Examples of the selected blocks for UT-Interaction dataset. Each row shows sample frames of a certain action video.

Second, we used the GIST3D which is the frequency spectrum based holistic descriptor presented in Chapter 3. This descriptor consists of 512 spatio-temporal blocks with size $16\times16\times16$ for each clip of a video. For each block the length of individual descriptors is 68 which is the average of responses to 68 filters. Since we extracted three clips per video, there are 1,536 blocks for this descriptor. The final descriptor is the concatenation of individual descriptor blocks and is a 104,448-dimensional vector for each video.

Third, we use the action bank features [15]. In that, we use a bank of action detectors, where each action detector is a template which is used to filter the video clip and generate a correlation volume. Consequently, max-pooling is employed as in [15] using three levels in the octree.

This generates a 73-dimensional vector per action detector. The final video feature vector is a concatenation of the features from the different action detectors. Note that in the action bank features, each action detector from the bank corresponds to a block; therefore, in these features, our method detects the discriminative detectors and eliminates the confusing ones. We use a bank of 205 action templates (blocks) in all experiments, similar to [15]. In all experiments, we followed the evaluation protocol set forth by the authors [7, 6, 8, 9, 10, 5]. Please note that we refer to our method as OSB in all tables.

4.3.1 UT-Interaction Dataset

The UT-Interaction dataset includes videos of 6 classes of human-human interactions as in 4.2. Each class includes 20 videos making a total of 120 videos. We computed holistic HOG3D features on both space-time and spatio-temporal frequency volumes. The discriminativity of the blocks is shown in Figure 4.5. The final selected blocks are shown in *red* in Figures 4.5 and 4.6. In Figure 4.7 we observe that this configuration of blocks is able to capture the salient regions of the scenes among all action classes. After the descriptors of the selected blocks are computed, the principal component analysis [70] is used for dimensionality reduction on the final feature vectors. Additionally, we combined the two descriptors on time and frequency domain by concatenating the feature vectors. The classification accuracies after 10-fold cross validation are shown in Table 4.1.

4.3.2 KTH Dataset

KTH dataset contains videos of six types of human actions performed 4 times by 25 actors in outdoor and indoor settings. Using this dataset as a benchmark, we compute the action bank features and we select the sparse and highly discriminative blocks shown in Figure 4.8. We compare the performance of our approach to the original method in [15]. For training and testing, we used the original splits as in [6].

Table 4.1: Classification accuracies on UT-Interaction dataset.

Method	Accuracy
HOG3D TD	80.0
HOG3D FD	74.1
HOG3D TD+FD	82.5
Perez et al [78]	84
Yuan et al [79]	86
Team BIWI [80]	88
HOG3D TD+OSB	**86.8**
HOG3D FD+OSB	**78.3**
HOG3D TD+FD+OSB	**89.2**

Table 4.2: Classification accuracies on KTH dataset.

Method	Accuracy
Schuldt et al [6]	71.7
Klaser et al [2]	84.3
Ryoo and Aggarwal [81]	91.1
Laptev et al [11]	91.8
GIST3D	92.0
Le et al [82]	93.9
GIST3D+OSB	**94.5**
Gilbert et al [74]	94.5
Action Bank et al [15]	98.2
Action Bank+OSB	**99.5**

As depicted in Table 4.2, applying our method has a classification accuracy of 99.5%, which is an improvement over the state-of-the-art [15]. Figure 4.9 is the confusion table among the six action classes. In addition, we performed experiments by applying our method on the presented GIST3D descriptor, and obtained a 94.3% classification accuracy.

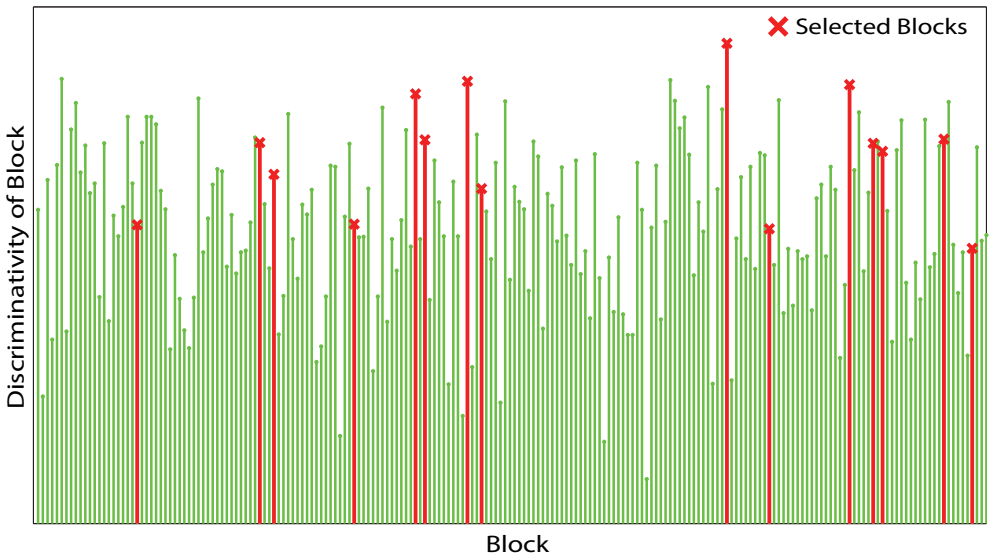

Figure 4.8: Discriminativity of blocks for KTH dataset: the blocks shown in *red* are selected for the final descriptor generation.

4.3.3 UCF Sports Action Dataset

UCF Sports Actions Dataset contains videos of humans performing various sports activities obtained from a wide range of online sources. The 10 action classes of this dataset are shown in Figure 4.2.c. On this dataset, we computed the HOG3D and action bank features similar to the other experimental settings. We applied the standard leave-one-out cross validation strategy as used in [83]. Using each of these feature representations we achieved a significant performance improvement through our approach. The selected sparse and highly discriminative blocks are shown in red color in Figure 4.10. The results are illustrated in Table 4.3 and Figure 4.11. Please note that the result of action bank method is not exactly the same as the result reported in [15] due to the variations of the experimental settings. However, our method improves the performance of the original features consistently using any experimental settings.

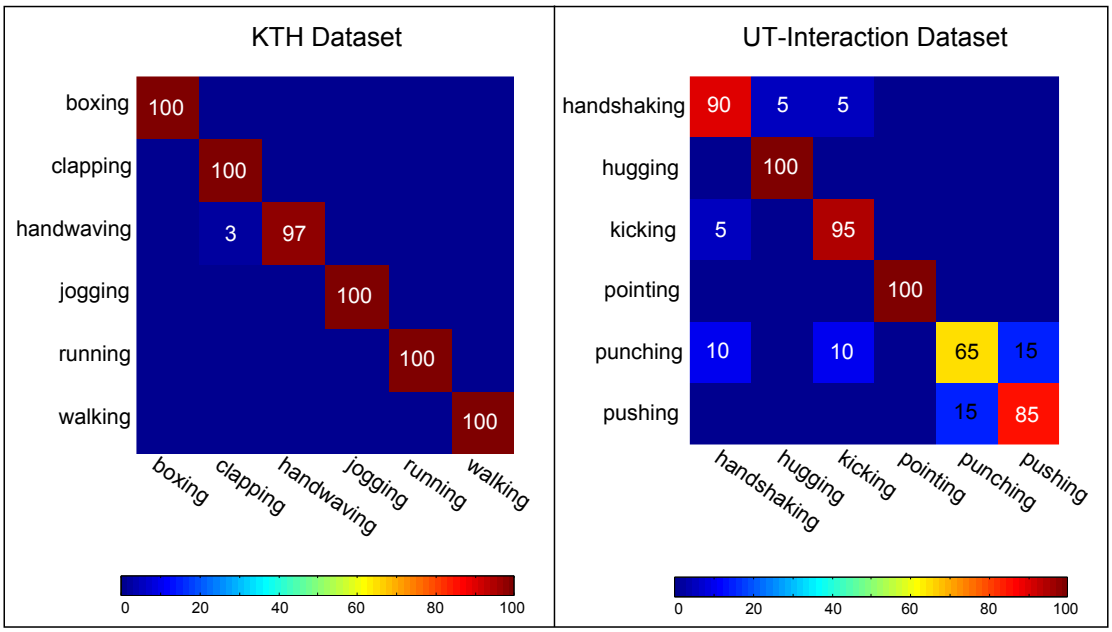

Figure 4.9: Confusion tables for KTH and UT-Interaction datasets: both datasets have six action classes.

Table 4.3: Classification accuracies on UCF Sports Action dataset.

Method	Accuracy
HOG3D TD+FD	77.3
HOG3D TD+FD+OSB	**81.3**
Kovashka and Grauman [84]	87.3
Wu et al [85]	91.3
Action Bank [15]	92.1
Action Bank+OSB	**95.0**

4.3.4 HMDB51 Dataset

HMDB51 is a collection of videos from various sources which includes 51 action categories, each containing a minimum of 101 videos. On this dataset we used action bank features [15] and GIST3D for video representation. We performed cross validation on the three split sets of the

dataset, which is provided by [5]. Using a sparse set of action banks, our method was able to improve the performance of [15] and GIST3D as shown in Table 4.4.

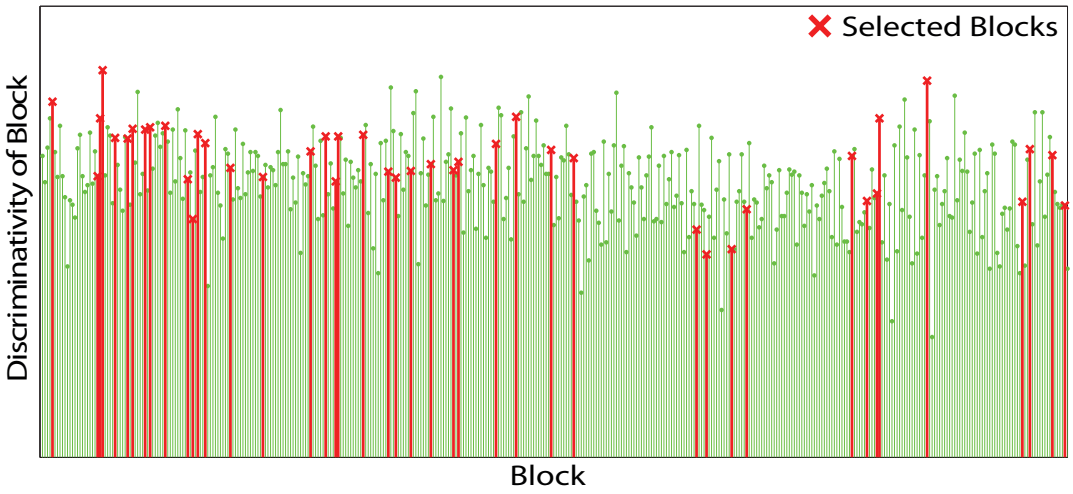

Figure 4.10: Discriminativity of blocks for UCF Sports Action dataset: the blocks shown in *red* are selected for the final descriptor generation.

Table 4.4: Classification accuracies on HMDB51 dataset.

Method	Accuracy
Gist [68]	13.4
Laptev et al [16]	20.2
C2 [5]	23.2
GIST3D	23.3
GIST3D+OSB	**24.5**
Action Bank [15]	26.9
Action Bank+OSB	**27.2**

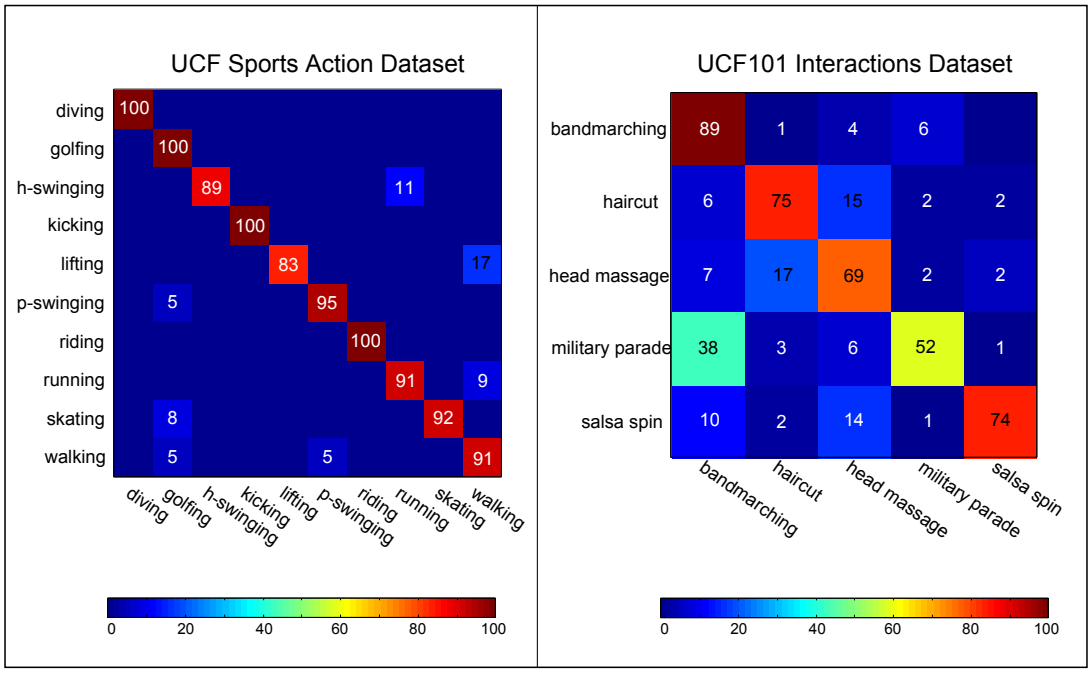

Figure 4.11: Confusion table for UCF Sports Action and UCF101-Interactions dataset which contain 10 and 5 action classes, respectively.

4.3.5 UCF50 and UCF101 Datasets for Human Actions

UCF50 and UCF101 are datasets of realistic videos which are collected from online sources. These datasets are composed of 25 groups which are designed for group-wise cross validation. Each group contains several clips performed by the same actor. In each experiment, we left one group for testing while the remaining groups are used for training the classifier. Table 4.5 demonstrates the performance gain of our method on various settings. We also tested our approach on UCF101 Interactions dataset, which is a sub-portion of UCF101 including 5 classes of human-human interactions. On UCF101 Interactions and UCF101, we achieved performance of 73.9% and 48.3% average classification accuracies applying our method. The average recognition performance is shown on the confusion matrix in Figure 4.14.

Table 4.5: Classification accuracies on UCF human action datasets.

Method	Dataset	Accuracy
GIST3D	UCF50	65.3
GIST3D+OSB	UCF50	**68.9**
HOG3D TD	UCF101 Int.	65.0
HOG3D FD	UCF101 Int.	65.9
HOG3D TD+FD	UCF101 Int.	67.8
HOG3D FD+OSB	UCF101 Int.	**68.9**
HOG3D TD+OSB	UCF101 Int.	**67.6**
HOG3D TD+FD+OSB	UCF101 Int.	**73.9**
HOG3D TD	UCF101	36.3
Soomro et al [10]	UCF101	44.5
HOG3D TD+FD+OSB	UCF101	**48.3**

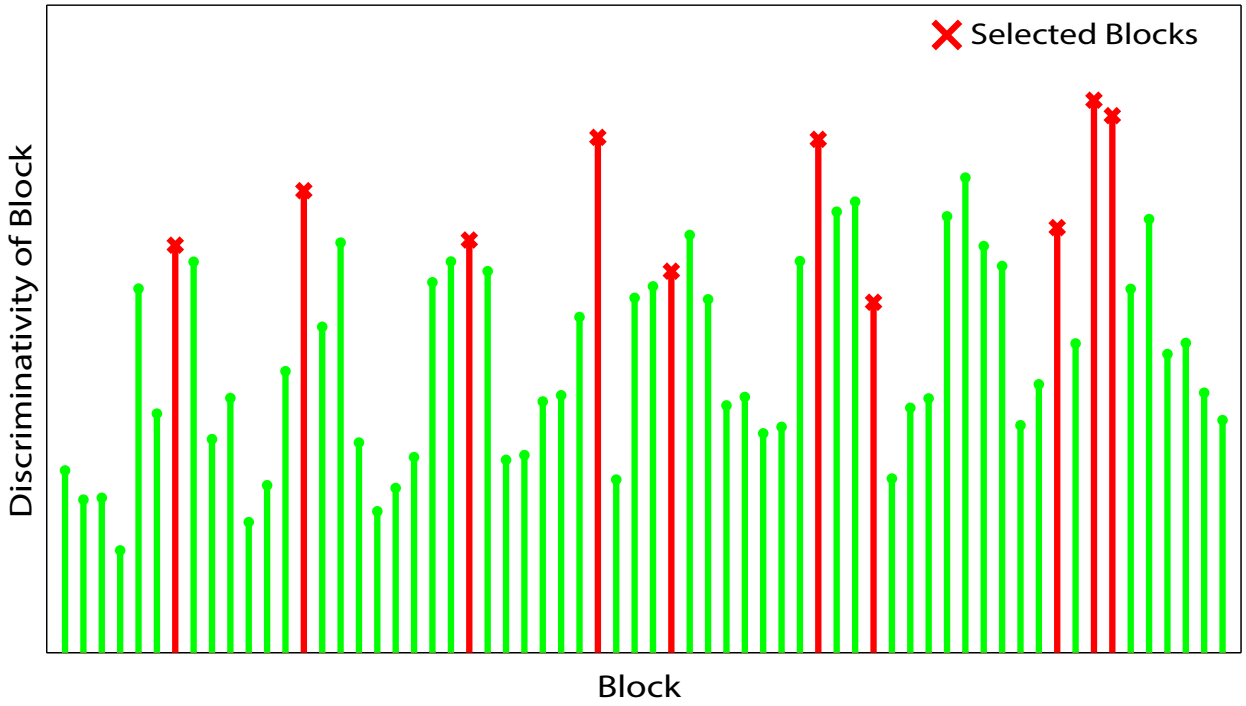

Figure 4.12: Discriminativity of blocks for UCF101-Interaction dataset: the blocks shown in *red* are selected for the final descriptor generation.

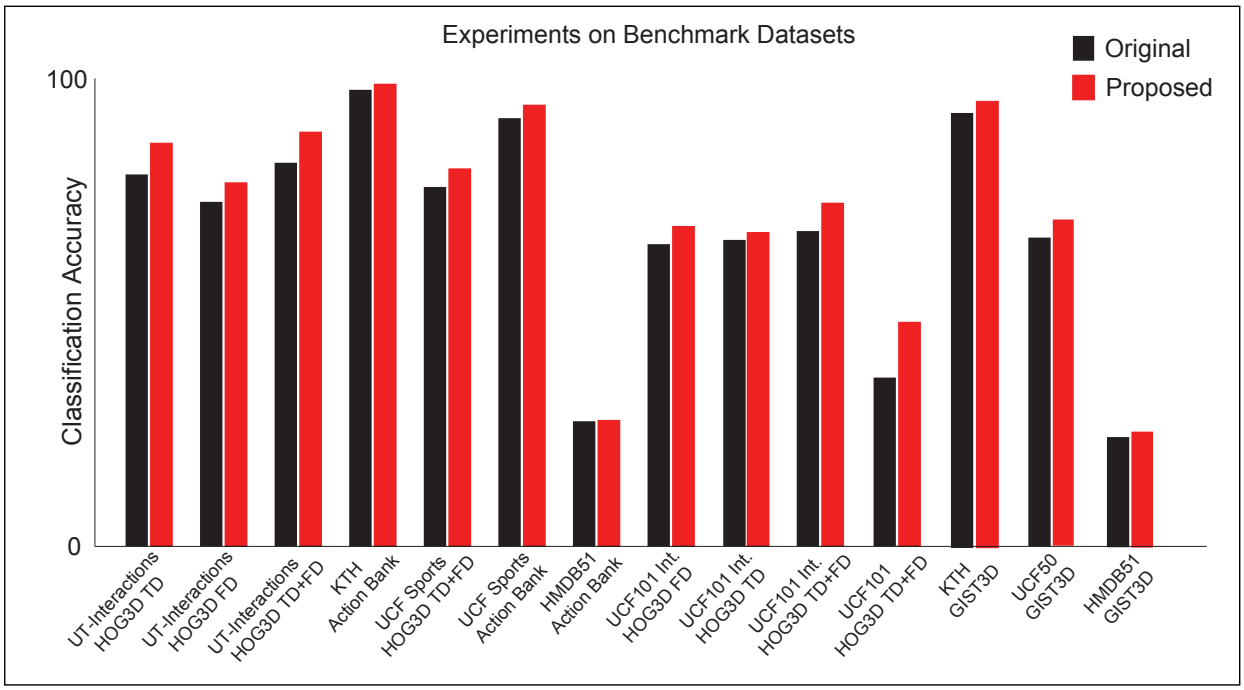

Figure 4.13: Performance gain over benchmark datasets.

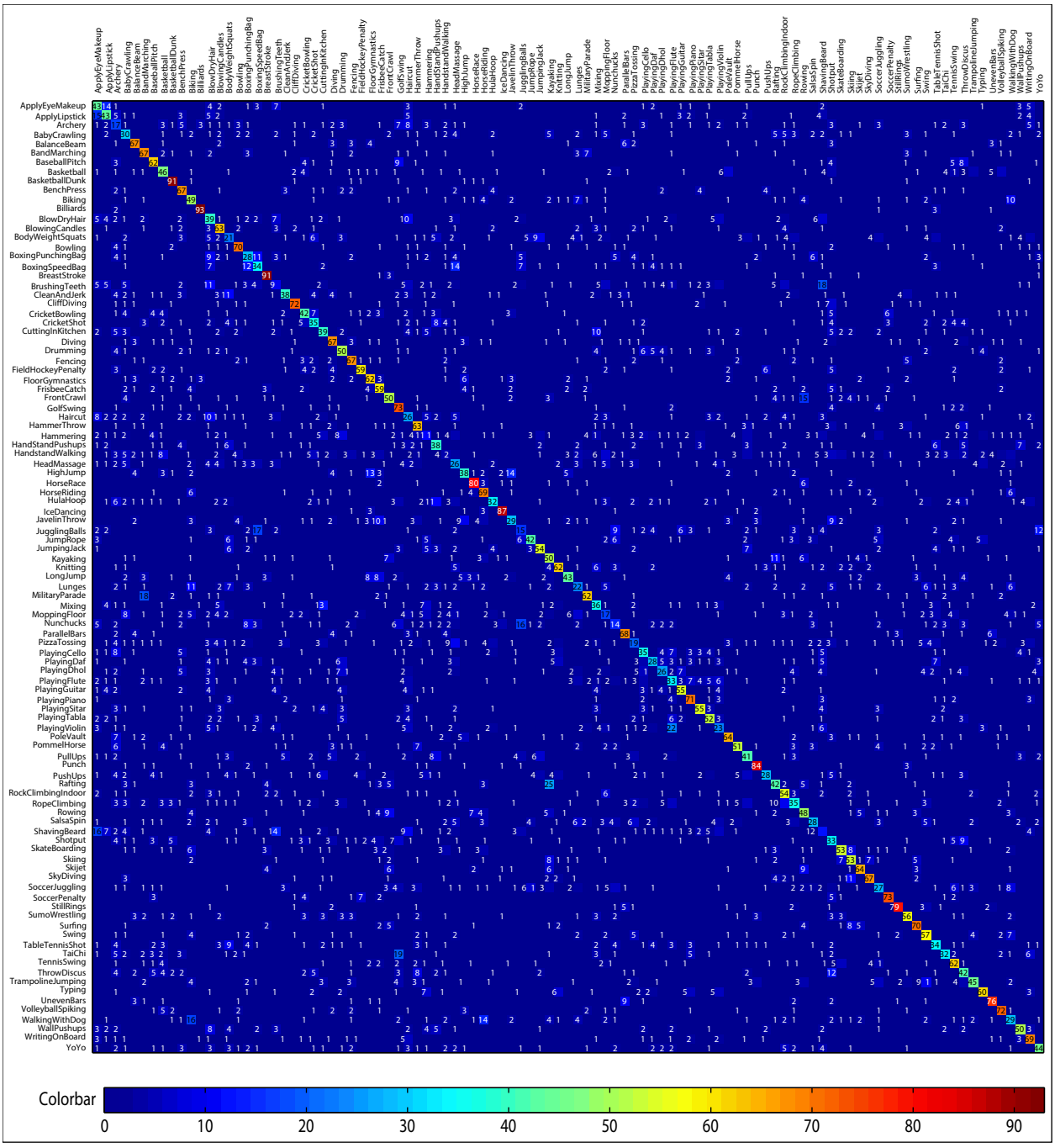

Figure 4.14: Confusion table for 101 action classes of UCF101 Actions dataset.

4.4 Summary

The holistic descriptors are composed of individual descriptor blocks which may be redundant or confusing for the classifier. In this chapter, we presented a new method to improve the classification performance of holistic descriptors by selecting an optimal set of discriminative blocks of the descriptor. The selection of blocks was formulated to minimize a function of individual and global discriminativities, which ensures the sparsity of blocks and a high classification performance. We computed the descriptors only for the selected blocks and demonstrated the performance gain of our approach over the tested features, as shown in Figure 4.13. Our method ameliorated the performance of each tested holistic descriptor as well as inherently accomplishing a dimensionality reduction and a computational speed up.

CHAPTER 5: IDENTIFYING CROWD BEHAVIORS BY STABILITY ANALYSIS FOR DYNAMICAL SYSTEMS

In last two chapters we focused on actions performed by a single person. In this chapter, we focus on video sequences of crowd scenes which provide challenging problems to the world of computer vision. The approaches that are presented in the literature for recognizing the actions of individuals are not applicable to crowd scenes since high densities of objects in real-world situations make individual object recognition and tracking impractical. The challenge is to know and understand behaviors in a crowd without knowing the actions of individuals, and the challenge becomes increasingly stiff as real-world conditions take on a variety of characteristics, which could easily be mistaken. Nevertheless, the problem remains an important one, as automated detection of crowd behaviors has numerous applications. Some examples include prediction of congested areas, which may help avoid unnecessary crowding or clogging, and discovery of any abnormal behaviors or flow, which may help avoid tragic incidents such as a stampede.

In this particular work, our goal is to devise an algorithm that identifies five common and specific crowd behaviors in visual scenes. (1) Bottlenecks occur when many pedestrians/vehicles from various points in the scene enter through one narrow passage. (2) Fountainheads occur when many pedestrians/vehicles emerge from a narrow passage only to separate in many directions. (3) Lanes are formed when many pedestrians/vehicles are moving at same speeds in the same direction. (4) Arches are formed when the collective motion is curved or circular. (5) Blocking occurs when there is opposing motion and the desired movement of groups of pedestrians is prohibited. Naturally, this provides a way to detect changes from one behavior to another, which may occur in panic situations or in overly crowded ares.

Our approach views the optical flow in a scene as a dynamical system. In the algorithm, we overlay a grid of particles on a scene initializing the dynamical system. Time integration of the

dynamical system provides the particle trajectories that represent the motion in the scene; these trajectories are used to locate regions of interest (ROI) in the scene that are candidates for the types of behavior we seek. Linear approximation of the dynamical system provides behavior classification through the Jacobian matrix; the eigenvalues determine the dynamic stability of points in the flow and each type of stability corresponds to one of the five crowd behaviors. The eigenvalues are only considered in the regions of interest, consistent with the linear approximation and the implicated behaviors. The algorithm is repeated over sequential clips of a video in order to record changes in eigenvalues, which may imply changes in behavior. The method was tested on over 60 crowd and traffic videos.

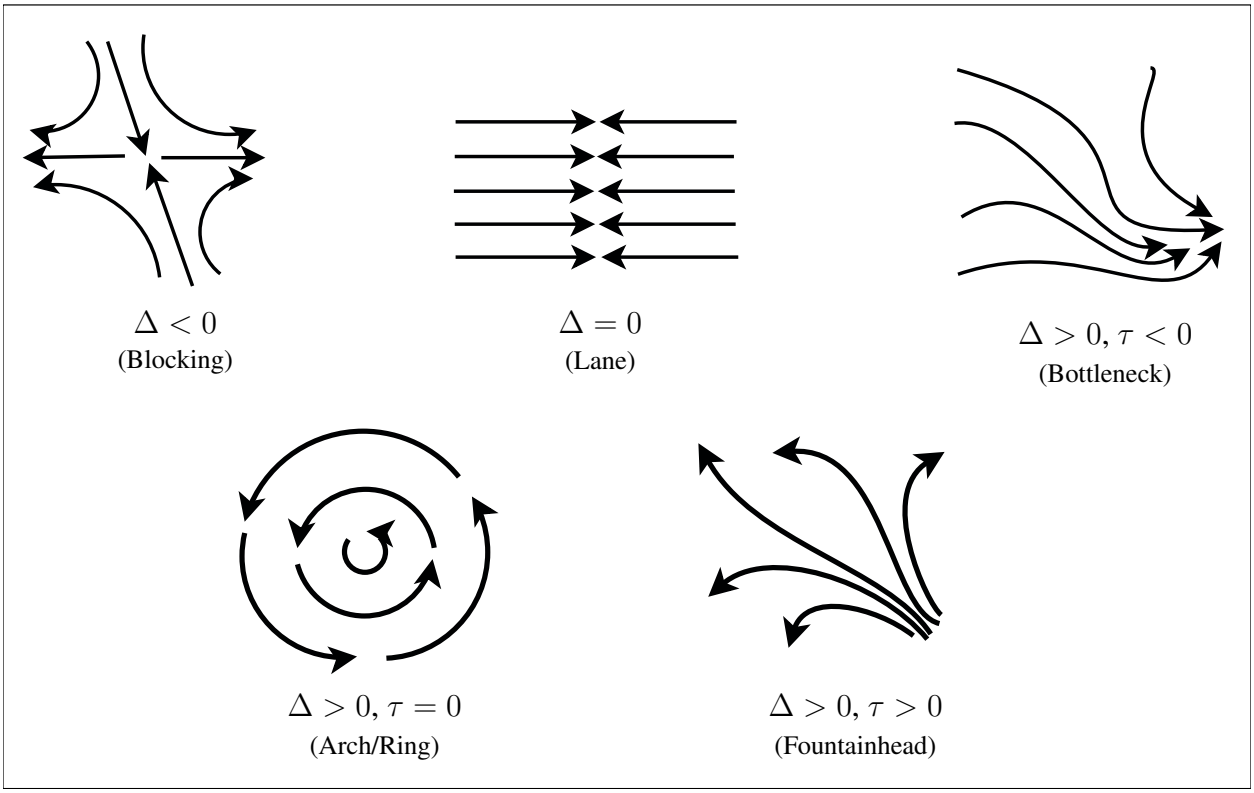

Figure 5.1: Five flows corresponding to Δ and τ, along with the related crowd behaviors

5.1 Stability in Dynamical Systems

Consider a continuous dynamical system

$$\dot{w} = F(w), \tag{5.1}$$

relating particle velocities and positions, where we define

$$w(t) = [x(t), y(t)]^T \quad \text{and} \quad F(w) = [u(w), v(w)]^T. \tag{5.2}$$

Here, x and y are particle positions, and u and v represent particle velocities in the x and y directions, respectively. (In our application to video sequences, u and v are obtained from optical flow.) A first step toward understanding the solution behavior for (5.1) is finding the critical points w^* which satisfy $F(w^*) = 0$. Then we can determine the behavior of trajectories near these critical points by considering a linearization of the system about the critical point. To find a linearization (see, for example, [86]) let $z = w - w^*$, which means

$$\dot{z} = \dot{w} = F(w) = F(w^* + z). \tag{5.3}$$

By Taylor's theorem

$$F(w^* + z) = F(w^*) + J_F(w^*)z + \mathcal{O}(z^2), \tag{5.4}$$

where J_F denotes the Jacobian matrix for the function F,

$$J_F = \begin{pmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} \end{pmatrix}, \tag{5.5}$$

Since $F(w^*) = 0$ we have a linear approximation of (5.1) near the critical point w^*, given by

$$\dot{z} = J_F(w^*)z. \tag{5.6}$$

The solutions of the system (5.6) are completely defined by the initial conditions and the eigenvalues of the matrix J_F. The eigenvalues of a 2×2 matrix are solutions of a characteristic equation

$$\lambda^2 - \tau\lambda + \Delta = 0, \tag{5.7}$$

where τ is the trace and Δ is the determinant of the matrix. It is easy to show that

$$\lambda_{1,2} = \frac{1}{2}\left(\tau \pm \sqrt{\tau^2 - 4\Delta}\right) \tag{5.8}$$

with

$$\Delta = \lambda_1\lambda_2 \quad \text{and} \quad \tau = \lambda_1 + \lambda_2, \tag{5.9}$$

where λ_1 and λ_2 are the eigenvalues, yielding important information about the type of flow, as depicted in Figure 5.1.

$\Delta < 0$ implies the critical point w^* is a saddle, meaning that the particle trajectories are pulled toward the point in two directions, but pushed away in other directions.

$\Delta = 0$ implies at least one eigenvalue is zero, and the critical point w^* is non-isolated.

$\Delta > 0$ implies the eigenvalues are real and have the same sign, or they are complex conjugates. If $\tau < 0$, then the critical point w^* is stable and it acts as a sink for nearby particle trajectories. If $\tau > 0$, then the critical point is unstable and it acts as a source for nearby particle trajectories. If the eigenvalues are purely imaginary complex conjugates, so that ($\tau = 0$), then the critical point is a center, and near-by trajectories orbit the point indefinitely.

It is very important to notice that the overall flow in a crowd scene can not be expected to conform to the strict definition of the dynamical system (5.6), with global flow patterns depicted in Figure 5.1. This may be understood by noting that there is not one global function F that defines the entire flow, but instead each particle has a function F defining its motion. As a result, we can only expect the flow patterns of Figure 5.1 to represent the crowd flow locally, but there are global aspects of crowd flow that may be recognized by such a comparison.

5.2 Behavior in Crowd Scenes

A key factor of the analysis described in Section 5.1 is location of a critical point. With regard to video scene analysis, we need to find ROI, which locally correspond to critical points, and check properties of the Jacobian matrix at points in these regions. Locating ROI may be accomplished by numerically solving the equations of motion (5.1), the details of which are left for Section 5.3. It is possible that not all ROI have real significance for understanding the behaviors in the scene, but this can be determined through the Jacobian matrix. We now consider various crowd behaviors and their links to the Jacobian matrix. Now consider the flows arising from J_F, as depicted in Figure 5.1, in connection with specific crowd behaviors.

5.2.1 Bottlenecks

As illustrated in Figure 5.1, $\Delta > 0$ and $\tau < 0$ is a clear representation of particle trajectories from many points that converge to one location. This is evident of crowd behaviors in which many pedestrians or vehicles from various points in the scene enter through one narrow passage. Hence, we define a *bottleneck* to be the mouth of any narrow passage through which pedestrians regularly pass. This liberal definition allows consideration of many flows. However, it makes no distinction between bottlenecks that occur in normal situations and those that result in clogging, typical in panic situations when many pedestrians simultaneously try to exit through one narrow

passage. Though the present framework does not allow for easy detection of such behavior in a strict panic situation, it may identify clogged bottlenecks as a special case of the blocking behavior. We describe the case of a simple bottleneck here.

Our general approach for detecting a bottleneck is twofold. First, we find regions of the scene that have potential to be a bottleneck. This is achieved by locating areas in which many particles tend to accumulate over time, and we label the centroid of such regions as a candidate point for a bottleneck, if the particles arrived at that point from many different locations in the scene. Second, we check the eigenvalues of J_F in the vicinity of each candidate point. If $\Delta > 0$ and $\tau < 0$, then the region immediately surrounding that point can be labeled as a bottleneck.

5.2.2 Fountainheads

In the case of $\Delta > 0$ and $\tau > 0$, as shown in Figure 5.1, particle trajectories diverge from a particular region in the scene. This is representative of crowd behavior in which pedestrians leave a narrow passage and persist in many separate directions, and we define the mouth of such a passage as a *fountainhead*. This behavior can essentially be thought of as the opposite of a bottleneck, hence, fountainheads are detected simply as bottlenecks in backward time.

5.2.3 Lane Formation

In crowded situations, lanes of flow in opposite directions naturally form, as pedestrians moving against the flow step aside to avoid collision and end up moving with other pedestrians with the same basic direction and generally the same speed. In such instances, the motion near an individual appears to be nil, relative to other nearby individuals, because they are all moving together. This is precisely the type of behavior we see in what we define as a *lane*, and the behavior is well-described by non-isolated critical points, rendering $\Delta = 0$ along the path of the lane. Clearly, $\Delta = 0$ if the objects in the scene are stationary and the optical flow is zero, but we are not interested in this trivial case. Thus, we distinguish this case from the case in which many

pedestrians or vehicles are moving at the same speeds in the same direction (a straight line). In addition, it should be noted that a single object moving in a unique direction is not considered a lane.

5.2.4 Arch/Ring Formation

The motion described by $\Delta > 0$ and $\tau = 0$, shown in Figure 5.1, is characteristic of crowd motion that is curved or circular. This behavior may be typical of a crowded scene in which pedestrians must maneuver around obstacles, forming an *arch*. It may also be observed in less typical scenes such as people dancing or traffic in a round-about, forming a *ring*. In either case, the eigenvalues of the Jacobian matrix are purely imaginary complex conjugates, and we look for this eigenvalue response along oblique paths over which many trajectories may pass.

5.2.5 Blocking

As demonstrated in Figure 5.1, $\Delta < 0$ represents local flows in which particles are bouncing off of each other in somewhat random directions, unable to proceed in the direction originally desired. This is characteristic behavior of people in densely populated scenes where the surrounding crowd prevents the desired motion of many individuals, and we define this behavior as *blocking*. In particular, pedestrians moving in opposite directions may block each other as crowd density increases, preventing any advancement from either group. In some situations the density of the crowd may actually lead to total gridlock and no motion of particles, in which case the optical flow is zero. These instances can still be recognized as blocking because they are always preceded by some type of regular flow. In other words, regions with regular movement in a high density crowd that suddenly become void of motion are best explained by blocking.

5.2.6 *Changes in Behavior*

Anytime we are able to detect a regular pattern of motion, such as lanes or bottlenecks, we can easily find changes in the behavior when there is a change in the eigenvalue response. Such an event is known as a bifurcation in the study of dynamical systems, and it can result in sudden, sometimes drastic, changes in the flow. More specifically, bottlenecks that move from free-flowing to clogged experience a bifurcation. Other examples include cases when lanes are blocked by other opposing lanes as a result of increased density in the flow, or when the flow generally tends toward a point, but people suddenly run away from that spot to avoid danger.

5.3 Implementation

A key factor of the analysis described in Section 5.1 is location of a critical point. In video scene analysis, we locate, instead, a *region of interest* (ROI), which locally corresponds to a critical point, and check the eigenvalues of J_F at points in these regions. For multiple behaviors in a scene we have multiple ROI. It is possible that not all ROI have significance for understanding the behaviors in the scene, but this can be determined through J_F. This framework constitutes two main tasks, which may be executed in parallel and are summarized by the following computations.
-*Regions of interest* (ROI) are defined according to the behaviors we observe, and thereby consist of one of the three following locations.

- Candidate Points for Bottlenecks/Fountainheads
- Candidate Paths for Lanes and Arches/Rings
- Candidate Precincts for Blocking

-*Eigenvalue maps* are defined by the signs of Δ and τ for each point in a ROI.

Since our method is based on advection of particles by (5.1), defined by the optical flow, the implementation begins with computation of optical flow. We use Lucas-Kanade optical flow [87]

and since optical flow corresponding to some individuals may be different than the general crowd flow, we apply median filter. (For an image of 360x480 typical size of median filter is 40x40.) In order to detect type of flow observed in a video clip and the various crowd behaviors, using the dynamical systems framework of Section 5.1, we must first identify ROI in the scene, and then evaluate the eigenvalues of J_F at these points. For multiple behaviors in a scene we have multiple ROI. A flow chart describing implementation of our method is depicted in Figure 5.2, showing computation of ROI and eigenvalue maps from the optical flow.

It is essential to complete both tasks, because identifying the given behaviors is not possible using the particle trajectories alone, and the eigenvalues loose their significance without the ROI, which correspond to critical points of the dynamical system. Details on implementation are provided in Section 5.3.1 and 5.3.2, respectively, and the implementation process is fully depicted in Figures 5.3, 5.4, and 5.5.

Notice, our definition of a ROI is not equivalent to the definition of a critical point as given by dynamical systems theory and given by w^* in Section 5.1. This is because we do not want to consider *all* points where the optical flow is zero. Instead we consider points that are affected by the flow in ways that are consistent with the behaviors considered, and this is consistent with our set-up, as the system (5.6) can only be used to give local information about the flow. To be more specific, $F(w^*) = 0$ is nearly satisfied in 1) regions with blocking, as the motion of the individuals is prohibited, 2) bottleneck and fountainhead regions, as those points act as sinks or sources for the flow, and 3) the paths of lanes, as the motion of an individual is nearly zero relative to other near-by individuals.

To locate changes in behavior, the total sequence is divided into clips of fixed length, and the algorithm is repeated. Comparing sequential clips and recording appropriate changes in the eigenvalues reveals changes in behavior. This is demonstrated in Figure 5.5.

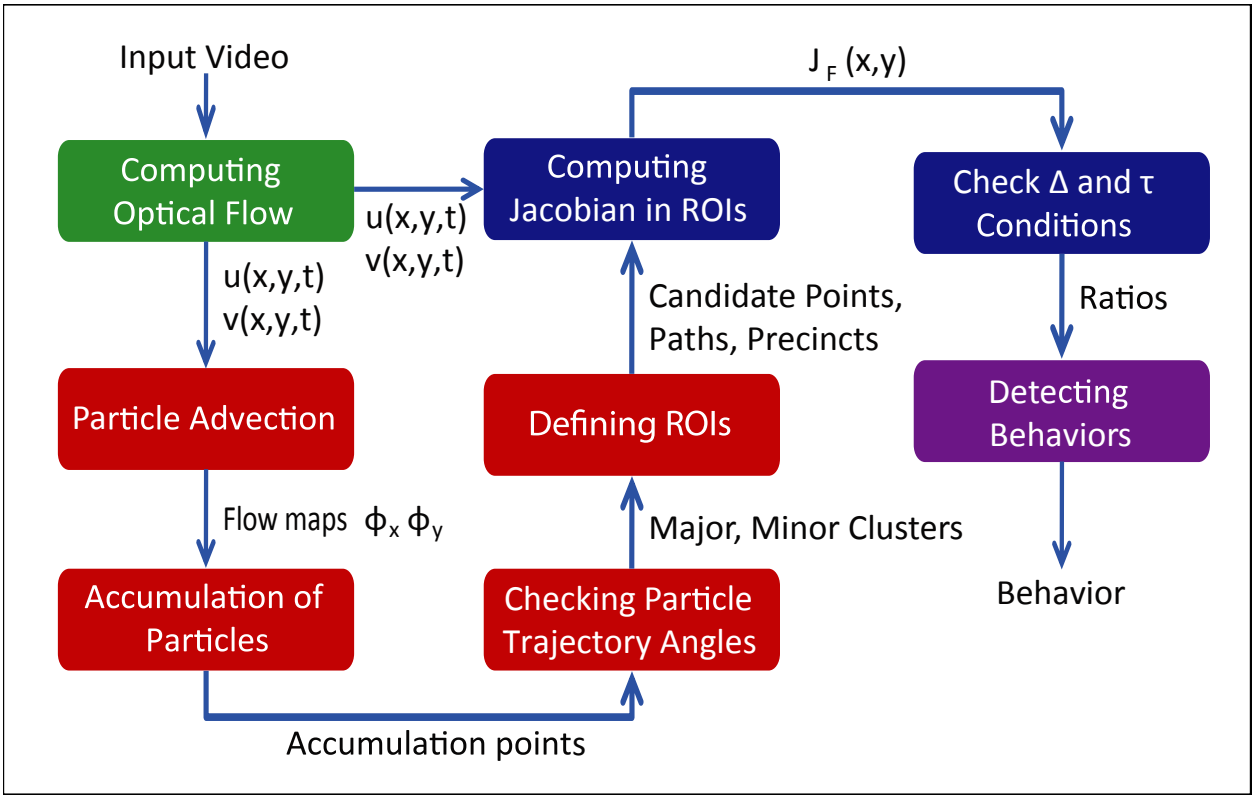

Figure 5.2: Overview of the Framework

5.3.1 Regions of Interest

Regions of interest (ROI) are necessary for finding possible locations of scene behaviors, for removing false positives, and for reducing the amount of computations. ROI are defined according to the behaviors we observe, and thereby consist of one of the three following locations; candidate points, candidate paths and candidate precincts. We describe computation of the these three types of ROI, but common to computation of each ROI is particle advection and the resulting accumulation of particles, which is explained first.

Particle Advection. A grid of particles is overlaid on the initial frame and advected with the flow. Numerically solving the system of equations (5.1), using

$$w(t+1) = w(t) + F(w(t)), \tag{5.10}$$

yields the particle positions over the time interval $[t_0, t_f]$. The evolution of particles through the flow is tracked using particle flow maps

$$\phi_{t_0}^{t}(w) = w(t; t_0, w_0), \tag{5.11}$$

which simply indicate the relation between the initial particle positions and their positions at time $t \in [t_0, t_f]$. We use $P(\phi_{t_0}^{t}(w))$ to denote the particle corresponding to a particular flow map. The flow maps are initialized in the first frame of the video. As the particles evolve through the flow field in time, they are accumulated in particular regions of the scene.

Accumulation Points. Accumulation of particles may occur at a bottleneck or at the end of a lane and results in higher particle densities in these regions. The particle densities of the frame are calculated using the flow maps. More precisely, if $D(w_f)$ denotes the density map for particles at position w_f, then it is equal to the cardinality of the set containing all particles at that point, i.e. if

$$\mathcal{P}_{w_f} = \{P(\phi_{t_0}^{t_f}(w)) \,|\, w = w_f\} \tag{5.12}$$

then

$$D(w_f) = |\mathcal{P}_{w_f}|. \tag{5.13}$$

A Gaussian filter is applied to the density map to obtain blobs of high particle densities; the variance of the typical gaussian filter is 1 and the size is 11×11 pixels. The centroids of the blobs, which are the local peaks of the density map, are clustered using mean-shift algorithm [88]. (This

approach results in a significant increase in speed over using mean-shift directly on the density map.) and the number of particles in each cluster is a significance measure for that cluster. Then significant cluster centers are defined as accumulation points.

We point out that finding accumulation points and using them to locate behaviors, may be considered similar to a sink seeking process. For instance, [40] uses particle trajectories to find preferred directions of motion in crowd scenes by finding sinks, which are typically preferred exits or frequently visited regions of the scene. Our approach differs significantly, because we do not use information from neighboring particles to make conclusions about particle paths, meaning our approach is applicable to crowd flows of varying densities, provided there is a characteristic flow. In addition, sink seeking alone implies nothing about types of flow, which is our main concern.

Candidate Points, Paths, and Precincts. Particles at an accumulation point may arrive at that point in one of two ways.

- At the end of a lane, trajectories arrive at an accumulation point from one direction; in this case we label the region along trajectories a candidate path.

- At a bottleneck, trajectories necessarily arrive at an accumulation point from various directions; in this case we label the point a candidate point.

The following exposition delineates our method to distinguish these cases.

After particle advection, trajectories should satisfy two criteria. First, the final position of a particle must be close to an accumulation point. Second, the distance between the initial and the final positions of the particle should be long enough. These criteria are essential for selecting particles that describe the motion in the scene properly and discarding the others. For instance, a trajectory that does not reach the accumulation point does not satisfy the first criterion, and a non-motion particle does not satisfy the second criterion.

Trajectories of particles satisfying both criteria are analyzed, accounting for two possibilities as follows.

C1 Particles accumulate unidirectionally.

C2 Particles accumulate multidirectionally.

C1 implies the trajectory region is a candidate path; C2 implies accumulation points are candidate points.

Let d_0 and d_f be unit vectors for particle directions at times t_0 and t_f, respectively. For every trajectory, the angles between these vectors $\theta = \arccos(d_0 \cdot d_f)$, are clustered by Mean-shift algorithm [88] and the clusters are categorized into two groups; major and minor. Typically, a major (resp. minor) cluster contains at least one third (resp. one tenth) of the total number of trajectories at an accumulation point. C1 implies the trajectories correspond to at most two major clusters. C2 implies there are several minor clusters. Since one (possibly two) lane may end at an accumulation point, a candidate path is defined by particle trajectories for major clusters, but several different lanes ending at an accumulation point are better defined by a bottleneck. Thus, an ROI for a bottleneck or fountainhead is the area around a candidate point, which is an accumulation point with three or more minor clusters. Finally, high density regions are labeled candidate precincts, because further increases in density may lead to blocking.

It is certainly conceivable that problems may be encountered at this step as a result of perspective effects. To make this clear, notice that a scene with traffic flow on a long highway, extending into the distance, will appear to have particle trajectories converging at a point in the distance. According to our set-up, the accumulation point may be falsely labeled as a candidate point for a bottleneck, when the behavior is clearly a lane. However, we did not encounter such problems during implementation.

The ROI for a lane (a candidate path) is found by combining particle paths from major clusters. The ROI for the bottleneck is the region around a candidate point, as is the ROI for a fountainhead but the particles that accumulate at such a point do so in backward time. Furthermore, any high density regions with lanes are the potential areas for blocking and are labeled as candidate

precincts, since increases in traffic density can easily stop the regular flow of the lanes causing gridlocks.

5.3.2 Eigenvalue Map

To reduce noise and neglect regions without motion, we start by discarding small magnitude optical flow. Then the optical flow is averaged in time and we apply a median filter in space, giving a representation of optical flow for the entire sequence or clip, denoted $(\tilde{u}, \tilde{v})$. Considering only pixels in a given ROI, we analyze the eigenvalues of J_F at each pixel through Δ and τ. Using δ_x and δ_y to denote difference operators in each spatial direction, we compute Δ and τ using

$$\Delta = \delta_x \tilde{u} \cdot \delta_y \tilde{v} - \delta_y \tilde{u} \cdot \delta_x \tilde{v} \tag{5.14}$$

$$\tau = \delta_x \tilde{u} + \delta_y \tilde{v}. \tag{5.15}$$

Inside a ROI, pixels are colored according to the eigenvalues (up to a tolerance ϵ) as shown in Table 5.1, and we choose $\epsilon = 0.005$ in practice. The number of pixels satisfying each condition is counted, as shown in Table 5.1, and we call the total number of pixels in the ROI T. Thus it is possible to determine if a ROI is dominated by a behavior using the ratio conditions in Table 5.2.

Table 5.1: Eigenvalue responses and designated labels. *Count* is the number of pixels in a ROI satisfying a condition. $\Delta > \epsilon^2$ for each condition unless stated otherwise.

Eigenvalues	Condition	Label	Count
real, $\lambda_1 > 0$, $\lambda_2 < 0$	$\Delta < -\epsilon^2$	Green	G
real, both positive	$\tau < -2\epsilon$, $\tau^2 > 4\Delta$	Red	R
real, both negative	$\tau > 2\epsilon$, $\tau^2 > 4\Delta$	Yellow	Y
complex conjugate, positive real part	$\tau < -2\epsilon$, $\tau^2 < 4\Delta$	Magenta	A
complex conjugate, negative real part	$\tau > 2\epsilon$, $\tau^2 < 4\Delta$	Cyan	C
purely imaginary	$\lvert\tau\rvert < 2\epsilon$	White	W
at least one zero	$\lvert\Delta\rvert < \epsilon^2$	Blue	B
all zero (no motion)	$J_F = 0$	Black	K

Table 5.2: Ratio conditions determine dominance of a ROI by an eigenvalue response, corresponding to a behavior. (Tolerance L is chosen through experimentation)

Identified Behavior	Ratio Condition
Lane	$B/T > L$
Blocking	$G/T > L$
Bottleneck	$(R + A)/T > L$
Fountainhead	$(Y + C)/T > L$
Arch/Ring	$(W + A + C)/T > L$

5.4 Experimental Results

The method was tested on real video sequences downloaded from the web (Getty-Images, BBC Motion Gallery, Youtube, Thought Equity) and on sequences from the Performance Evaluation of Tracking and Surveillance (PETS) 2009 Dataset, representing crowd and traffic scenes. The number of overlaid particles is equal to the number of pixels. The videos have different fields of view, resolutions, frame rates, and duration, yet our method performed well in most cases. The performance was measured on more than 60 video sequences, which contain single or multiple behaviors, as shown in Table 5.3. Some results are illustrated in Figure 5.8. The positions of the identified behaviors in 20 different scenes are shown by the corresponding behavior symbols which are defined in the figure. Our framework enables the detection of multiple behaviors in one time execution.

Detailed steps for detecting five crowd behaviors are outlined and illustrated in Figures 5.3-5.5. In all three figures, we calculate the ROI and eigenvalue maps. In Figure 5.3, one bottleneck and one fountainhead are correctly detected, and one false candidate point is eliminated, since the eigen-value map does not imply the fountainhead behavior. In Figure 5.4, a lane and an arch in a traffic scene are correctly identified. In Figure 5.5, blocking behavior is correctly identified as two opposing lanes of pedestrians collide, impeding further advancement.

Table 5.3: Crowd Behavior Detection Results

Behavior	Total # of Behaviors	# of Detections	# of Missed	# of False
Lane	66	56	10	11
Blocking	3	3	0	0
Bottleneck	20	16	4	3
Fountainhead	29	23	7	5
Arch/Ring	28	23	5	6

To evaluate method performance, we compared detection against manually generated ground truth, consisting of locations for bottlenecks, fountainheads and blockings, and regions for lanes and arches on all videos; some examples are shown in Figure 5.6. The results are shown in Table 5.3. (The ground truth was manually generated for each video by an independent computer vision researcher, based on the behavior definitions that we have provided in Section 5.2.) Following the PASCAL VOC challenge [89], detection accuracy is based on overlap of the detected region and groundtruth. For lanes and arches we require an overlap of more than 40%, a relaxation of the Pascal measure appropriate for our problem. Similarly, the region around points that identify bottlenecks, fountainheads, or blocking is required to overlap with the analogous region from groundtruth; we require that the Euclidean distance between the detected point and groundtruth be sufficiently small, typically within 40 pixels. These conditions may seem relaxed, but given the diversity of scenes tested and the results obtained, they are reasonable. For example, if a bottleneck is actually 60 pixels across, a correctly detected bottleneck point may be 30 pixels away from groundtruth. Similarly, traffic in a lane can not be expected to fill the entire lane, and some leeway is necessary for accurate detection. Figure 5.7 shows ROC curves with True Positive Rate vs False Positive Per Video for four behaviors obtained by varying the tolerance L from Table 5.2.

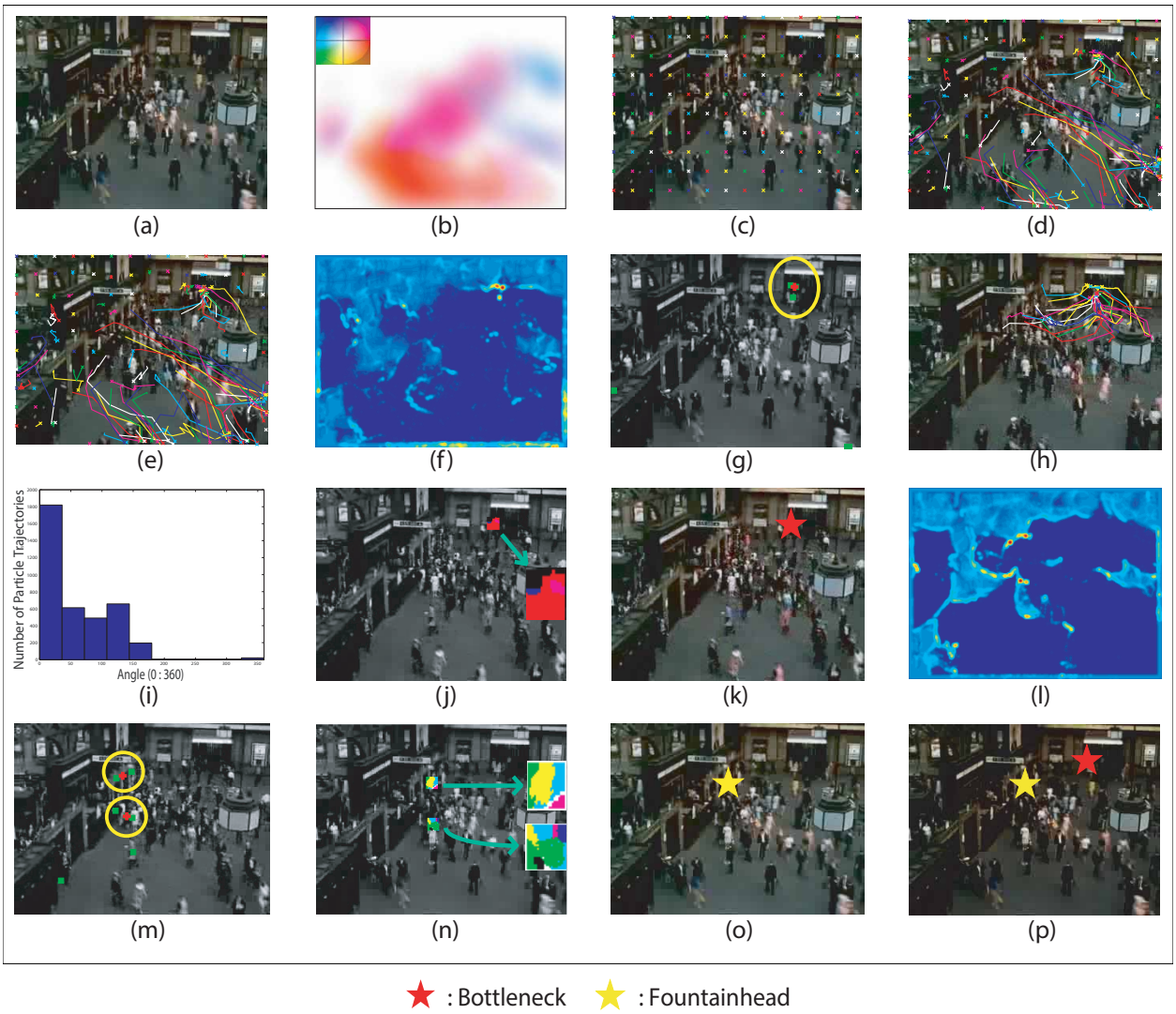

Figure 5.3: The process for detecting a bottleneck and a fountainhead: **(a)** Given a video scene, **(b)** compute optical flow, **(c)** overlay the scene with a grid of particles, **(d,e)** advect the particles according to the flow, **(f)** particles accumulate in some regions producing the density map, **(g)** local peaks ($green$) of the density map are clustered and centroids (red) of these clusters are found, **(h,i)** the particle trajectories around these accumulation points are clustered according to their angles, **(j)** candidate points are determined, these points are checked for bottlenecks using the eigenvalue map, **(k)** and a bottleneck is correctly detected at the red star, as the majority of entries in the eigenvalue map around the candidate points are red. **(l,m,n,o)** The same approach in backward time enables the correct identification of a fountainhead at the $yellow$ star. **(p)** shows the identified bottleneck and the fountainhead with a red and $yellow$ star, respectively.

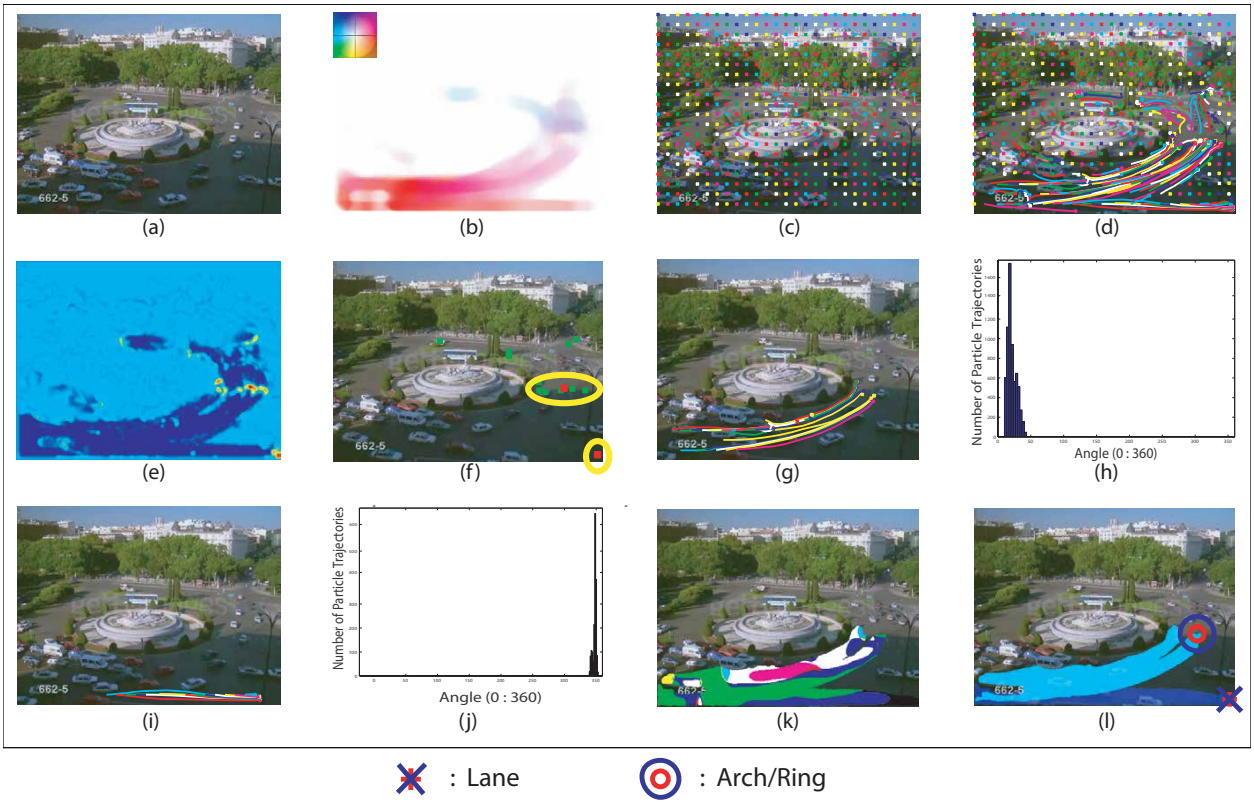

Figure 5.4: The process for detecting a lane and an arch: Steps **(a-f)** are the same as Figure 5.3, **(g-j)** particle trajectories around each accumulation point are clustered according to their angles, which reveal candidate paths and their directions **(k,l)** since the majority of the entries of the eigenvalue map is $blue$ along the straight path, that is correctly detected as a lane, whereas the majority of eigenvalue map entries are $white$, $magenta$ and $cyan$ along the path on the roundabout, where an arch is correctly detected.

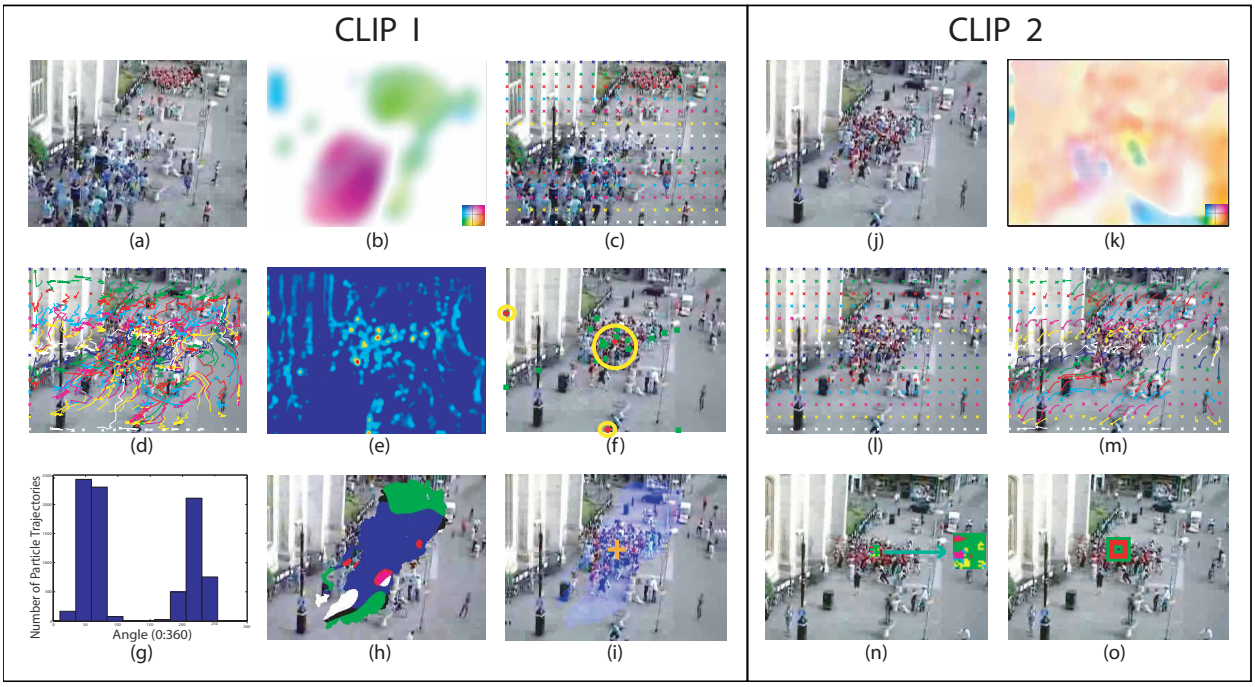

Figure 5.5: The process for detecting blocking: The sequence is divided into sequential clips and the process described in Figures 5.3 and 5.4 is applied to each clip. In Clip 1 **(a-i)** two lanes are detected, **(i)** they are opposing lanes, as the angle between the two lanes is near $180°$, so the center region is labeled a candidate precinct and saved for the next clip. In Clip 2 **(j-o)** the process is repeated, and the eigenvalue map around the saved candidate precinct **(n)** shows the majority of points have zero optical flow or $\Delta < 0$. So the region is correctly detected as blocking **(o)**, and this demonstrates a change in behavior due to bifurcation.

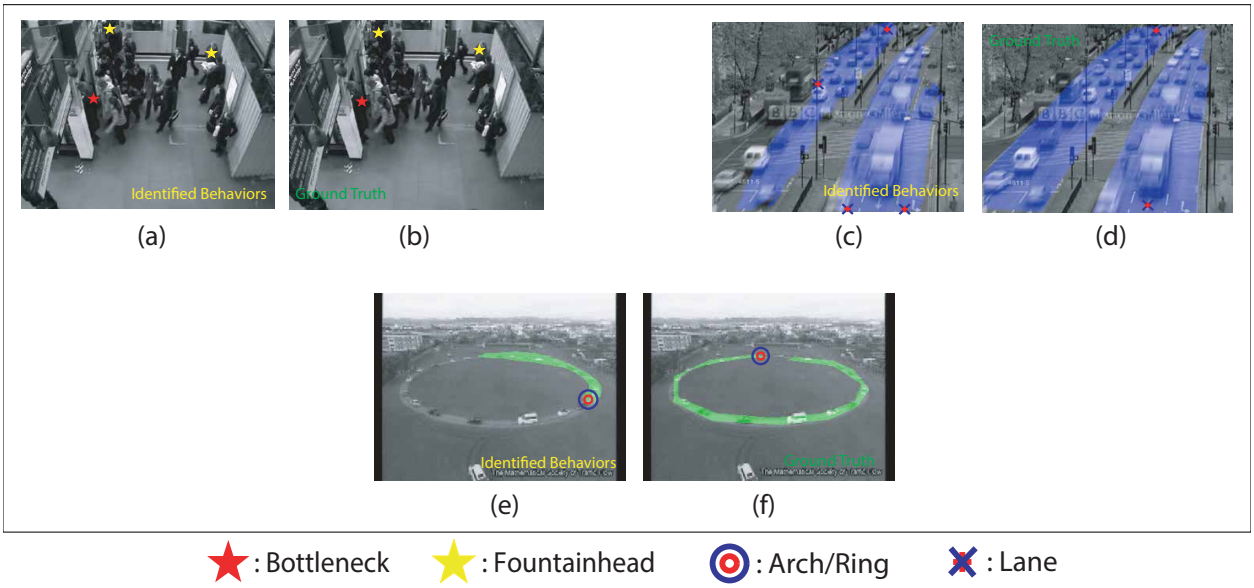

Figure 5.6: Comparisons of ground truth with algorithm results for three video sequences.

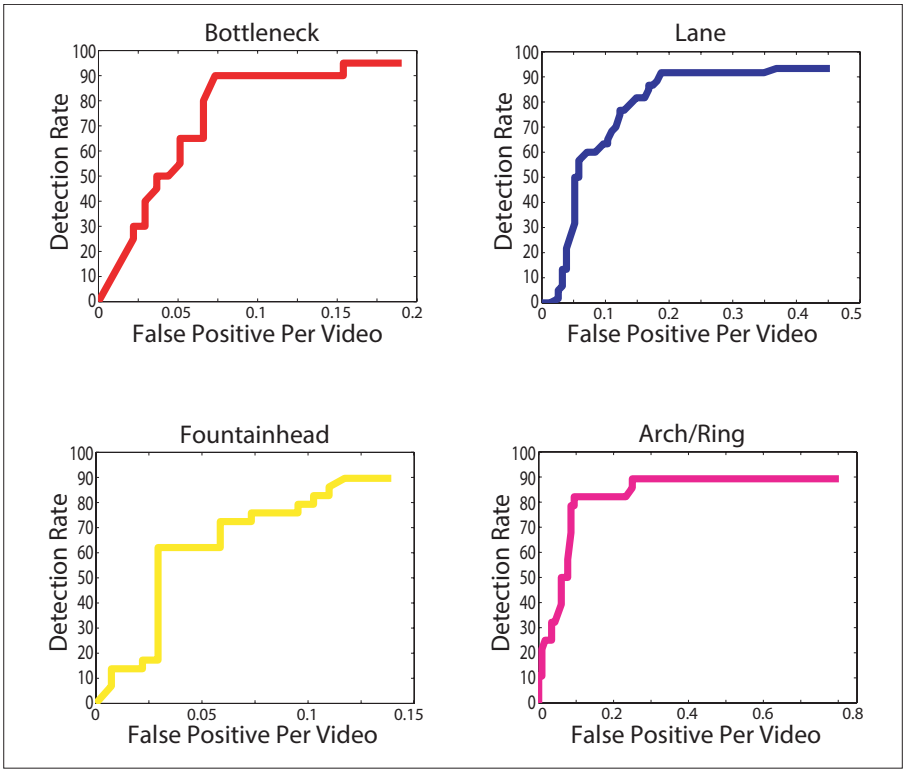

Figure 5.7: ROC curves for four behaviors: bottleneck, lane, fountainhead, arch/ring.

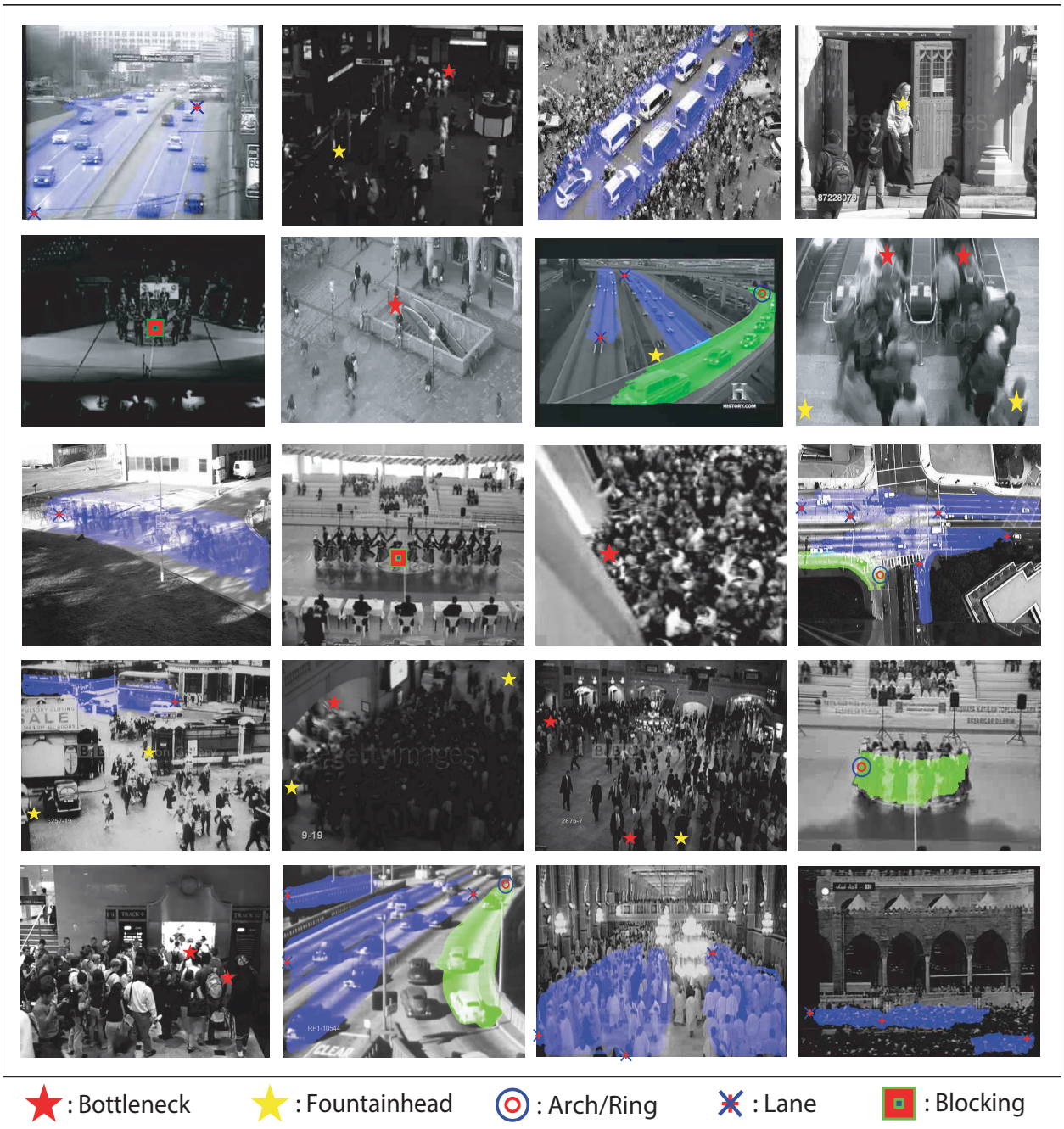

Figure 5.8: Scenes from 20 real video sequences, each showing the behaviors that are detected by the method

5.5 Summary

Video analysis of crowded scenes is a challenging problem in computer vision, since high densities of objects in real-world situations make individual object detection and tracking impractical. In this chapter, we presented a novel framework that can identify multiple crowd scene behaviors such as bottlenecks, fountainheads, lanes, arches/rings, and blocking without the need of object detection, tracking, or training. First, a dynamical system defined by the optical flow is initialized by overlaying a grid of particles on a scene. Numerical integration of this system provides particle trajectories. Next, these trajectories representing the motion in the scene are used to locate regions of interest. Finally, the linear approximation of the dynamical system provides behavior classification through the Jacobian matrix such that the type of dynamic stability of points in the regions of interest corresponds to one of the five crowd behaviors. The results illustrated in Figure 5.8 demonstrates the capability and flexibility of this method for a wide variety of scenes. In light of its strengths, the method does have shortcomings, which are listed here and open for future work. Our model is deterministic and can not capture the inherent randomness in the problem without a stochastic component. Our model can only identify five behaviors, which is an oversimplification of the complexities encountered in crowds. Our method is not useful when significant overlap of motion patterns is present in the scene, or when there is lack of consistent characteristic flow.

CHAPTER 6: CONCLUSION AND FUTURE WORK

In this dissertation we have addressed modeling and recognition of activities in videos. We have proposed a two-pronged approach that provides different models for activities at the fine and the coarse levels; the actions of individuals and the behaviors of crowds, respectively. We also presented a method for improving the classification performance of holistic descriptors by determining the subsets of discriminative descriptor blocks. We described our contributions and explained that the proposed work is aimed at filling a void in the literature.

Our approach for individual action recognition is based on the holistic descriptor, GIST3D, which captures the scene and motion components. This holistic descriptor is computed by applying a bank of 3-D filters on frequency spectrum of a video and each video is represented by a single feature vector. GIST3D is advantageous since it has a simple structure, which preserves the spatio-temporal relations of scene components and it avoids the detection of local interest points or quantization of local video descriptors into a codebook. For performance evaluation, we modeled and learned actions of individuals for categorizing complex user uploaded videos which are provided on multiple benchmark datasets and obtained good performance.

Despite their benefits, the holistic descriptors have drawbacks such as the long dimensionality of feature vectors. A holistic feature vector of a video clip is attained by concatenating the features of the individual spatio-temporal blocks together disregarding their contribution in classification. In this dissertation, we presented a method for specifying the optimal set of blocks by analyzing the discriminativity of individual descriptor blocks. Our method improved the performance of holistic descriptors on all benchmark datasets.

For behaviors in the crowds, we proposed a method which does not require tracking or training yet efficiently describes specific behaviors by viewing optical flow in a scene as a dynamical system. Numerically integrating the optical flow provides us the particle trajectories that represent the motion in the scene and the linearization of this dynamical system allows a practical

analysis of the behavior through the Jacobian matrix. To the best of our knowledge, we have used the relevant concepts from Lagrangian particle dynamics and stability analysis for dynamical systems for the first time to represent and label crowd behaviors in computer vision literature. Our method is able to identify five common and specific crowd behaviors; bottlenecks, fountainheads, lanes, arch/rings and blocking.

6.1 Future Directions

In this section we describe possible future directions to the research that was carried out in this dissertation. The approaches proposed in this dissertation can be improved in many ways.

In Chapter 4, a method was proposed for improving the performance of holistic descriptors as well as reducing their dimensionality. This method can further be improved by taking into account the scaling of blocks. For example, neighboring blocks which are not discriminative in one scale can be beneficial when they are combined in a higher scale. Or a block which is not discriminative can be divided into smaller blocks which contributive to classification. Hence, discriminativity of blocks can be analyzed by following a pyramidal approach.

As an extension of the work in Chapter 5, the temporal changes in crowd behaviors can be analyzed. There can be transitions between the crowd events in certain scenarios such as two lanes turning into a blocking, which was studied in this dissertation. Other types of transitions will exist in scenarios such as multiple lanes and arches start emerging from the same point where a fountainhead will occur or multiple lanes and arches merge to a path where a bottleneck will occur.

The presented holistic descriptor, GIST3D, can be tested for modeling crowd behaviors. Filters with different orientations and bandwidths can capture the scene and motion components which can distinguish the density of crowd and the type of the activity being performed.

This dissertation presented novel approaches for recognizing individual actions and crowd behaviors. The future research may focus on analyzing activities of groups of people. Such activi-

ties can be present in surveillance scenarios as well as in musical and sports events. For example, an orchestra is composed of a group of musicians who play together on various instruments. Each group such as the violinists, trombonists, basses, flutes has specific motion characteristics and coherency since the group members play the same partitures of the songs on same types of instruments. Likewise, in a sports event, the spectators, different team members and referees have motion and appearance similarities. These cues may help identifying groups and their activities.

LIST OF REFERENCES

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

[2] A. Klaser, M. Marszalek, and C. Schmid, "A spatio-temporal descriptor based on 3d-gradients," in *BMVC*, 2008.

[3] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. Comput. Vision*, vol. 42, pp. 145–175, May 2001.

[4] http://vision.eecs.ucf.edu/datasetsActions.html.

[5] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: a large video database for human motion recognition," in *ICCV*, 2011.

[6] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: a local svm approach," in *Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04)*, vol. 3, pp. 32–36, aug. 2004.

[7] M. S. Ryoo and J. K. Aggarwal, "UT-Interaction Dataset, ICPR contest on Semantic Description of Human Activities (SDHA)." http://cvrc.ece.utexas.edu/SDHA2010/Human_Interaction.html, 2010.

[8] M. Rodriguez, J. Ahmed, and M. Shah, "Action mach a spatio-temporal maximum average correlation height filter for action recognition," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pp. 1–8, 2008.

[9] K. Reddy and M. Shah, "Recognizing 50 human action categories of web videos," *Machine Vision and Applications*, pp. 1–11, 2012.

[10] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *CoRR*, vol. abs/1212.0402, 2012.

[11] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'08)*, 2008.

[12] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *VS-PETS*, pp. 65–72, 2005.

[13] P. Scovanner, S. Ali, and M. Shah, "A 3-dimensional sift descriptor and its application to action recognition," in *ACM Multimedia*, pp. 357–360, 2007.

[14] H. Wang, A. Kläser, C. Schmid, and C. Liu, "Action recognition by dense trajectories," in *CVPR*, 2011.

[15] S. Sadanand and J. J. Corso, "Action bank: A high-level representation of activity in video.," in *CVPR*, pp. 1234–1241, IEEE, 2012.

[16] I. Laptev, "On space-time interest points," *Int. J. Comput. Vision*, vol. 64, pp. 107–123, September 2005.

[17] J. Liu, J. Luo, and M. Shah, "Recognizing realistic actions from videos in the wild," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'09)*, pp. 1996 –2003, june 2009.

[18] N. Ikizler-Cinbis and S. Sclaroff, "Object, scene and actions: combining multiple features for human action recognition," in *Proceedings of the 11th European conference on Computer vision (ECCV'10)*, pp. 494–507, 2010.

[19] "Trecvid 2011," pp. http://www–nlpir.nist.gov/projects/tv2011/tv2011.html.

[20] S. Ali and M. Shah, "A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis,"

[21] G. Haller and G. Yuan, "Lagrangian coherent structures and mixing in two-dimensional turbulence," *Phys. D*, vol. 147, no. 3-4, pp. 352–370, 2000.

[22] R. L. Hughes, "A continuum theory for the flow of pedestrians," *Transportation Research Part B: Methodological*, vol. 36, no. 6, pp. 507–535, 2002.

[23] B. Solmaz, S. Assari, and M. Shah, "Classifying web videos using a global video descriptor," *Machine Vision and Applications*, pp. 1–13, 2012.

[24] R. Poppe, "A survey on vision-based human action recognition," *Image Vision Comput.*, vol. 28, pp. 976–990, June 2010.

[25] D. Weinland, R. Ronfard, and E. Boyer, "A survey of vision-based methods for action representation, segmentation and recognition," *Comput. Vis. Image Underst.*, vol. 115, pp. 224–241, February 2011.

[26] C. Harris and M. Stephens, "A combined corner and edge detector," in *In Proc. of Fourth Alvey Vision Conference*, pp. 147–151, 1988.

[27] A. Bobick and J. Davis, "The recognition of human movement using temporal templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, pp. 257–267, mar 2001.

[28] A. Yilmaz and M. Shah, "A differential geometric approach to representing the human actions," *Comput. Vis. Image Underst.*, vol. 109, pp. 335–351, March 2008.

[29] M. Black, "Explaining optical flow events with parameterized spatio-temporal models," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'99)*, vol. 1, pp. 326–332, 1999.

[30] R. Polana and R. C. Nelson, "Detection and recognition of periodic, non-rigid motion," *International Journal of Computer Vision*, vol. 23, pp. 261–282, June - July 1997.

[31] G. Johansson, "Visual perception of biological motion and a model for its analysis," *Perception & Psychophysics*, vol. 14, no. 2, pp. 201–211, 1973.

[32] L. Campbell and A. Bobick, "Recognition of human body motion using phase space constraints," in *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pp. 624–630, 1995.

[33] S. Wu, O. Oreifej, and M. Shah, "Action recognition in videos acquired by a moving camera using motion decomposition of lagrangian particle trajectories," in *IEEE International Conference on Computer Vision (ICCV'11)*, pp. 1419 –1426, nov. 2011.

[34] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'11)*, pp. 3169 –3176, june 2011.

[35] L. Kratz and K. Nishino, "Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009.*

[36] H. Zhong, J. Shi, and M. Visontai, "Detecting unusual activity in video," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004.*

[37] V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos, "Anomaly detection in crowded scenes," in *Computer Vision and Pattern Recognition, 2010. CVPR 2010.*

[38] G. Brostow and R. Cipolla, "Unsupervised bayesian detection of independent motion in crowds," in *Computer Vision and Pattern Recognition, 2006. CVPR 2006.*

[39] L. Kratz and K. Nishino, "Tracking with local spatio-temporal motion patterns in extremely crowded scenes," in *Computer Vision and Pattern Recognition, 2010. CVPR 2010.*

[40] S. Ali and M. Shah, "Floor fields for tracking in high density crowd scenes," in *ECCV '08: Proceedings of the 10th European Conference on Computer Vision*, pp. 1–14, 2008.

[41] S. Pellegrini, A. Ess, K. Schindler, and L. Van Gool, "You'll never walk alone: Modeling social behavior for multi-target tracking," *2009 IEEE 12th International Conference on Computer Vision*, 2009.

[42] T. Zhao and R. Nevatia, "Tracking multiple humans in crowded environment," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004.*

[43] V. Rabaud and S. Belongie, "Counting crowded moving objects," in *Computer Vision and Pattern Recognition, 2006. CVPR 2006.*

[44] D. Yang, H. Gonzalez-Banos, and L. Guibas, "Counting people in crowds with a real-time network of simple image sensors," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pp. 122 –129 vol.1, 13-16 2003.

[45] A. Chan and N. Vasconcelos, "Bayesian poisson regression for crowd counting," in *Computer Vision, 2009 IEEE 12th International Conference on*, pp. 545 –551, sept. 2009.

[46] A. B. Chan and N. Vasconcelos, "Mixtures of dynamic textures," in *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, pp. 641–647, 2005.

[47] P. Sand and S. Teller, "Particle video: Long-range motion estimation using point trajectories," in *Computer Vision and Pattern Recognition, 2006. CVPR 2006.*

[48] P. Tu, T. Sebastian, G. Doretto, N. Krahnstoever, J. Rittscher, and T. Yu, "Unified crowd segmentation," in *ECCV*, 2008.

[49] J. Wright and R. Pless, "Analysis of persistent motion patterns using the 3d structure tensor," in *Motion and Video Computing, 2005. WACV/MOTIONS '05. IEEE Workshop on*, vol. 2, pp. 14 –19, jan. 2005.

[50] R. Pini, M. Ofer, A. Shai, and S. Amnon, "Crowd detection in video sequences," *IEEE Intelligent Vehicles Symposium 2004*, pp. 66–71, 2004.

[51] B. Zhan, D. N. Monekosso, P. Remagnino, S. A. Velastin, and L.-Q. Xu, "Crowd analysis: a survey," *Mach. Vision Appl.*, vol. 19, no. 5-6, pp. 345–357, 2008.

[52] C. Stauffer and W. Grimson, "Learning patterns of activity using real-time tracking," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, pp. 747 –757, aug 2000.

[53] P. Viola, M. Jones, and D. Snow, "Detecting pedestrians using patterns of motion and appearance," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, vol. 2, pp. 734–741, 13-16 2003.

[54] J. S. Marques, P. M. Jorge, A. J. Abrantes, and J. M. Lemos, "Tracking groups of pedestrians in video sequences," *Computer Vision and Pattern Recognition Workshop*, vol. 9, p. 101, 2003.

[55] N. Johnson and D. Hogg, "Learning the distribution of object trajectories for event recognition," in *BMVC '95: Proceedings of the 6th British conference on Machine vision (Vol. 2)*, pp. 583–592, BMVA Press, 1995.

[56] R. Li and R. Chellappa, "Group motion segmentation using a spatio-temporal driving force model," in *Computer Vision and Pattern Recognition, 2010. CVPR 2010.*

[57] I. Laptev, "On space-time interest points," *Int. J. Comput. Vision*, vol. 64, no. 2-3, pp. 107–123, 2005.

[58] E. L. Andrade, S. Blunsden, and R. B. Fisher, "Modelling crowd scenes for event detection," in *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition*, pp. 175–178, IEEE Computer Society, 2006.

[59] M. Pittore, M. Campani, and A. Verri, "Learning to recognize visual dynamic events from examples," *Int. J. Comput. Vision*, vol. 38, no. 1, pp. 35–44, 2000.

[60] R. Mehran, A. Oyama, and M. Shah, "Abnormal crowd behavior detection using social force model," in *CVPR*, pp. 935–942, 2009.

[61] D. Helbing and P. Molnár, "Social force model for pedestrian dynamics," *Phys. Rev. E*, vol. 51, pp. 4282–4286, May 1995.

[62] M. Hu, S. Ali, and M. Shah, "Learning motion patterns in crowded scenes using motion flow field," in *International Conference on Pattern Recognition, 2008. ICPR 2008.*

[63] P. Widhalm and N. Brandle, "Learning major pedestrian flows in crowded scenes," *International Conference on Pattern Recognition, 2010. ICPR 2010.*

[64] A. R. Rao and R. C. Jain, "Computerized flow field analysis: Oriented texture fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, pp. 693–709, July 1992.

[65] R. Ford, "Critical point detection in fluid flow images using dynamical system properties," *Pattern Recognition*, vol. 30, no. 12, pp. 1991–2000, 1997.

[66] D. Helbing, "A Fluid Dynamic Model for the Movement of Pedestrians," *Complex Systems*, vol. 6, pp. 391–415, May 1992.

[67] A. Okubo, "Horizontal dispersion of floatable trajectories in the vicinity of velocity singularities such as convergencies," *Deep Sea Res*, vol. 17, pp. 445–454, 1970.

[68] A. Oliva, A. B. Torralba, A. Guerin-Dugue, and J. Herault, "Global semantic classification of scenes using power spectrum templates," *Challenge of Image Retrieval*, pp. 1–12, 1999.

[69] D. J. Heeger, "Notes on motion estimation, http://white.stanford.edu/∼heeger,," Nov. 1998.

[70] L. V. D. Maaten, E. O. Postma, and H. J. V. D. Herik, "Dimensionality reduction: A comparative review," 2008.

[71] H. Wang, M. M. Ullah, A. Klaser, I. Laptev, and C. Schmid, "Evaluation of local spatio-temporal features for action recognition," in *BMVC*, p. 127, sep 2009.

[72] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011.

[73] H. Jhuang, T. Serre, L. Wolf, and T. Poggio, "A biologically inspired system for action recognition," in *IEEE 11th International Conference on Computer Vision (ICCV'07)*, pp. 1–8, 2007.

[74] A. Gilbert, J. Illingworth, and R. Bowden, "Action recognition using mined hierarchical compound features.," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 883–897, 2011.

[75] X. Wang, T. Han, and S. Yan, "An hog-lbp human detector with partial occlusion handling," in *Computer Vision, 2009 IEEE 12th International Conference on*, pp. 32–39, 2009.

[76] M. S. Asif and J. K. Romberg, "Dynamic updating for l1 minimization," *CoRR*, vol. abs/0903.1443, 2009.

[77] S. Wright, R. Nowak, and M. A. T. Figueiredo, "Sparse reconstruction by separable approximation," *Signal Processing, IEEE Transactions on*, vol. 57, no. 7, pp. 2479–2493, 2009.

[78] A. Patron-Perez, M. Marszalek, I. Reid, and A. Zisserman, "Structured learning of human interactions in tv shows," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 12, pp. 2441–2453, 2012.

[79] F. Yuan, V. Prinet, and J. Yuan, "Middle-level representation for human activities recognition: The role of spatio-temporal relationships," in *Trends and Topics in Computer Vision* (K. Kutulakos, ed.), vol. 6553 of *Lecture Notes in Computer Science*, pp. 168–180, Springer Berlin Heidelberg, 2012.

[80] D. Waltisberg, A. Yao, J. Gall, and L. Van Gool, "Variations of a hough-voting action recognition system," in *Proceedings of the 20th International conference on Recognizing patterns in signals, speech, images, and videos*, ICPR'10, (Berlin, Heidelberg), pp. 306–312, Springer-Verlag, 2010.

[81] M. S. Ryoo and J. Aggarwal, "Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities," in *Computer Vision, 2009 IEEE 12th International Conference on*, pp. 1593–1600, 2009.

[82] Q. Le, W. Zou, S. Yeung, and A. Ng, "Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 3361–3368, 2011.

[83] A. Kovashka and K. Grauman, "Learning a hierarchy of discriminative space-time neighborhood features for human action recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'10)*, pp. 2046 –2053, june 2010.

[84] A. Kovashka and K. Grauman, "Learning a hierarchy of discriminative space-time neighborhood features for human action recognition," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 2046–2053, 2010.

[85] X. Wu, D. Xu, L. Duan, and J. Luo, "Action recognition using context and appearance distribution features," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 489–496, 2011.

[86] S. H. Strogatz, *Nonlinear dynamics and chaos : with applications to physics, biology, chemistry, and engineering*. Addison-Wesley, 1994.

[87] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *IJCAI'81: Proceedings of the 7th international joint conference on Artificial intelligence*, pp. 674–679, 1981.

[88] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *Information Theory, IEEE Transactions on*, vol. 21, no. 1, pp. 32–40, 1975.

[89] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes (VOC) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.