ABSTRACT

Title of dissertation: SEGMENTATION, RECOGNITION, AND ALIGNMENT OF COLLABORATIVE GROUP MOTION

Ruonan Li

Dissertation directed by: Professor Rama Chellappa

Modeling and recognition of human motion in videos has broad applications in behavioral biometrics, content-based visual data analysis, security and surveillance, as well as designing interactive environments. Significant progress has been made in the past two decades by way of new models, methods, and implementations. In this dissertation, we focus our attention on a relatively less investigated sub-area called *collaborative group motion* analysis. Collaborative group motions are those that typically involve multiple objects, wherein the motion patterns of individual objects may vary significantly in both space and time, but the collective motion pattern of the ensemble allows characterization in terms of geometry and statistics. Therefore, the motions or activities of an individual object constitute local information. A framework to synthesize all local information into a holistic view, and to explicitly characterize interactions among objects, involves large scale global reasoning, and is of significant complexity. In this dissertation, we first review relevant previous contributions on human motion/activity modeling and recognition, and then propose several approaches to answer a sequence of traditional vision questions including 1) which of the motion elements are those relevant to a group motion pattern of interest (Segmentation); 2) what is the underlying motion pattern (Recognition); and 3) how two motion ensembles are similar and how we can 'optimally' transform one to match the other (Alignment). Our primary practical scenario is American football play, where the corresponding problems are 1) who are offensive players; 2) what are the offensive strategy they are using; and 3) whether two plays are using the same strategy and how we can remove the spatio-temporal misalignment between them due to internal or external factors. The proposed approaches discard traditional modeling paradigm but explore either concise descriptors, hierarchies, stochastic

mechanism, or compact generative models to achieve both effectiveness and efficiency. In particular, the intrinsic geometry of the spaces of the involved quantities is exploited and statistical models are established on these nonlinear manifolds. These initial attempts have identified new challenging problems in complex motion analysis, as well as in more general tasks in video dynamics. The insights gained from nonlinear geometric modeling and analysis in this dissertation may hopefully be useful toward a broader class of computer vision applications.

SEGMENTATION, RECOGNITION, AND ALIGNMENT OF COLLABORATIVE GROUP MOTION

by

Ruonan Li

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2011

Advisory Committee:
Professor Rama Chellappa, Chair/Advisor
Professor Larry Davis
Professor David Jacobs, Dean's Representative
Professor Min Wu
Doctor Shaohua Zhou
Professor Todd Zickler

Dedication

To all individuals and organizations who made this dissertation possible.

Acknowledgments

First and foremost I owe my gratitude to my advisor, Professor Rama Chellappa for supporting me to work on this challenging and interesting topic over the past several years. Professor Chellappa was always available whenever I sought his help and advices. My discussions with him were always encouraging and inspiring. More importantly, he gave me the maximum freedom to explore the exciting unknown. It has been a precious experience to work with and learn from him, which I will continue to benefit from and cherish in my future career.

I would like to thank Professor Larry Davis for valuable guidance and support on research projects and other affairs, Professor David Jacobs for serving on the committee, Professor Min Wu for instructive interactions with me on many aspects, Dr. Shaohua Zhou for following up my progress and significant help with my paper submissions, and Professor Todd Zickler for kindly hosting my stay at Harvard and the pleasant cooperation during my stay.

Other unforgettable individuals include Professor Nuno Martins and Professor Adrian Papamarcou who helped me in my early stage of PhD study, Professor Prakash Narayan and Professor P. S. Krishnaprasad who offered great courses from which I learned a lot, and Professor Steven Marcus from whom I learned about how to be a good teacher.

It was a pleasure to work with every fellow member of Professor Chellappa's group, among whom I should particularly mention Raghuraman Gopalan and Sima Taheri for fruitful collaborations on research projects, Dikpal Reddy for assistances and 'synchronization' with me in a series of academic activities. Within the Center for Automation Research, I was happy as well to work or share a great time with Behjat Siddiquie and Arpit Jain. I am so grateful to Janice Perrone who took care of administrative issues and made my stay

with CfAR smooth.

I express my gratitude to Dr. Xu Liu, Dr. Yongle Wu, and Minhua Chen for their substantial help and making my life enjoyable and thoughtful.

I apologize for missing anyone out. This dissertation will be impossible without you.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Modeling and recognition of human motion in video data has broad applications in vision such as behavioral biometrics, content-based visual data analysis, security and surveillance, designing interactive environments, as well as animation and synthesis. Research in this area has seen promising contributions including new models, methods, and implementations. In this dissertation, we focus our attention on a relatively less investigated sub-area called *collaborative group motion* analysis. Practically, collaborative group motion will involve multiple participating objects rather than a single one. The motion patterns among individual objects may vary significantly in both space and time, and we want to identify models and/or design discriminative tools for the collective motion pattern of the object ensemble. Note that in this setting the motions or activities of an individual object constitute local information, and the framework to fuse all local information into a holistic view and to explicitly characterize interactions and cooperations among objects necessities large scale inference and is of significant complexity. In this dissertation, we first review related prior work on human motion/activity related vision problems, and more importantly, propose solutions to several novel problems including 1) which of the motion elements among all are the ones relevant to a group motion pattern of interest (Segmentation); 2) what is the underlying motion pattern (Recognition); and 3) how two motion ensembles are similar and how we can 'optimally' transform one to match the other (Alignment). Our primary practical scenario is American football plays, in which we want to locate, recognize, and match the offensive strategies using the motion trajectories extracted from football play videos. The proposed models and algorithms, at the same time, can also be used toward various dynamic visual data as well. Overviews about these contributions are given in the following four sections and are further elaborated in later chapters. These initial attempts have identified many challenging new problems in complex motion analysis, as well as in more general tasks in video dynamics. These potential research topics will be discussed in the end as potential future work.

1.1 Group Motion Recognition on Discriminative Temporal Interaction Manifold

In the first contribution we explore a 'data-driven' scheme to recognize the offensive play strategies in a football play. For this purpose, we assume that the players' roles and their motion trajectories are already available. For the former, we may recognize the roles from the initial play formation with the help of landmark shape theory [14], and for the latter we may employ a multi-object tracker [15]. These two problems are still being researched and are not considered in this dissertation. Specifically, we describe a group motion pattern with a full four-dimensional object-time interaction array, and learn an optimized array reduction kernel to condense it to a discriminative temporal interaction matrix. The temporal interaction matrix serves as a compact 'descriptor' for the group activity pattern, and has been empirically observed to be stable under view changes. More importantly, given a Riemannian metric the set of all temporal interaction matrices forms a Riemannian manifold, on which we are able to establish a probabilistic framework to characterize a given group motion pattern. We call this manifold Discriminative Temporal Interaction Manifold (DTIM). To learn a multi-modal 'likelihood' density for each class, we create a basic exponential density component on the manifold, and incrementally build up the complete manifold-resided densities with the basic components. Within this framework, a MAP classifier is used to recognize a new group motion observation. In addition we show that many other spatio-temporal features also fall into this framework and the proposed method is therefore applicable to other scenarios as well.

1.2 Group Motion Segmentation using a Hierarchical Driving-Force Model

In the second contribution we consider the 'group motion segmentation' problem and provide a solution for it. The group motion segmentation problem aims at analyzing motion trajectories of multiple objects in videos and finding among them the ones involved in a 'group motion pattern'. This problem is motivated by and serves as the basis for the recognition problem. By 'segmenting' a group motion we mean dividing all objects in motion into two sets, one of which behaves according to a group-wise motion pattern. In a football play only the offensive players follow the offensive strategy of interest, while the whole trajectory set also includes those of defensive players which should be 'filtered

out'. Specifically, we learn a Spatio-Temporal Driving Force Model to characterize a group motion pattern and design an approach for segmenting the group motion. Note that our segmentation is based on motion only, *i.e.*, we do not use other discriminative features. We illustrate the approach using videos of American football plays, where we identify the offensive players, who follow an offensive motion pattern, from motions of all players in the field. Since motion trajectories arise in other scenarios, such as tracks of local spatio-temporal interest points, we will also demonstrate the applicability of the model to generic trajectory ensembles, for the purpose of discovering a relevant group of motion of interest.

1.3 Matching Two Motion Ensemble: A Stochastic Optimization on the Alignment Manifold

In the third contribution we investigate the spatio-temporal alignment of two group motion patterns, and more generally, videos or features/signals extracted from them. Specifically, we formally define an *alignment manifold* and formulate the alignment problem as an optimization procedure on this non-linear space by exploiting its intrinsic geometry. We focus our attention on semantically meaningful videos or signals, *e.g.*, those describing or capturing human motion or activities, and propose a new formalism for temporal alignment accounting for executing rate variations among realizations of the same video event. By construction, we address this static and deterministic alignment task in a dynamic and stochastic manner: we regard the search for optimal alignment parameters as a recursive state estimation problem for a particular dynamic system evolving on the alignment manifold. Subsequently, a Sequential Importance Sampling iteration on the alignment manifold, as well as an extended version incorporating stochastic gradient information, are designed for effective and efficient alignment. We demonstrate the performance on not only motion ensembles but also several other types of input data that arise in vision applications.

1.4 A Generative Model for Joint Segmentation and Recognition

In the final contribution we jointly recognize and segment coordinated group motions by proposing a generative model which describes the spatial coordination among objects. Under this compact but comprehensive graphical model, we start from an offensive play strategy, take all the factors contributing to an football play into consideration, and end

up with the observed tracks for the players. With the Bayesian framework we are able to jointly perform segmentation and recognition. This work can be seen as extending the 'static' problem of shape segmentation [16] to a 'dynamic' case. In particular, in this framework we also statistically characterize the variations due to view changes for group activities, and achieve view-invariant recognition and segmentation. As a by-product, we also show that the co-occurrence function occurring as a component of the framework, is a discriminative feature itself, and thus can be used for group motion recognition as well.

1.5 Organization of the Dissertation

The rest of the dissertation is organized as follows. In Chapter 2, we present a brief literature survey on human motion and activity recognition from videos. Then, we discuss the recognition problem using a statistical formulation on DTIM in Chapter 3. We elaborate on the group motion segmentation problem in Chapter 4, followed by a detailed discussion about the alignment manifold and two particle filters implemented on it in Chapter 5. Then, we present the probabilistic generative model of group motions together with a joint-recognition-and-segmentation method in Chapter 6. Finally, in Chapter 7 we discuss future research directions.

Chapter 2

Related Work on Human Motion Recognition

In this chapter we give a brief literature review regarding human motion modeling and recognition. After an overview section we discuss simple low-level features and non-parametric methods for human motion and activities. Then we turn to more complex models as well as invariance issues in recognition problems. In particular, we describe recent works on multi-object group motion recognition, which is closely related to our work. This chapter is largely based on a recent survey by Turaga et.al. [17], which is one of the most comprehensive surveys on past and current research realted to this dissertation. In addition, we briefly review differential geometric approaches in computer vision as the background of using analytical manifolds for group motion analysis.

Human motion is closely related to human actions and activities. We repeatedly see the terms 'action' and 'activity' in the vision literature at times in a non-distinguishable way. In [17], the authors suggest the following definition. By 'action' they refer to a simple motion pattern usually executed by a single person, and mostly lasting for short durations of time typically on the order of a few seconds. Examples of this type include bending, walking etc., captured in standard datasets like KTH dataset. On the other hand, by 'activity' they refer to the complex sequence of actions performed by several people who could be interacting with each other in a constrained manner. The activities are typically observed over much longer durations, for example, two persons shaking hands, a football team scoring a goal or a coordinated bank attack by multiple robbers. However, in fact it is not trivial to find a well defined boundary between the two categories and there is a 'transition zone' between the two extremes. For example, the gestures of a music conductor conducting an orchestra, or the constrained dynamics of a group of humans, can neither be classified as a simple 'action' nor as a complex 'activity' following the above interpretation. In this dissertation we do not pursue a precise distinction among 'human motion', 'action', or 'activity', but use the term 'human motion' in a loose sense. In the context of a specific problem, an explicit interpretation will not be difficult.

Other survey papers on topics related to human motion modeling and recognition from videos have appeared over the past two decades. The following ones have been widely

cited. Aggarwal and Cai [18] divided a complete action recognition system into three components – extraction of human body structure from images, tracking across frames, and finally, recognizing the action, and then discussed each of them. Cedras and Shah [19] presented a survey on motion-based approaches to recognition as opposed to structure-based approaches. They argue that motion is a more important cue for action recognition than the structure of the human body. Gavrila's survey [20] mainly focused on tracking of hands and humans via 2D or 3D models and presented a discussion of action recognition techniques. Moeslund et al [21] recently summarized problems and approaches in human motion capture including human model initialization, tracking, pose estimation and activity recognition. For a survey about the new progress made in the recent past, one may refer to [22].

As pointed out in [17], a generic human motion recognition system can be viewed as proceeding in a series of steps, from a sequence of images to a higher level semantic characterization. The major steps involved are sequentially 1)Extraction of descriptive/discriminative low-level features; 2)Mid-level representations from low-level features, if necessary; and 3)High-level semantic interpretations from primitive motions. Lower-level modules of detection and tracking as those discussed in length in the above surveys as well as in [15] are beyond the scope of this dissertation, and in this chapter we will mostly focus on the mid to high level vision tasks – modeling and recognition, assuming that pre-processing has been done to a reasonable extent.

2.1 Features and Simple Non-Parametric Methods

In this section, we discuss relevant aspects on low-level features extracted from raw videos containing the human motion of interest. We also present a list of basic tools that make use of these features for human motion recognition.

The reason for feature extraction is due to the fact that video consist of massive amounts of raw information, and not all content in videos is directly relevant to the task of understanding and identifying the activity captured in the video. Johansson [23] demonstrated that humans can perceive gait patterns from point light sources placed at a few limb joints without additional information. Additional visual information in terms of color of the clothing, illumination conditions, background clutter do not contribute more to the recognition task. The following popular low-level features have been widely adopted in research and practical applications.

Optical flow, defined as the apparent motion of individual pixels on the image plane, frequently serves as a good approximation of the true physical motion projected onto the image plane. Brightness constancy assumption is the basis of most methods for computing optical flow. Optical flow provides a local description of both the regions of the image undergoing motion and the velocity of motion. In practice, computation of optical flow significantly suffers from noise and illumination changes. Applications as early as [24] used optical flow to detect and track vehicles in an automated traffic surveillance scenario.

Point trajectories of moving objects are also among popular features for characterizing the motion of the object. The image-plane coordinates of trajectories may be useful only in limited cases, as the absolute locations will change with typical transformations like translations, rotations and scaling. Augmented features like trajectory velocities, trajectory speeds, spatio-temporal curvature, relative-motion etc have been shown to be invariant to some of these variations. See [19] for relevant approaches. Extracting unambiguous point trajectories from video is sensitive to nuisance factors such as occlusions, noise, background clutter etc., which highlights the necessity of accurate tracking, another very challenging research topic [15].

Background subtracted blobs come from background subtraction, a popular method for identifying moving regions of a scene by segmenting it into background and foreground. Several approaches to background modeling exist, among which the most popular one may be the one that learns a statistical distribution of pixel intensities from the background as in [25]. By adaptively updating the background model according to incoming data, the method can also be applied to scenarios with changing background.

Shape of the human silhouette plays a very important role in perceiving human motions. Approaches to quantify human shape include global, boundary and skeletal based ones. Global methods compute the numerical shape descriptors from the whole region of the moving object, e.g., shape moments [26]. Boundary methods, on the other hand, consider only the shape contour as the characterization of the shape. Chain codes [27] and landmark-based statistical shape descriptors [28] belong to this category. Skeletal methods represents a complex shape as a set of 1D skeletal curves, for example, the medial axis transform [29]. Applications include shape-based dynamic modeling of the human silhouette [30] for gait recognition.

With low-level features in hand we may turn our attention to the design of classifiers.

Classifiers for recognizing human motion vary from simple non-parametric ones to complex parametric ones. Parametric approaches typically impose a model on the dynamics of the motion. Non-parametric approaches rely on coarse representations drawn from data such as motion-templates. We will first discuss the non-parametric methods in this section. Parametric approaches, e.g., Hidden Markov Models (HMMs), Linear Dynamical Systems (LDSs) etc., identify a model involving mathematical constraint and controlling parameters, which for a class of human motions is normally estimated from training data. We will discuss them in the next section.

One of the earliest attempts at motion/action recognition by Polana and Nelson [31] resembles *2D-templates* matching. Motion detection and tracking of humans in the scene was first performed in their approach. Then a cropped sequence containing the human is generated with the scaling effect compensated for. After that, 'periodicity index' is computed for the given human motion to describe quantitatively how 'periodic' the underlying human motion is. To perform recognition, the nearly periodic sequence is segmented into individual cycles using the periodicity estimate and combined to get an average-cycle. The average-cycle is divided into a few temporal segments and flow-based features are computed for each spatial location in each segment. The flow-features in each segment are averaged into a single representation. The average-flow frames within an activity-cycle form the templates for each action class. Another early attempt for representation and recognition of quasi-cyclic actions appeared in [32], which is also based on periodicity analysis and successfully applied to quasi-periodic actions such as walking, running, swimming etc.

Alternatively, 'temporal templates' were proposed by Bobick and Davis [33] as models for human motions. In their implementation, background subtraction comes first, followed by an aggregation of a sequence of background subtracted blobs into a single static image. They propose two methods of aggregation – the first method gives equal weight to all images in the sequence, which gives rise to a representation called the 'Motion Energy Image' (MEI). The second method gives decaying weights to the images in the sequence with higher weight given to new frames and low weight to older frames. This leads to a representation called the 'Motion History Image' (MHI). The MEI and MHI together comprise a 'template' for a given action. From the templates, translation, rotation and scale invariant moments are extracted which are then used as features for recognition. It turned out that MEI and MHI have sufficient discriminating capability for several simple human motion classes such

as 'sitting', 'bending', 'crouching' and other aerobic postures. However, it was also noted that MEI and MHI do not perform well for long-term complex human motions due to over aggregation.

In contrast to extracting features from individual frames of the video sequence, another line of study directly regards human motions as *3-D spatio-temporal volumes*. Chomat et al. [34] model a segment of video as a spatio-temporal volume indexed by (x, y, t) and compute local appearance models at each pixel using a Gabor filter bank at various orientation and spatial scales and a single temporal scale. A given activity is recognized using a spatial average of the probabilities of individual pixels in a frame. As activities are analyzed at a single temporal scale, this method can only be employed without variations in execution rate among activities. Extending this approach, local histograms of normalized space-time gradients at several temporal scales are extracted by Zelnik-Manor and Irani [35]. The sum of the chi-square statistic between histograms is used to match an input video with pre-stored training samples.

The *Bag-of-Features* approaches have drawn sufficient attention from motion analysis community in recent years, due to its promising performance in still object recognition. Laptev and Lindeberg [36] initially proposed a spatio-temporal generalization of the Harris interest point detector and applied it to modeling and recognition of actions. Subsequently, Dollar et al. [37] modeled a video sequence using distribution of space-time (ST) feature prototypes. The feature prototypes are obtained by k-means clustering of a large set of space-time gradients extracted at extracted ST interest points. Neibles et al. [38] used a similar approach where they use a bag-of-words model to represent actions. The bag-of-words model is learnt by extracting spatio-temporal interest points and clustering of the features. Note that since most interest points are obtained by linear operations such as filtering and spatio-temporal gradients, the descriptors are sensitive to changes in appearance, noise, occlusions etc.. Moreover, interest points are often sparse in space and time and certain types of actions do not give rise to distinctive features [38, 37].

Also inspired by the success in still object recognition, researchers working on videos tried to seek alternate representations of motion patterns as spatio-temporal objects. Syeda-Mahmood et al. [39] proposed a representation of actions as generalized cylinders in the joint (x, y, t) space. Yilmaz and Shah [40] represent actions as 3-D objects induced by stacking together tracked 2-D object contours. A sequence of 2-D contours in (x, y) space

can be treated as an object in the joint (x, y, t) space. This representation encodes both the shape and motion characteristics of the human. From the (x, y, t) representation, concise descriptors of the object's surface are extracted corresponding to geometric features such as peaks, pits, valleys and ridges. Since this approach is based on stacking together a sequence of silhouettes, accurate correspondence between points of successive silhouettes in the sequences needs to be established. Parallel to this approach, [41] proposed using background subtracted blobs, instead of contours, which are stacked together to create an (x, y, t) binary space-time volume. Since this approach uses background subtracted blobs, the problem of establishing correspondence between points on contours in the sequence does not exist. From this space-time volume, 3-D shape descriptors are extracted and used as features for classifying different human motions.

2.2 Parametric Methods

In the previous section we focused on simple non-parametric representations and models for human motion classes. The parametric approaches that we describe in this section are much more powerful for modeling spatially and/or temporally complex activities. Methods such as HMMs, LDSs are well suited for activities governed by complex temporal dynamics. By comparison, simple non-parametric matching methods often require too many templates or are too weak to capture the dynamics. Examples of such complex actions include the steps in a ballet dancing video, a juggler juggling a ball and a music conductor conducting an orchestra using complex hand gestures. Accurate modeling and recognition of these complex motions requires more sophisticated methods that explicitly characterize the temporal dynamics.

One of the most popular methods used for modeling complex temporal dynamics is the so called *state-space* approaches. State-space approaches model the temporal evolution of features as a trajectory in some configuration space, where each point on the trajectory corresponds to a particular 'configuration' or 'state'. Among different state-space models, *Hidden Markov Models* or HMMs has been prevalent. In the discrete HMM formalism, the state space is considered to be a finite set of discrete states. The temporal evolution is modeled as a sequence of probabilistic jumps from one state to another. A useful source for a detailed technical introduction to HMMs and its three core modules – Inference, Decoding and Learning – is detailed in [42], targeted for speech precognition applications. Since the

90's, HMMs have been finding wide applicability in computer vision systems. One of the earliest approaches to recognize human actions with HMMs was proposed by Yamato et al. [43] where they recognized tennis shots such as backhand stroke, backhand volley, forehand stroke, forehand volley, smash etc by modeling a sequence of background subtracted images as outputs of class-specific HMMs. Several successful gesture recognition systems reported in [44, 45], make extensive use of HMMs by modeling a sequence of tracked features such as hand blobs as HMM outputs. Apart from gesture recognition, HMMs have also found application in modeling the temporal evolution of human gait patterns both for human motion recognition and biometrics (cf. Kale et al. [46], Liu and Sarkar [47]).

The HMM-based approaches assume that the feature sequence being modeled is the result of a single person performing an action. Hence, they are not directly applicable to applications where there are multiple agents performing an action or interacting with each other. To extend its application beyond single-object activities, Brand et al [48] proposed a coupled HMM to represent the dynamics of interacting targets. The superiority of their approach is demonstrated over conventional HMMs in recognizing two-handed gestures. Incorporating domain knowledge into the HMM formalism has been investigated by researchers like Moore, Essa and Hayes [49]. They use HMMs in conjunction with object detection modules to exploit the relationship between actions and objects. Hongeng and Nevatia [50] incorporate *a priori* beliefs of state-duration into the HMM framework to get a 'Hidden semi-Markov Model' (semi-HMMs). Recently, Cuntoor and Chellappa [51] have proposed a mixed-state HMM formalism to model non-stationary activities, where the state-space is augmented with a discrete label for higher-level behavior modeling. With both generative and discriminative capabilities, HMMs have proved to be efficient for modeling temporal data. HMMs are well-suited for tasks that require recursive probabilistic estimates or when explicit segmentation into atomic action units is difficult. However, their utility is also restricted due to the simplifying assumptions that the model is based on. Specifically, the assumption of Markovian property and the time-invariant nature of the model restricts the applicability of HMMs to relatively short-duration activities and in particular to stationary temporal patterns.

Linear Dynamical Systems are special cases of state-space models, where the probabilistic dynamics are usually constrained to be linear Gaussian ones, but also generalized cases of HMMs in the sense that the state-space is not necessarily a finite set of symbols but

can take on continuous values in $\mathbb{R}^k$ where k is the dimensionality of the state-space. One of the simplest forms of LDS is the first order time-invariant Gauss-Markov process which can be interpreted as a continuous state-space generalization of HMMs with a Gaussian observation model. Quite a few of applications such as recognition of humans and actions based on gait (Bissacco et al [52], Veeraraghavan et al [53], Mazzaro et al. [54]), as well as dynamic texture modeling and recognition [55, 56] have been proposed using LDSs. First order LDSs were used by Vaswani et al [2] to model the configuration of groups of people in an airport setting by considering a collection of moving points (humans) as a deforming shape. Advances in system identification theory for learning LDS model parameters (c.f. from data [57]) and distance metrics on the LDS space [58] have given rise to the popularity of LDSs for learning and recognition of high-dimensional time-series data. Studies on extended LDS models enable novel application such as dynamic boosting [59], kernel density estimation [60], and kernel distances [61].

To further extend LDS in order to cope with more involved dynamics, *Switching Linear Dynamical Systems* (SLDS) have been proposed. To motivate SLDS, consider the following long-term motion – a person bends down to pick up an object, then walks to a nearby table and places the object on the table and finally rests on a chair. This activity is composed of a sequence of short segments each of which can be modeled as a LDS. The entire process, therefore, switches among different LDSs. SLDS is also called Jump Linear Systems (JLS), consisting of a set of LDSs with a switching function that causes model parameters to change by switching between models. As a prototype of SLDS, a multi-layered approach is presented by Bregler [62] to recognize complex movements consisting of several levels of abstraction. The lowest level is a sequence of input images. The next level consists of 'blob' hypotheses where each blob is a region of coherent motion. At the third level, blob tracks are grouped temporally. The final level, consists of a HMM which represents the complex behavior. North et al [63] augment the continuous state vector with a discrete state component to make a 'mixed' state. The discrete component represents a mode of motion or more generally a 'switch' state. Corresponding to each state, a Gaussian Auto-Regressive Process (ARP) model is used to represent the dynamics. A maximum likelihood approach is used to learn the model parameters for each motion class. Pavlovic and Rehg [64] model the non-linearity in human motion in a similar framework, where the dynamics are modeled using LDS and the switching process is modeled using a

probabilistic finite state-machine. Though the SLDS framework has greater modeling and descriptive power than LDSs, learning and inference in SLDS is much more complicated, often requiring approximations [65]. In practice, determining the appropriate number of switching states is challenging and often require large amounts of training data. Apart from maximum likelihood (ML) approaches, algebraic approaches that can simultaneously estimate the number of switching states, the switching instants and also the parameters of the model for each switch state have been proposed by Vidal, Chiuso and Soatto [66], though they are often not robust to noisy data as well as outliers.

HMMs and LDSs belong to the class of a *graphical model* or a *Bayesian network* (BN). A BN is a statistical model that encodes complex conditional dependencies between a set of random variables using a graph. Specifically, A BN consists of nodes representing random variables and directed edges representing causality relations. Dynamic Belief networks (DBNs) are a special class of the general BNs, which incorporate temporal dependencies among random variables. A HMM can be seen as one of the simplest DBNs. In general, DBNs encode more complex conditional dependence relations among several random variables than just one hidden random variable in the case of HMMs. Development of efficient algorithms for learning and inference in graphical models (c. f. [67, 68]) have made them popular tools to model structured activities [24]. Methods to learn the topology or structure of BNs from data [69] have also been investigated in the machine learning community. Other typical applications of BNs in activity-related problems include: 1) [70], which uses BNs to capture the dependencies between scene layout and low-level image measurements for a traffic surveillance setting; 2) [71], which presents an approach using DBNs for scene description at two levels of abstraction — agent level descriptions and inter-agent interactions; 3) [6], which uses BNs for multi-agent interactions where the network structure is automatically generated from the temporal structure provided by an user; and 4)[72], which proposes a pose-based hierarchical approach to recognize two-person interactions.

Petri Nets are not considered as a member of the statistical BN family, though they still find use in human motion related applications. Particularly designed for describing relations between conditions and events, Petri Nets are useful to model and visualize behaviors such as sequencing, concurrency, synchronization and resource sharing. Conditions refers to the state of an entity, and events refer to changes in the state of the entity. Petri-Nets were used by Castel et al [73] to develop a system for high-level interpretation of image

sequences and by Ghanem et al [74] as a tool for querying surveillance videos. The concept of a probabilistic Petri Net (PPN) is also introduced by Albanese et al [75].

2.3 Syntactic, Knowledge and Logic Based Approaches

Syntactic approaches for human motion recognition is a natural outcome of the success achieved by syntactic pattern recognition approaches such as *Context-free grammars* (CFG). They express the structure of a process using a set of production rules. To draw a parallel to grammars in language modeling, the production rules specify how complex sentences (human motions) can be constructed in a grammatically sound manner from simpler words (human motion primitives), and how to recognize if a given sentence (video) conforms to the rules of a given grammar (human motion model). Syntactic approaches are a good alternative for parametric approaches discussed above, when the structure of a process is difficult to learn but may be known a priori. Syntactic pattern recognition approaches were firstly applied to still-image recognition tasks, and the outstanding performance in these domains coupled with the success of HMMs and DBNs in action-recognition tasks, extend syntactic approaches into human motion recognition problems. One of the earliest uses of grammars for visual activity recognition was proposed by Brand [76], who used a grammar to recognize hand manipulations in sequences containing disassembly tasks. They made use of simple grammars with no probabilistic modeling or error analysis. Ryoo and Aggarwal [77] used the CFG formalism to model and recognize composite human activities and multi-person interactions. They followed a hierarchical approach where the lower-levels are composed of HMMs and Bayesian Networks. The higher-level interactions are modeled by CFGs.

Despite the strength of CFG, algorithms for detection of low-level primitives are frequently probabilistic in nature. Thus, *Stochastic Context-free grammars* (SCFGs) which are a probabilistic extension of CFGs were found to be suitable for integration with real-world vision modules. SCFGs appeared in Ivanov and Bobick's work [78] to model the semantics of activities whose structure was assumed to be known. They used HMMs for low-level primitive detection. The grammar production rules were augmented with probabilities and a 'skip' transition was introduced. This led to improved robustness to insertion errors in the input stream and also to errors in low-level modules. Results on surveillance videos and complex gestures of a music conductor showed promising results. Moore and Essa

[79] used SCFGs to model multi-tasked human motions – human motions that have several independent threads of execution with intermittent dependent interactions with each other, as demonstrated in a Blackjack game with several participants. In addition to these efforts, Ogale et al automatically [80] learn a probabilistic context-free grammar (PCFG) for human actions from sequences of human silhouettes. Probabilistic attribute grammars have also been used by Joo and Chellappa [81] for recognizing multi-agent activities in surveillance settings.

Different from grammar based approaches, logic and knowledge-based approaches express human motions in terms of primitives and constraints on them. These methods can incorporate far more complex constraints than grammar-based approaches. While grammars can be efficiently parsed due to their syntactic structure, logical rules can lead to a computational overhead due to constraint satisfaction checks. However, logical rules are often far more intuitive and human-readable than grammatical rules.

Specifically, logic-based methods rely on formal logical rules to describe constraints in activities. Logical rules are useful to express domain knowledge as input by a user or to present the results of high-level reasoning in an intuitive and human-readable format. Medioni et al. [82] propose a hierarchical representation to recognize a series of actions performed by a single agent. Symbolic descriptors of actions are extracted from low-level features through several mid-level layers. Then, a rule-based method is used to approximate the probability of occurrence of a specific activity, by matching the properties of the agent with the expected distributions (represented by a mean and a variance) for a particular action. In a later work, Hongeng et.al. [83] extend this representation by considering an activity to be composed of several action threads. Each action thread is modeled as a stochastic finite-state automaton. Constraints between the various threads are propagated in a temporal logic network. Shet et al [84] propose a system that relies on logic programming to represent and recognize high-level activities. Low level modules are used to detect primitive events. The high level reasoning engine then recognizes activities represented by logical rules between primitives.

In practice, whichever of the afore-mentioned approaches is used for system deployment, symbolic motion definitions are constructed in an empirical manner. Though empirical constructions are fast to design and probably work well in most cases, they are limited in their utility to the specific deployment for which they have been designed. Hence, it

turns out to be necessary for a centralized representation of human motion definitions or *ontologies* for human motions which are independent of algorithmic choices. Ontologies standardize human motion definitions, allow for easy portability to specific deployments, enable inter-operability of different systems and allow easy replication and comparison of system performance. Chen et al. [85] use ontologies for analyzing social interaction in nursing homes. Hakeem et al have used ontologies for classification of meeting videos [86]. Georis et al [87] use ontologies to recognize activities in a bank monitoring setting. Bremond and Thonnat [88] have investigated the use of contextual information in human motion recognition through domain ontologies. In the Video Event Challenge Workshops held in 2003 [89], ontologies were defined for six domains of video surveillance - 1) Perimeter and Internal Security, 2) Railroad Crossing Surveillance, 3) Visual Bank Monitoring, 4) Visual Metro Monitoring, 5) Store Security, 6) Airport-Tarmac Security. This led to the development of two formal languages - The Video Event Representation Language (VERL) [90], which provides an ontological representation of complex events in terms of simpler sub-events, and the Video Event Markup Language (VEML) which is used to annotate VERL events in videos.

2.4 Invariance in Motion Analysis

One of the most significant challenges in motion/activity recognition is to find 'invariants', i.e., quantities or descriptors that can uniquely characterize the human motion while are robust to the wide variability in features that is observed within the same activity class. Sheikh et. al. [91] have identified three important sources that give rise to variability in observed features. They are 1) Viewpoint, 2) Execution Rate, and 3) Anthropometry. Any real-world human motion recognition system should aim to discover invariants to these factors. In this section, we will review efforts and contributions in these areas.

A fundamental problem in video-based recognition of human motion/activity is achieving *view invariance* when representing these human motions. In general, it may be easy to establish statistical models of simple actions based on the representations discussed so far from a single view, but it is absolutely non-trivial to generalize them to other views even for very simple action classes. This is due to the wide variations in motion-based features induced by camera perspective effects and occlusions. A straight forward solution to this problem is to keep templates from several canonical views as done by Bobick et al. [33] and

interpolate across the stored views as proposed by Darrell, Essa and Pentland [92]. This approach however is in general not scalable since one does not know how many views to consider as canonical. Another approach is to assume that point correspondences across views are available as in Syeda-Mahmood et al. [39] and compute a transformation that maps a stored model to an example from an arbitrary view. Seitz and Dyer [32] present an approach to recognize cyclic motion that is affine-invariant by assuming that feature correspondence between successive time-instants is known. It is shown by Rao and Shah [93] that extrema in space-time curvature of trajectories is preserved across views. The extrema in space-time curvature of hand trajectories are denoted as 'dynamic instants'. An action is then considered as a sequence of dynamic instants which is preserved across several views. Another example is the work of Parameswaran and Chellappa [94, 95], where a view invariant representation of actions is defined based on the theory of 2D and 3D invariants. In this approach, an action is considered to be a sequence of *poses*, and it is assumed that there exists at least one *key-pose* in the sequence in which five points are aligned on a plane in the 3-D world coordinates. Using this assumption, a set of view-invariant descriptors is derived. More recently, Weinland et al [96] extend the notion of motion-history [33] to 3-D, where multiple views are combined from multiple cameras to obtain a three-dimensional binary occupancy volume. Motion history is computed over these 3-D volumes and view-invariant features are extracted by computing circular FFT of the volume.

The issue of *Rate Invariance* is brought by another major source of observed variability in features arising from the differences in execution rates while performing the same action. Variations in execution style exist both in inter-person and intra-person settings. State-space approaches have robustness toward minor changes in execution rates, but are not rigorously rate-invariant since they do not explicitly address transformations of the temporal axis ((c. f. Bobick and Wilson [97], Hoey and Little [98])). Mathematically, the variation in execution rate can be described with a warping function of the temporal scale. The simplest case of linear time-warps can be usually dealt with fairly easily (c. f. [33, 99]). For those highly non-linear warping functions, typical methods are based on Dynamic Time Warping of the feature sequence (c.f. [100, 92, 101, 102, 103]).

Anthropometric Invariance aims to minimize the negative effect from variations induced by the size, shape, gender etc. of humans, which is another important class of variabilities that requires careful attention. Unlike viewpoint and execution-rate variabili-

ties which have been well-studied, a systematic study of anthropometric variations has only been receiving limited interests until in recent years. Ad-hoc methods which normalize the extracted features to compensate for changes in size, scale etc. are usually employed when no further information is available. Drawing on studies on human anthropometry, Gritai et al. in [104] suggested that the anthropometric transformation between two different individuals can be modeled as a projective transformation of the image co-ordinates of body joints. Based on this, they define a similarity metric between actions, by using epipolar geometry to provide constraints on actions performed by different individuals.

2.5 Multi-Object Cooperative Group Motion

As reviewed up to this point, human motion analysis and classification has been a research focus of computer vision community for at least two decades, and significant contributions have been emerging over the years. However, most of these previous works focused on single object cases, where the motion and dynamics of an individual object are investigated. Activities of multiple objects exist widely in surveillance, sports, and biological observation records, *etc.*, and consequently modeling and analysis of multi-object activities will be useful in these applications. Although the multi-object tracking problem has been extensively studied [15], much less attention has been paid to putting the tracked motion pattern of the whole group in a learning and recognition framework. Limited work on multi-object coordinated group motions has been done only in the context of limited objects and highly structured motion patterns, as will be briefly discussed below.

In a less complex scenario, where the individuals in a group undergo a structurally fixed motion [1] (see Figure 2.1) or follow identical dynamics or trajectories [2] (see Figure 2.2). In the former [1] only a deviation from a modeled formation is detected; in the latter [2], an 'abnormal' activity is claimed when the configuration of the individuals exceeds an admissible bound. However, more meaningful semantics may be extracted for less-structured but coordinated activities. In other words, the individual objects will have distinctive and varying motion patterns but the group collectively demonstrates an underlying activity with an explicit sematic identity. A most illustrative example is a football game, in which we would like to recognize the strategy used in each play rather than individual players' movements. Similar examples/applications exist in other domains, *e.g.*, activities of a group of social insects like bees [105].

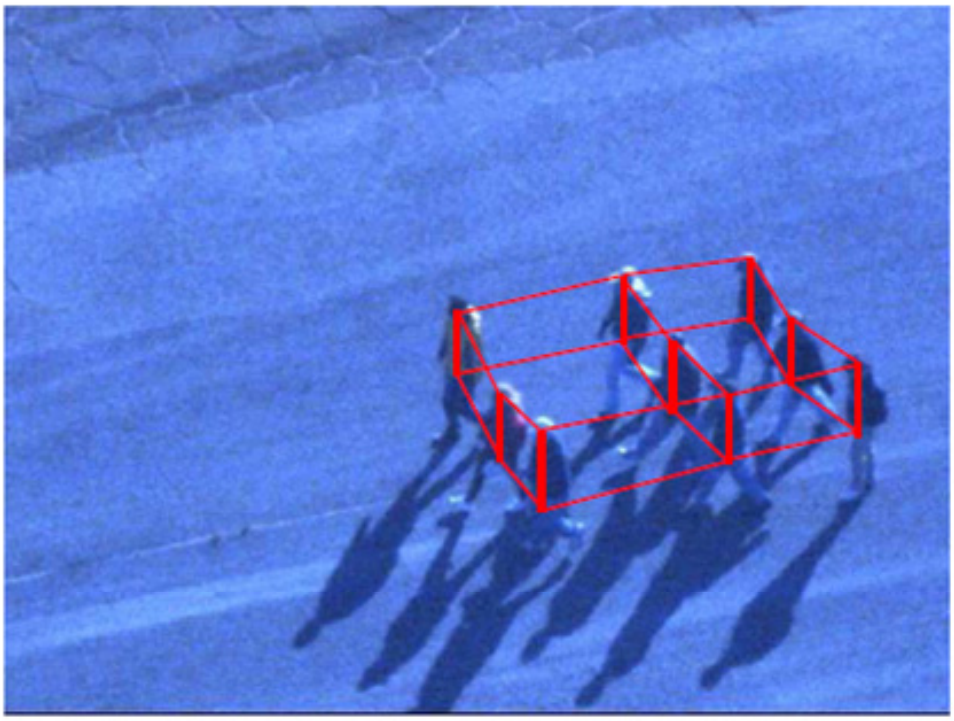


Figure 2.1: The structural group motion configuration considered in [1].

(a) A 'normal activity' frame with 4 people

(b)Abnormality introduced by making one person walk-away in an abnormal direction

Figure 2.2: 'Shape activity' for anormaly detection in [2].

Before the investigation into large-scale coordinated sport games or social behaviors, efforts have also been made on common group activities involving a smaller number of objects. Cases include [106] which employs the causality of time-series to describe and recognize pairwise activities, and [107] which looks into the multi-linear property of a multi-dimensional tensor stacked from the trajectory ensemble arising from the activity participants. [3] follows the line of causality analysis and focus on the localized temporal effects over short period in a long-term complex interaction (see Figure 2.3). [108] also makes use of low-level motion features to discover events of interest in space and time. The work closely related to our contribution [4] characterizes the interactive pattern with a descriptor, which essentially encodes the spatio-temporal co-occurrences of atomic action elements as in Figure 2.4.

A group activity usually occurs according to a planned goal. In each football play, the offensive players will collaboratively follow a pre-determined strategy. The individual action of each player, meanwhile, is also a result of interaction with and response to the motion of other players. The same interpretation can be also made for conventional everyday interactive behaviors. A specific group activity pattern, therefore, is determined by the *interactions* among objects and their *temporal evolution*. Modeling and recognizing the temporal inter-object relationship (*i.e.*, the group activity pattern) has been naturally handled using a Bayesian network framework [109, 5, 110, 111, 112]. Figure 2.5 illustrates the dynamic Bayesian networks used in [5].

Bayesian network formulation, though successfully applied to modeling activities of single object or motion, is in a new position and has new challenges when dealing with group activities. Intuitively, the spatio-temporal relationships among multiple agents easily fall into the scope of dynamic Bayesian network, which is exactly supposed to explicitly encode the causal constraint among the individual actions or elementary evens. On the other hand, however, to completely characterize the roles of individual objects, their action primitives, interactions, and overall plan, the complexity of the network turns out to be high. This inherent difficulty manifested itself in previous work (*e.g.*, [6], Figure 2.6), where individual objects' identities, roles and their individual action primitives were pre-labeled. As simultaneous recognition of individual actions and group activity pattern is computationally intensive, the number of objects considered previously, was usually small, which is not the case for a large group as a football team. Compared to the size of the state

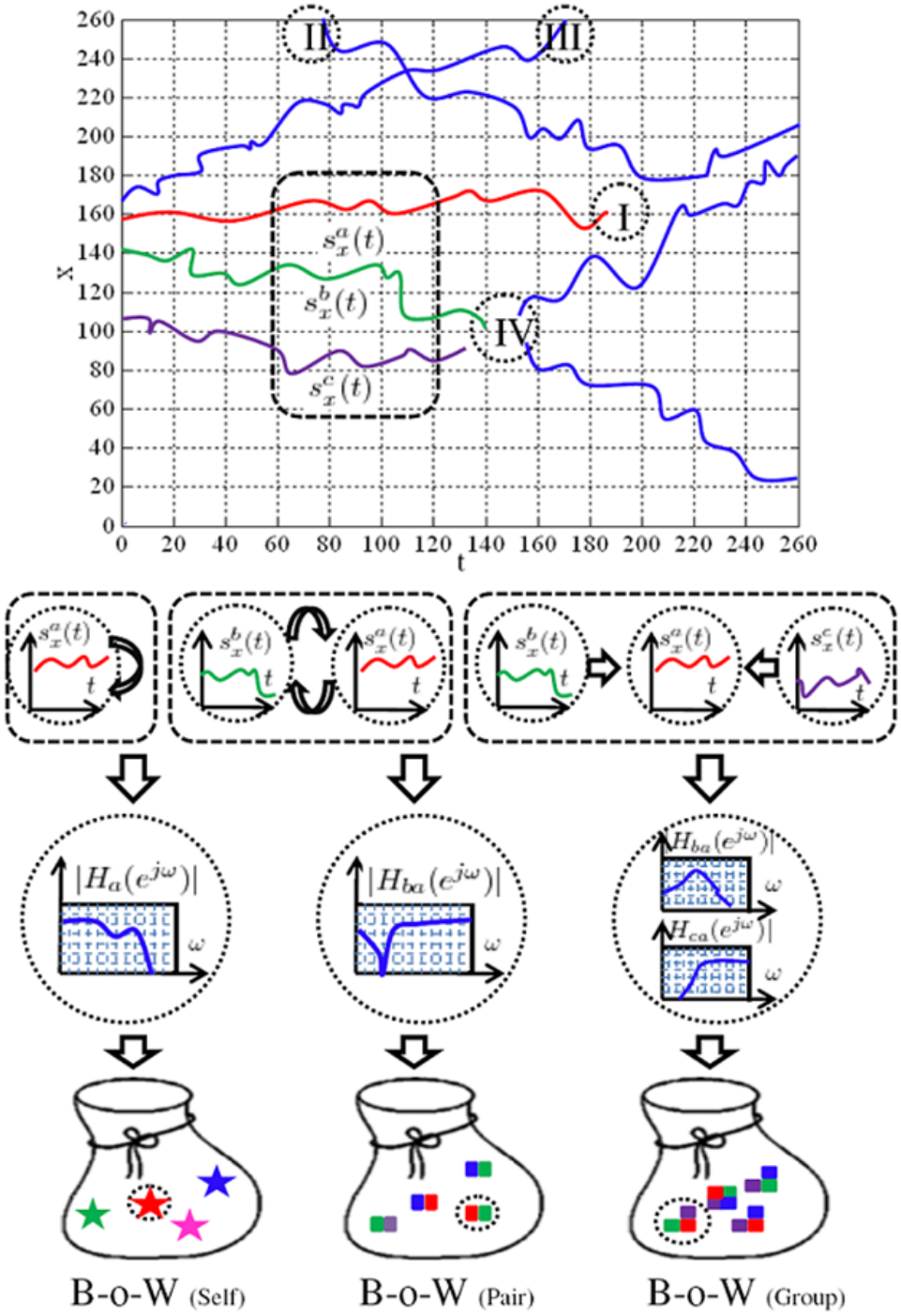


Figure 2.3: Spectral feature extraction for causality analysis of multi-trajectories in [3].

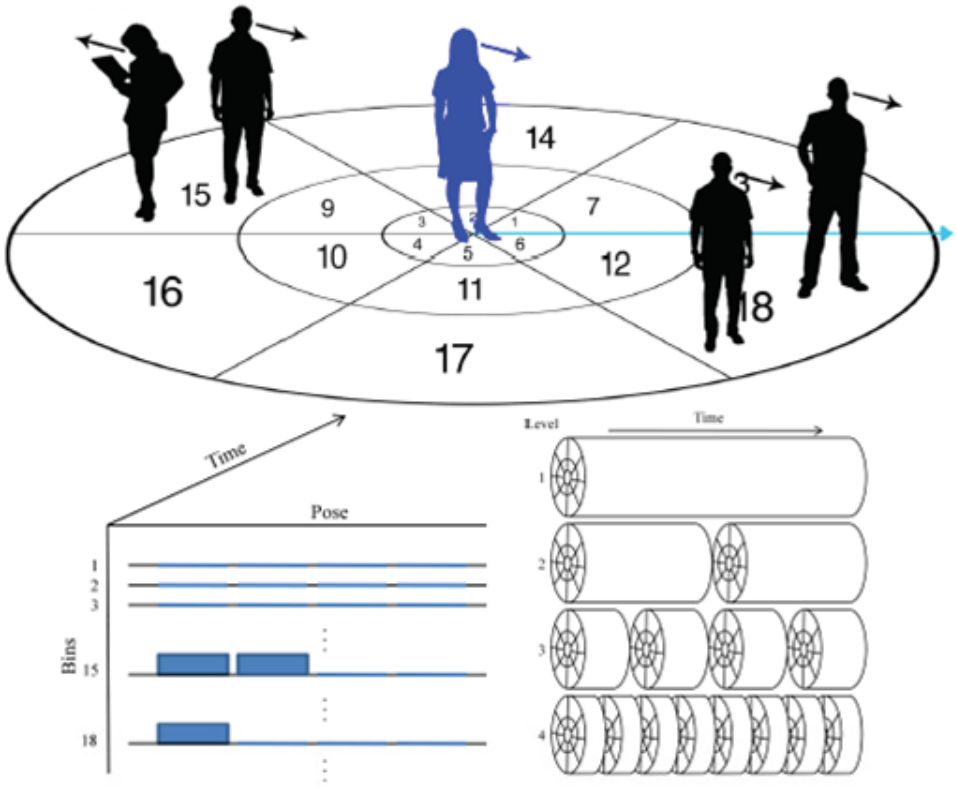


Figure 2.4: The design of contextual group activity descriptor in [4].

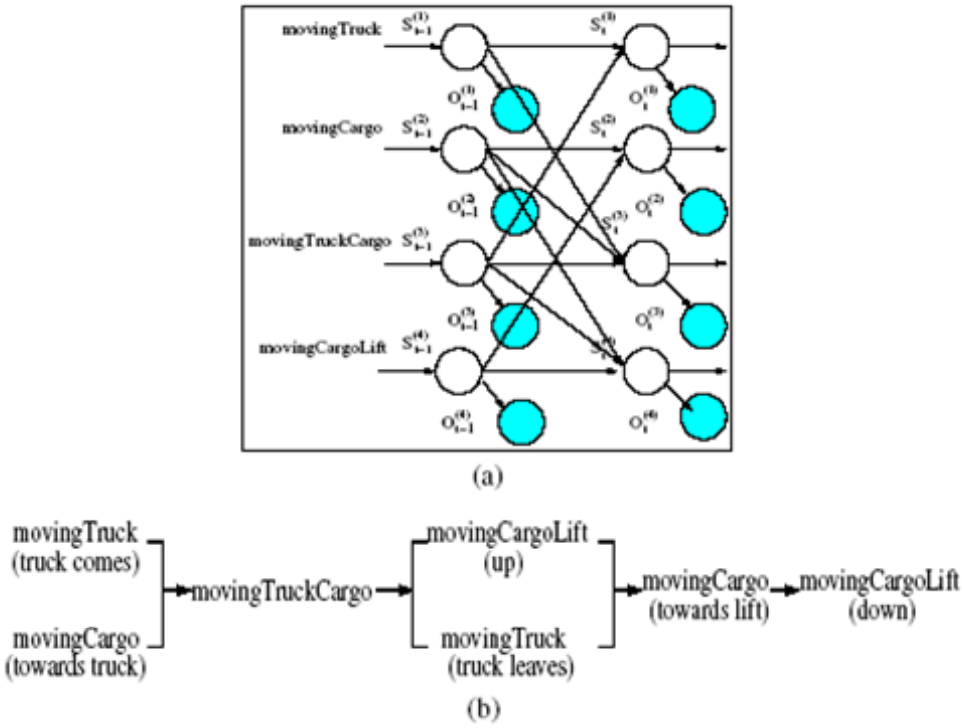


Figure 2.5: Bayesian network for outdoor multi-agent activites in [5].

space and feature space of the network, the amount of available training data is insufficient. Thus not only the probabilistic dependence might very possibly be 'over-fitted', but also necessary priors are hard to learn from available data.

The difficulty on model scalability is alleviated to some extent in a recent effort [7] where temporal modeling is eliminated and replaced by hierarchical modeling of the overall collective motion, individual motion, and low-level features, while relationship among entities is still explicitly manipulated, as in Figure 2.7. Syntactic construction using a grammar or rules [113] is another alternative, while fine-scale modeling demands addressing the challenges mentioned above.

Extensive work, meanwhile, has been done in sports video analysis, especially for football or soccer [114, 115, 116]. These efforts attempted to detect or recognize specific types of semantics in the games using camera motion, color, low-level motion, field markers, lines, texture, and so on, but did not treat the plays as multi-object group activities. The perspective to treat motion patterns in sport videos as multi-agent group activities is

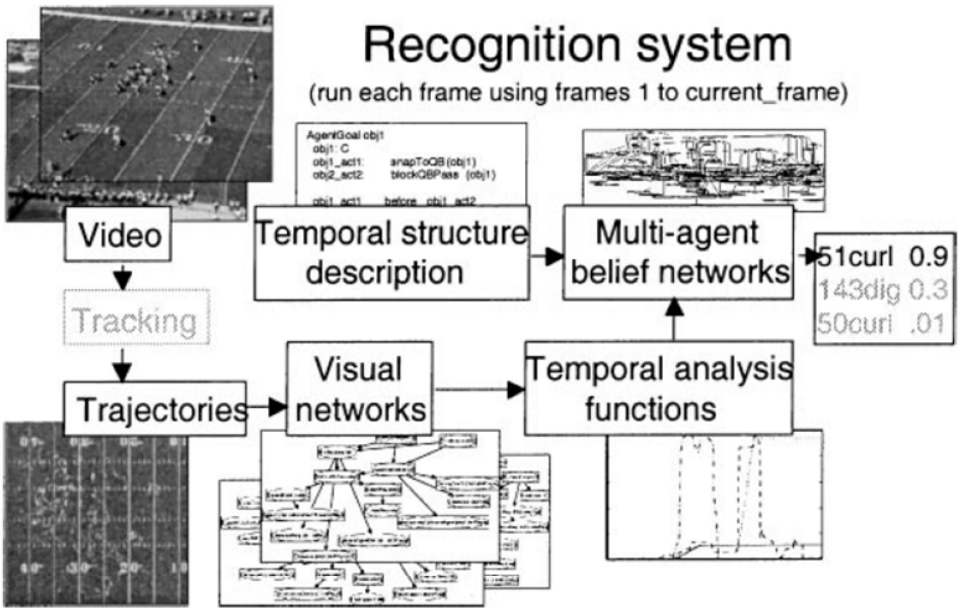


Figure 2.6: Bayesian network pipeline for football play recognition in [6].

demonstrated in [6, 117], though the formalism falls into the category of parametric probabilistic Bayesian networks again. Simplified statistical mechanism for trajectory segments is proposed for basketball games [118] by representing rules and heuristics from basketball domain into the probabilistic terms. Recent companion work [8] (Figure 2.8) uses logic network instead, which turns out to be effective and efficient when capturing the coordinations in a basketball game.

2.6 Differential Geometry in Computer Vision

An important contribution of this dissertation from the theoretical perspective is the exploitation of differential geometry, or more specifically the study of analytical manifolds, for representing visual motion. For this purpose we also summarize the state-of-art of applications of geometric models and methods in computer vision. The material in this section is based on [119], a recent survey on statistical analysis on differential manifolds.

Non-Euclidean treatment of visual data observed in practice has been considered for many decades. The Euclidean motion group plays a fundamental role in rigid body dynamics and uncertainties in modeling dynamic systems have been characterized using

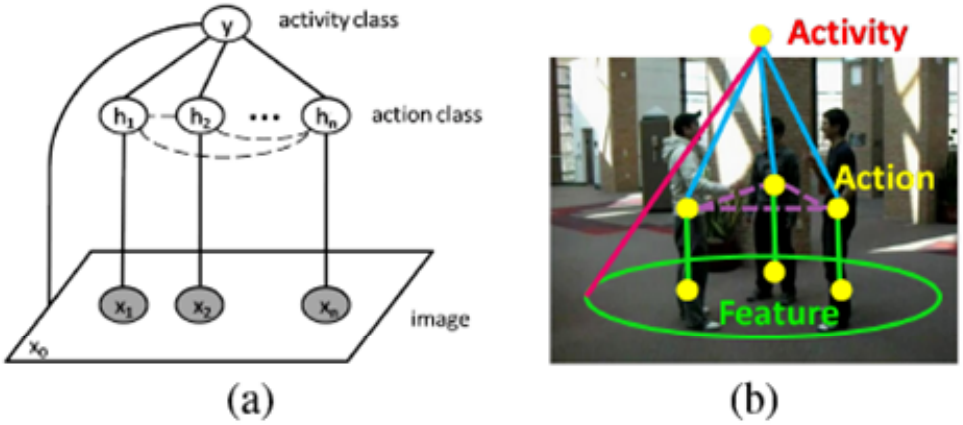


Figure 2.7: Hierarchical graphical relationship among group motion, individual motion, and features proposed in [7].

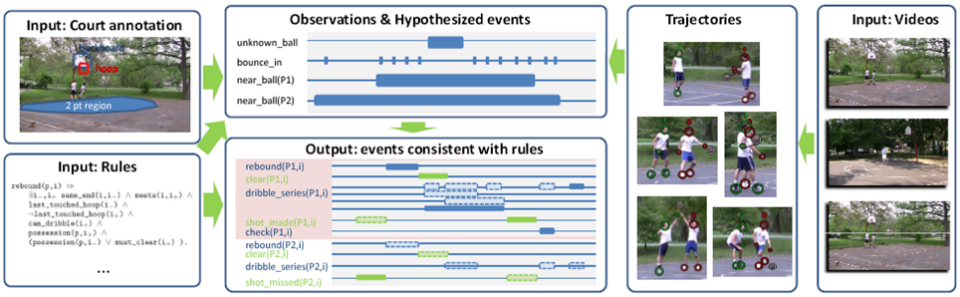


Figure 2.8: Framework of rule based modeling of basketball games in [8].

probability measures on this group. Researchers have combined the strengths of geometry and statistics in the area of stochastic control [120, 121] where system variables and controls are constrained to be on certain non-Euclidean manifolds. The first major effort in using geometry and statistics in pattern recognition, meanwhile, was introduced by Ulf Grenander in the early 70s [122, 123]. Grenander created the field of pattern theory which has the following important components: 1) represent the systems of interest using algebraic structures that favor rule-based compositions, 2) capture variability in these systems using probabilistic super-structures, and 3) develop efficient algorithms for inferences using geometries of underlying spaces. Over the last three decades, this philosophy has been implemented in a number of contexts with explicit involvement of statistics on non-Euclidean manifolds. The work on analyzing anatomical variability using noninvasive imaging (such as MRI, PET, etc) involved probabilistic structures on high-dimensional deformation groups [124, 125]. An algebraic pattern theoretic approach has not been exclusive to medical imaging only. It has also been used in addressing computer vision and image analysis problems. For example, in the problem of recognizing objects in images, the variability due to viewing angle of the camera is very important. [126] deals with the problem of estimating the pose as an element of SO(3) and that of bounding the estimating error using statistical bounds. [127] studies the problem of using Markov Chain Monte Carlo methods for performing estimation on some matrix Lie groups e.g. SO(n), and their quotient spaces, e.g. a Grassmann manifold, while [128] studies the problem of subspace tracking (in signal processing) as a problem of nonlinear filtering on a complex Grassmann manifold. While these efforts involve statistical inferences on manifolds, there is a strong literature on more general optimization problems. For example, a major work in the area of optimization algorithms on Grassmann and Stiefel manifolds was presented by Edelman et. al. [129, 130].

Another prominent area that employed statistical models and inferences on non-Euclidean manifolds is shape analysis. Starting with a trend-setting paper by Kendall [28, 131], there has been a remarkable progress on representing and analyzing shapes of objects, in images or otherwise, using a landmark based approach. In terms of statistical analysis, this is perhaps the most mature area involving manifolds as domains [14, 132]. In more recent years, there has been an extension of Kendalls shape theory to infinite-dimensional representations of shapes of curves and surfaces [133, 134, 135]. The area of statistics and inference on manifolds has seen a large growth in recent years. Many of the ideas have

been formally introduced and advanced through the efforts of many researchers. One of the landmark works in establishing mean estimation and central limit theorems for manifold-valued variables is Bhattacharya and Patrangenaru [136, 137]. Another important piece of work comes from Pennec [138] who has applied these notions for detection and classification of anatomical structures in medical images. Recent applications in computer vision have included study of Kendalls shape spaces for human gait analysis [53], and Hilbert sphere modeling of time warp functions for human activities in [139]. Other applications include classification over Grassmann manifolds for shape and activity analysis [140, 141], and face recognition [142]. A recently developed formulation of using the covariance of features in image-patches has found several applications such as texture classification [143], and pedestrian detection [144]. Mean-shift clustering has been extended to general Riemannian manifolds in [145].

Chapter 3

Statistical Model on Discriminative Temporal Interaction Manifold for Group Motion Recognition

3.1 The Group Motion Recognition Problem

In this chapter we present a method for modeling and recognizing collaborative group motions involving multiple objects from videos by designing and exploiting a temporal descriptor. We use football play modeling and recognition as our primary example and in particular test our algorithm on football videos. The work [6], most similar to ours, designed large connected Bayesian networks for football play recognition. In contrast, we explore a 'data-driven' approach. Here we only assume that the players' roles and their motion trajectories are already available. For the former, we may recognize the roles from the initial play formation with the help of landmark shape theory [14], and for the latter we may employ a multi-object tracker [15]. These two problems are still being researched and are beyond the scope of this chapter.

A central issue is the level of complexity needed for describing the interaction patterns in a coordinated group activity, for the purpose of recognition. Is it possible that a compact and discriminative descriptor, instead of large-scale probabilistic schemes, exists and can be learned from videos? This work is motivated by these two fundamental considerations. Specifically, we describe a group activity pattern with a full four-dimensional object-time interaction tensor, and learn an optimized tensor reduction kernel to condense it to a discriminative temporal interaction matrix. The temporal interaction matrix serves as the compact descriptor for the group activity pattern, and is stable under view changes. More importantly, the temporal interaction matrices are generic for various activity representations. For videos of football games, from which we extract point trajectories of the players, simple dynamic features give rise to the quantitative terms to measure interactions. For videos with multiple articulated human actions, individual action descriptors will be integrated in a particular way. We will show different approaches to reach the object-time interaction array with different inputs, while the temporal interaction matrices will be the unified output representing the group motion feature.

A conventional deed with descriptors in hand is to connect them directly with a learning machine, which can be typically a discriminative one such as K-Nearest Neighbor, a supporting vector machine, or a boosting based classifier. Instead, one may integrate the descriptors into simple generative mechanisms including, for example, naive Bayesian, probabilistic latent semantic analysis, or latent Dirichlet allocation. No effort, to the best of our knowledge, has been devoted to an exploitation of the space of the descriptors, and to make use of the structure, or geometry, of the possibly non-linear manifold. Our second contribution is to identify a Riemannian metric for the set of all temporal interaction matrices which form a Riemannian manifold. On this manifold we are able to establish a probabilistic framework to characterize every class of group activity pattern. We call this manifold Discriminative Temporal Interaction Manifold (DTIM). To learn a multi-modal 'likelihood' density for each class, we create a basic exponential density component on the manifold, and incrementally build up the complete manifold-resided densities with the basic components. With the established framework, a MAP classifier is used to recognize a new group activity.

The rest of this chapter is organized as follows. In Section 3.2 we obtain a view-stable and discriminative temporal interaction matrix, via an optimized tensor reduction, to compactly characterize each group activity. We will, in particular, show how different motion features arising from different scenarios are unified under the same framework, and discuss how we can achieve robustness when we have noisy input. Then in Section 3.3 we focus on the space of temporal interaction matrices, *i.e.*, DTIM, and in particular create a basic probability density on this non-linear manifold by exploiting its geometric property. To account for possibly multi-modal likelihood distribution of temporal interaction matrices on DTIM, in Section 3.4 we introduce an incremental, or boosting procedure to build the complete model with the basic components. Finally, we show the performance using data from football plays and everyday common interactive behaviors in Section 3.5. See Figure 3.1 for a general flow chart of the proposed approach.

3.2 View-Stable Discriminative temporal interaction matrix

As mentioned, a coordinated group activity pattern is characterized by the temporally evolving interactions among objects. To describe mathematically the interaction, we start from the object-time interaction tensor as $Y(t_1, t_2, p_1, p_2)$, where $1 \leq t_1, t_2 \leq T$, $1 \leq p_1, p_2 \leq$

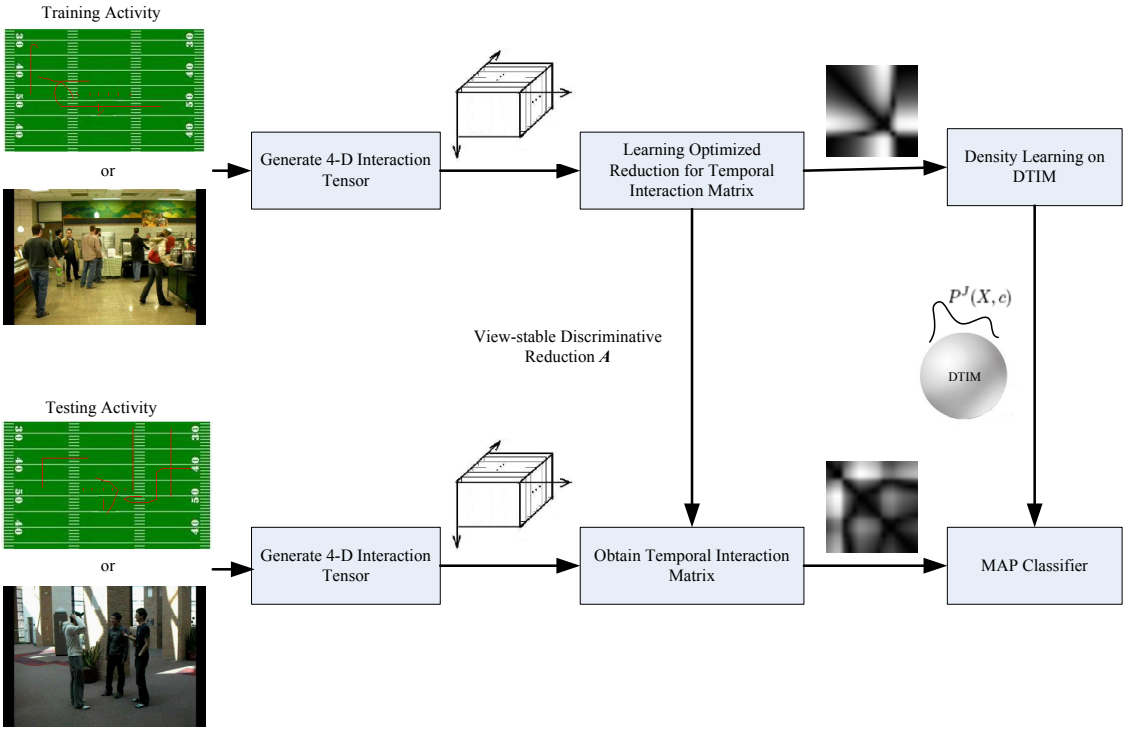


Figure 3.1: The flowchart of the modeling and recognition framework.

P. Here T is the duration during which we observe the group activity, and P is the total number of objects involved in the activity (*e.g.*, the total number of players in a football play). The scalar term $Y(t_1, t_2, p_1, p_2)$ describes the 'interaction' between object p_1 at time t_1 and object p_2 at time t_2. The rationale of this construction lies in the consideration that causal correlations in general exist between any pair of participants in a group activity and between any pair of time instants during the occurrence of the event. By letting t_1, t_2 run through the whole time span and letting p_1, p_2 touch all involved objects, Y is able to capture all the pairwise correlations. A natural question is whether all these information is critical to describe a group activity and which are the critical ones if so. By reducing the object-time interaction tensor to temporal interaction matrix as to be described below, we actually select the significant and discriminative information in an automatic manner. On the other hand, the scalar $Y(t_1, t_2, p_1, p_2)$ serves as a quantity numerically encoding the interaction, and its physical or semantic interpretation will be clear in the context of the specific motion feature used. (One may think that a vectorial quantity may be possibly more powerful in capturing a full set of information available: As an initial attempt to formulate a group motion descriptor, we employ a scalar.) We will detail how to compute the interaction tensor for different applications shortly, before which we first discuss the optimal tensor reduction, *i.e.*, the approach to reach temporal interaction matrix - the group activity descriptor.

The four-dimensional tensor Y is possibly a redundant descriptor for the activity pattern and now we now seek a more compact and discriminative descriptor, namely, the temporal interaction matrix $X(t_1, t_2)$ via a tensor reduction mapping $R : Y \mapsto X$. The motivation for this reduction, in addition to those mentioned above, is two-fold. On the one side, as the tensor is in a high dimensional space, a dimensionality reduction step is generally necessary so that the need for many training samples may be reduced. On the other side, more importantly, the temporal interaction matrix empirically turns out to be quite stable under view changes, though it is not strictly view-invariant. Figure 3.2 shows this 'view-stability' of the temporal interaction matrix obtained from the discriminative tensor reduction method presented below. Figure 3.2(a) and 3.2(b) show the players' motion trajectories for the same play under different views, and Figure 3.2(c) and 3.2(d) show the corresponding temporal interaction matrices, which appear quite close to each other. Although the example is synthesized from a play diagram, the same behavior is

also observed for real trajectories. The search for a discriminative and view-stable matrix descriptor is also inspired by the previous work on video self-similarity [146] and a recent one [147], in which view-stability is observed in 'self-similarity matrix' of point trajectories for a single person action.

Instead of using an arbitrary tensor reduction mapping R, here we use a $P \times P$ matrix reduction kernel A. Let $Y(t_1, t_2)$ to be the $P \times P$ matrix sliced from tensor $Y(t_1, t_2, p_1, p_2)$ when t_1, t_2 are fixed, and we define

$$x(t_1, t_2) = tr(A^T Y(t_1, t_2)) \tag{3.1}$$

and

$$X = R(Y) = \frac{x}{\|x\|}. \tag{3.2}$$

Note that both $Y(t_1, t_2)$ and A are symmetric due to the interpretation of 'interaction' above. Similar notion of symmetry also holds for X. The normalization of x helps to maintain a constant scale for X.

Note that A in fact weighs each element of $Y(t_1, t_2)$ with its corresponding element. Therefore, A serves as a pairwise interaction selector, which emphasizes the interaction between the ith and the jth objects if $A(i, j)$ is large. If we nomially take A to be the identity matrix, *i.e.*, discarding the interaction between different objects but only keeping the objects' self-motion, then the resulting temporal interaction matrix X is essentially equivalent to the one used in [147]. Therefore, the question is: is there another A (or weighting pattern) other than the identity matrix, which can achieve intra-class view-stability as well as better inter-class separability? To get such an optimized tensor reduction kernel, we mathematically enforce view-stability and separability in our optimization target as described below. In other words, we formally look for view stability instead of achieving it in an ad-hoc manner.

From now on we use different subscripts to denote temporal interaction matrices from different sample group activities. A pairwise similarity between the kth sample X_k and the lth sample X_l, $s(k, l)$, is defined as

$$s(k, l) = tr(X_k^T X_l) = \frac{< x_k, x_l >}{\|x_k\| \|x_l\|}. \tag{3.3}$$

Then the target function to be maximized is defined as

$$J(A) = \sum_k (\alpha\beta \sum_{l \in C_1(k)} s(k,l) + \alpha(1-\beta) \sum_{l \in C_2(k)} s(k,l) - (1-\alpha) \sum_{l \in C_3(k)} s(k,l)) \tag{3.4}$$

where $C_1(k)$ is the set of same activities as k but from possibly different views, $C_2(k)$ is the set of activities different from k but belonging to the same class as k, and $C_3(k)$ is the set of activities not belonging to the class of k. By maximizing $J(A)$ with respect to A with the controllable parameters $0 < \alpha, \beta < 1$, we are able to find an optimized A such that the cross-view similarity and intra-class similarity are both maximized while the inter-class similarity is minimized. In other words, the resulting A will weigh the interactions between every pair of objects properly such that view-stability and class separability are simultaneously achieved.

To perform the above maximization, we take a gradient ascent based approach due to the non-linearity of the target function with respect to A. To evaluate $\nabla_A J(A) = \frac{\partial J(A)}{\partial A}$, we evaluate $\frac{\partial s(k,l)}{\partial A}$. After some calculations it can be shown that

$$\frac{\partial s(k,l)}{\partial A} = \frac{1}{\|X_k\|\|X_l\|}(\sum_{t_1,t_2}(X_k(t_1,t_2)Y_l(t_1,t_2) + X_l(t_1,t_2)Y_k(t_1,t_2)) - b(k,l)) \tag{3.5}$$

where

$$b(k,l) = tr(X_k^T X_l)(\frac{\sum_{t_1,t_2} X_k(t_1,t_2)Y_k(t_1,t_2)}{\|X_k\|^2} + \frac{\sum_{t_1,t_2} X_l(t_1,t_2)Y_l(t_1,t_2)}{\|X_l\|^2}). \tag{3.6}$$

The optimization steps are implemented as repeated line searches along the gradient directions specified in (5) and (6). Because a global maximum is not guaranteed, we initialize A from multiple symmetric matrices in multiple optimization processes and pick out the one yielding the maximum $J(A)$.

3.2.1 Interaction Tensor from Point Trajectory Ensemble

We now discuss how we may compute the object-time interaction tensor from different features available. If the involved objects are represented as mass points and their motion

across frames are in the form of point trajectories, we are in a position to compute Y from the ensemble of trajectories. We assume that the objects are ID-specific, *i.e.* the football players' roles are associated with each trajectory. In practice, we need to find the IDs for the objects. For this purpose, in football play videos we may recognize the roles from the initial play formation with the help of landmark shape theory [14], or use simple heuristics such as labeling the objects using the coordinates of their initial locations, from the left of the field to the right side. In our experiments, we use the existing IDs provided with the dataset. We also assume their motion trajectories are already available. Practically, to get trajectories we may employ a multi-object tracker such as any appropriate one introduced in [15]. In our experiments, we used the one we developed [9]. Automatic tracking is usually non-robust and its output tends to be noisy. We will discuss how to handle non-robust motion information in the third subsection, but assume that clean tracks are available for now.

Denote the motion trajectory for object p as $T_p(t)$, regarded as a one-dimensional curve indexed by t, we may simply use the Euclidean distance (in the image plane) between p_1 at time t_1 and p_2 at time t_2 as $Y(t_1, t_2, p_1, p_2)$, *i.e.*,

$$Y(t_1, t_2, p_1, p_2) = \|T_{p_1}(t_1) - T_{p_2}(t_2)\|. \tag{3.7}$$

Alternatively, we may consider first-order information $\frac{\partial T_p(t)}{\partial t}$, in which case the 'interaction' is the inner-product of the two velocities of the objects, *i.e.*,

$$Y(t_1, t_2, p_1, p_2) =< \frac{\partial T_{p_1}(t_1)}{\partial t_1}, \frac{\partial T_{p_2}(t_2)}{\partial t_2} > . \tag{3.8}$$

A similar approach can be applied to the second-order information (acceleration), or even higher order. For a more sophisticated consideration one may use a combination of the quantities across multiple orders. In our experiments we used the distances and velocities individually, and it turned out the performance of the accuracy is not sensitive toward the two choices.

Samples of the computed temporal interaction matrices are given in Figure 3.2.

3.2.2 Interaction Tensor from Articulated Human Action Images

When high-resolution images of the involved articulated humans are available, we exploit the low-level local features extracted from images. For this purpose we assume a

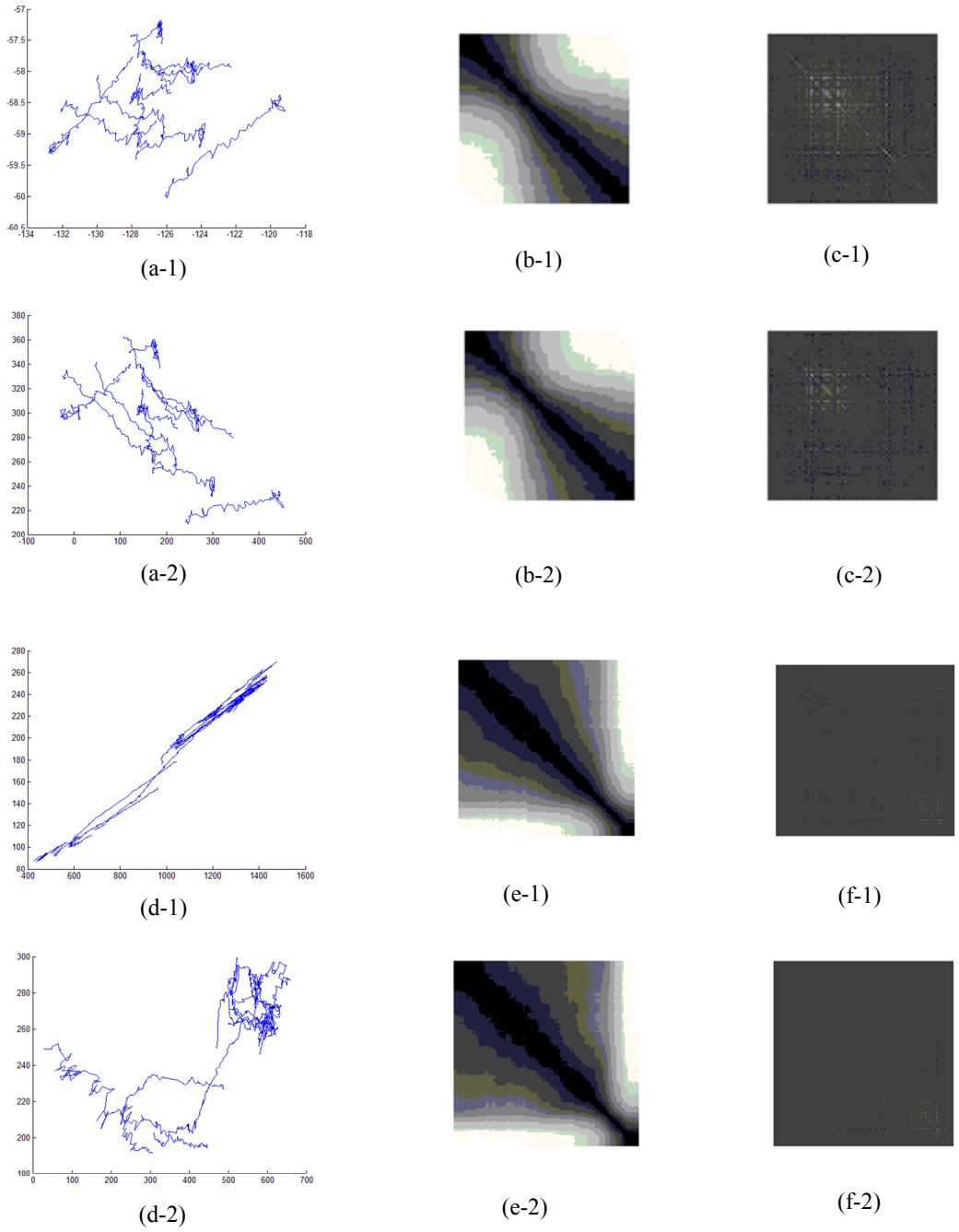


Figure 3.2: (a-1)(a-2): Players' trajectories from different view points of sample U60 in GaTech Football Play Dataset; (b-1)(b-2): The corresponding temporal interaction matrices obtained from pairwise distances using the view-stable optimization; (c-1)(c-2): The corresponding temporal interaction matrices obtained from pairwise correlations of velocities using the view-stable optimization; (d)-(f): The same set of illustrations on sample V20 in the same order as (a)-(c).

bounding box is available for each object/human, through human detection and/or tracking. Again additional effort to handle noisy bounding boxes will be discussed separately in the next subsection, while we regard the boxes here are clean and reliable.

Within each bounding boxes, various low-level local features are extractable. One may, for example, fix a spatially regular grid (which may be as dense as pixels or as sparse as partitioned rectangular blocks), and then compute optical flow, local tracks estimated using a KanadeLucasTomasi feature tracker, spatio-temporal gradient, or spatio-temporal SIFT descriptors. The 'interaction term' $Y(t_1, t_2, p_1, p_2)$ may be simply taken as the direct difference between the feature vectors, averaged or accumulated from within the two bounding boxes (with sizes normalized, if necessary). The same type of feature can also be extracted around spatio-temporal interest points instead of a regular grid.

A more noise-resistant and compact representation of the low-level local features is the histogram of them. By this approach we apply vector quantization on the local descriptors and reach a 'code book'. Then motion in each bounding box is represented by a histogram encoding the distribution of the quantized 'visual words' in that box. Denote the histogram for object p at time t as $H_p(t, b)$, where $b = 1, 2, \cdots, B$ and B is the total number of bins of the histogram. Then, a natural choice for $Y(t_1, t_2, p_1, p_2)$ originates from χ^2 kernel, *i.e.*,

$$Y(t_1, t_2, p_1, p_2) = 1 - \sum_{b=1}^{B} \frac{(H_{p_1}(t_1, b) - H_{p_2}(t_2, b))^2}{\frac{1}{2}(H_{p_1}(t_1, b) + H_{p_2}(t_2, b))}. \tag{3.9}$$

Mid-level implicit features are also investigated, where the histograms of the low-level features are input into simple learning machines such as Support Vector Machines, which output the likelihood of each bounding box containing a particular type of individual action [7] or a particular pose [4]. In this case, $H_p(t, b)$ is the probability of object p executing action b or posing toward direction b at time t, and we just compute $Y(t_1, t_2, p_1, p_2)$ as

$$Y(t_1, t_2, p_1, p_2) = \sum_{b=1}^{B} H_{p_1}(t_1, b) H_{p_2}(t_2, b). \tag{3.10}$$

Samples of the computed temporal interaction matrices using different local features and distance measures are given in Figure 3.3, for the datasets used in [4] and [7].

3.2.3 Handling Non-Robust Input and Other Issues

The input to object-time interaction array is usually obtained from detection/tracking. Though noisy input has been a common phenomenon, eventually it is still the responsibility

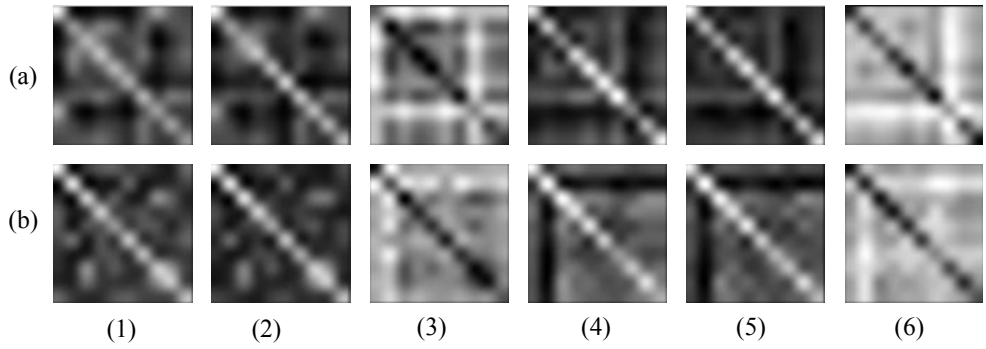


Figure 3.3: (a): temporal interaction matrices for a sample of the activity 'crossing'; (b) temporal interaction matrices for a sample of activity 'talking'. (1) using Chi-square distance between histogram of flows; (2) using cosine distance between histogram of flows;(3) using Euclidean distance between histogram of flows;(4) using Chi-square distance between histogram of oriented gradients;(5) using cosine distance between histogram of histogram of oriented gradients;(6) using Euclidean distance between histogram of histogram of oriented gradients.

of detection module and tracking module to improve their performance so that complete, continuous, and reasonable trajectories and/or bounding boxes are provided. However, we do take non-robust input into account in our effort, as to be discussed now. Approaches to handle noisy input are application specific. We now look into the cases that we experiment with.

In football plays, colors suffice for differentiating offensive players from all players in the field. When players are far apart from each other, the eleven offensive players will be correctly identified and tracked, while the main issue is that tracks are missing when occlusion/overlap happens and individual players are connected into large blobs. We begin from the initial formation, follow through the initial period with eleven tracks, and stop at the time instant when the number of tracks reduces. We then compute the displacements between each of the previous eleven trajectories and each of the afterward trajectories, and apply Kuhn-Munkres optimal assignment algorithm [148] to find the best association for each of the afterward trajectories with the previous ones. For those non-matched previous ones, we first identify those afterward blobs of large sizes, and then assign them to the center of the nearest blob. We keep associating trajectories at every time instant when the

number of tracks changes until we reach the end of the activity. This strategy to generate a complete set of eleven trajectories is largely heuristic, but turns out to be effective as in the experiments section.

A similar strategy can be used toward everyday collective activities, such as crossing, waiting, queueing, talking, dancing, and jogging, as investigated in [4]. We identify the total number of participants in the group activity by checking the detections in all frames, locate the segments with the right number of participants, and check neighboring frames forward and backward. For any segment with a constant number of participants, which is however less than the correct number, we associate the bounding boxes in them with those in neighboring segment with the desirable number, by comparing pairwise similarity in terms of color histogram and spatial proximity. For any non-paired bounding boxes, we perform local search around the detected/tracked location/size and look for the best matched color histogram. In this way, we expect a constant number of reasonable bounding boxes throughout the duration of the activity.

If the same category of activity involves varying numbers of participants, *i.e.*, varying P's, then a separate interaction tensor and reduction should be considered for each P. As a unique ID, or labeling, is necessary for each participant to compute the interaction array, we label the objects according to their initial locations in the image. We project the centers of initial bounding boxes onto the directed line confirming to the overall dominant motion direction of all boxes, and order the boxes from the smallest projected number to the largest.

3.3 Basic Exponential Distribution on the Discriminative Temporal Interaction Manifold

With an optimized tensor reduction kernel A, we compute the temporal interaction matrix X_i from the full interaction tensor Y_i for the ith activity sample. As has been shown, the temporal interaction matrices serve as compact, view-stable, and discriminative descriptors for group activities, and we would like to establish a probabilistic generative model for them to characterize the activity class distribution. However, note that X is symmetric with a unit norm and thus the space of all X's is not Euclidean. Therefore, to establish the probabilistic setting we need to first exploit the geometric property of the space and then build a probability distribution on it.

3.3.1 The Riemannian Property of the Discriminative Temporal Interaction Manifold

Although the set of all temporal interaction matrices is not an Euclidean space, with a properly defined Riemannian metric, it becomes a Riemannian manifold. For any two elements X_1' and X_2' in the tangent space $\mathscr{T}_X$ at X, the Riemannian metric is defined as

$$< X_1', X_2' > \triangleq tr(X_1'^T X_2'). \tag{3.11}$$

A related case was detailed recently in [149]. This manifold is what we mentioned above as DTIM, denoted as $\mathscr{X}$.

With the above defined Riemannian metric the basic geometry of DTIM is straightforward. The intrinsic distance between two temporal interaction matrices X_1 and X_2 is given by

$$d(X_1, X_2) = \arccos < X_1, X_2 >, \tag{3.12}$$

where

$$< X_1, X_2 > \triangleq tr(X_1^T X_2). \tag{3.13}$$

The geodesic , *i.e.*, the curve of minimum length connecting two temporal interaction matrices X_1 and X_2 , is given by

$$X(\lambda) = \frac{(1-\lambda)X_1 + \lambda X_2}{\lambda^2 + (1-\lambda)^2 + 2\lambda(1-\lambda) < X_1, X_2 >} \tag{3.14}$$

where λ is a real parameter between 0 and 1.

The exponential map and logarithmic map are important for manipulations on the Riemannian manifold. For DTIM defined above, the exponential map $\mathscr{E}_X : \mathscr{T}_X \to \mathscr{X}$ for $X' \in \mathscr{T}_X$ is defined as

$$\mathscr{E}_X(X') = \cos(< X', X' >^{\frac{1}{2}})X + \frac{\sin(< X', X' >^{\frac{1}{2}})}{< X', X' >^{\frac{1}{2}}} X'. \tag{3.15}$$

The logarithmic map $\mathscr{L}_X : \mathscr{X} \to \mathscr{T}_X$, which is actually the inverse map of exponential map, is then given by

$$\mathscr{L}_X(X_m) = \frac{\arccos(< X, X_m >)}{< X^*, X^* >^{\frac{1}{2}}} X^* \tag{3.16}$$

where

$$X^* = X_m - < X, X_m > X. \tag{3.17}$$

It is worth noting that the temporal interaction matrix and the corresponding DTIM investigated here are closely related to and rooted in the general theoretic framework of information geometry [150, 151]. We only present the necessary geometric properties to be employed in the subsequent section. For further study on information geometry the reader is referred to [150, 151].

3.3.2 A Basic Exponential Distribution on DTIM and its Parameter Estimation

It is expected that the temporal interaction matrices from different activity classes will reside distinctively on the DTIM. This motivates us to establish a probabilistic approach on DTIM to model the distribution of the temporal interaction matrices.

The 'Gausssian' distribution on a Riemannian manifold is initially addressed in [152] with the help of exponential/logrithmic mapping between the manifold and the tangent plane. Here, we take a direct approach to define a uni-modal exponential distribution for the temporal interaction matrix as

$$p(X;\mu,\sigma,z) = \frac{1}{z}\exp(-\frac{d^2(X,\mu)}{2\sigma^2}), \tag{3.18}$$

where μ is regarded as the 'center' of temporal interaction matrices, σ characterizes the scattering of the matrices on the manifold, and z is a normalization factor. Moreover, d is the intrinsic distance defined in (3.12). A temporal interaction matrix intrinsically close to the center will have a high probability value.

To estimate the parameters involved in the distribution p, a statistical approach based on observed samples is practically useful. Generally, we will have a set of weighted samples $\{(X_1,w_1),(X_2,w_2),\cdots,(X_N,w_N)\}$ observed from the distribution. We define the weighted Karcher mean,

$$\mu = \arg\min_{\psi}\sum_{i=1}^{N} w_i d^2(X_i,\psi), \tag{3.19}$$

to be the mean parameter μ. To numerically find μ, the iterations

$$\mu'^{(g+1)} = \frac{\sum_{i=1}^{N} w_i \mathscr{L}_{\mu^{(g)}}(X_i)}{\sum_{i=1}^{N} w_i} \tag{3.20}$$

and

$$\mu^{(g+1)} = \mathscr{E}_{\mu^{(g)}}(\mu'^{(g+1)}). \tag{3.21}$$

alternate and $\mu^{(g)}$ will converge to the weighted Karcher mean as g increases. Here $\mathscr{E}$ and $\mathscr{L}$ are the exponential and logrithmic maps given in (3.15) and (3.16) respectively.

Once the mean is determined, the scattering factor can be defined in a similar manner as

$$\sigma = (\frac{\sum_{i=1}^{N} w_i d^2(X_i, \mu)}{\sum_{i=1}^{N} w_i})^{1/2}. \tag{3.22}$$

The calculation for the normalization factor z is analytically infeasible and we need to take the Monte Carlo approach to find the integral

$$I = \int_{\mathscr{X}} \exp(-\frac{d^2(X, \mu)}{2\sigma^2}) dX \tag{3.23}$$

and consequently the estimate is $z = 1/I$. To perform the Monte Carlo integration we need to generate uniformly distributed samples on DTIM. To achieve this note that a $T \times T$ temporal interaction matrix is essentially equivalent to a $(1+T)T/2$ dimensional unit vector. Therefore we generate $(1 + T)T/2$ dimensional homogeneous Gaussian vectors and scale them into unit length. Then we transform the unit length vectors to temporal interaction matrices, which become uniformly distributed on DTIM.

3.4 Learning Multi-Modal Densities on DTIM

Suppose we have a training set $\{(X_1, c_1), (X_2, c_2), \cdots, (X_M, c_M)\}$ where $c_i \in \{1, 2, \cdots, C\}$ is the activity class label for the ith activity sample and there are totally C classes of group activities. Trivially we may learn a uni-modal distribution for each activity class using the method in the previous section. However, the actual scattering of temporal interaction tensors on DTIM may not be well approximated by a uni-modal model. This motivates the necessity to learn a multi-modal density for each activity class to achieve a better classification performance.

We aim to model the joint probability density function of the temporal interaction matrix X_i and the class label c_i, denoted as $P^J(X_i, c_i)$, defined as

$$P^J(X_i, c_i) = \sum_{j=1}^{J} b^j f^j(X_i, c_i) = \sum_{j=1}^{J} b^j \pi_{c_i}^j p_{c_i}^j(X_i) \tag{3.24}$$

where $p_{c_i}^j(X_i) = p(X_i; \mu_{c_i}^j, \sigma_{c_i}^j, z_{c_i}^j)$ is the uni-modal exponential component introduced in (3.18). Here we regard the joint probability as a linear mixture of J uni-modal likelihood functions where the jth component is $p_{c_i}^j(X_i)$. b^j is taken as the mixing coefficient for the

jth component, which for convenience is assumed to be independent of the class labels. $\pi_{c_i}^j$ is the class prior for class c_i in the jth component. For simplicity we take $\pi_{c_i}^j \equiv \frac{1}{C}$ regardless of j or c_i. Apparently, the 'mixture-of-p' distribution will behave as the 'mixture-of-Gaussian' in an Euclidean space, providing us the capability to approximate the irregular distribution on a Riemannian manifold analytically.

The determination of a proper number of components J is non-trivial, preventing us from directly applying an Expectation-Maximization (EM) procedure to learn all the components and mixing coefficients. Instead, we build up this linearly-combined multi-modal distribution in an incremental manner. In other words, we will 'boost' the distribution on DTIM. Suppose that we have in some way achieved a J-component density $P^J(X, c)$ and we want to update it into a $(J+1)$-component one by linearly mixing it with a new $f(X, c)$,

$$P^{J+1}(X, c) = (1 - \alpha)P^J(X, c) + \alpha f(X, c), \tag{3.25}$$

and we aim to maximize the *log* posteriori class probability for all samples in the training set $\sum_{i=1}^{M} \log P^{J+1}(c_i|X_i)$. Expanding the expression we get

$$\begin{aligned}
\sum_{i=1}^{M} \log P^{J+1}(c_i|X_i) &= \sum_{i=1}^{M} \log \frac{P^{J+1}(X_i, c_i)}{P^{J+1}(X_i)} \\
&= \sum_{i=1}^{M} \log \frac{(1-\alpha)P^J(X_i, c_i) + \alpha f(X_i, c_i)}{(1-\alpha)P^J(X_i) + \alpha f(X_i)} \\
&= \sum_{i=1}^{M} \log \frac{P^J(X_i, c_i) + \epsilon f(X_i, c_i)}{P^J(X_i) + \epsilon f(X_i)}
\end{aligned} \tag{3.26}$$

where $P^J(X_i) = \sum_{c=1}^{C} P^J(X_i, c)$, $f(X_i) = \sum_{c=1}^{C} f(X_i, c)$, and $\epsilon = \frac{\alpha}{1-\alpha}$.

A practically feasible optimization method to determine both $f(\cdot, \cdot)$ and ϵ, is expanding $\log P^{J+1}(\cdot|\cdot)$ into a Taylor's series around $P^J(\cdot, \cdot)$ with ϵf as the deviation (or increment) from $P^J(\cdot, \cdot)$, and ignoring the higher order terms as

$$\begin{aligned}
\sum_{i=1}^{M} \log P^{J+1}(c_i|X_i) &\doteq \sum_{i=1}^{M} \log P^J(c_i|X_i) \\
&\quad + \epsilon \sum_{i=1}^{M} \frac{\partial \log P^{J+1}(c_i|X_i)}{\partial P^J(X_i, c_i)} f(X_i, c_i) \\
&\doteq \sum_{i=1}^{M} \log P^J(c_i|X_i) + \epsilon \sum_{i=1}^{M} \frac{1 - P^J(c_i|X_i)}{P^J(X_i, c_i)} f(X_i, c_i) \\
&= \sum_{i=1}^{M} \log P^J(c_i|X_i) + \epsilon \sum_{i=1}^{M} h_i f(X_i, c_i).
\end{aligned} \tag{3.27}$$

Here the gradient of $\log P^{J+1}(\cdot|\cdot)$ w.r.t. $P^J(\cdot,\cdot)$ is a functional gradient rather than the usual one w.r.t. a variable. The approximate identity

$$\frac{\partial \log P^{J+1}(c_i|X_i)}{\partial P^J(X_i,c_i)} \doteq \frac{1-P^J(c_i|X_i)}{P^J(X_i,c_i)} \triangleq h_i \tag{3.28}$$

is derived in the Appendix. Note that samples with a small posterior probability will receive a larger weight, *i.e.*, the samples not well accounted for under the current model will be paid more attention through weight h. For this reason the h_i's can be regarded as 'discriminative weights'.

It is now clear that once $\sum_{i=1}^{M} h_i f(X_i, c_i)$ is maximized we can easily find the best ϵ (or α) such that the posteriori probability is maximized. Therefore, the key optimization is to maximize

$$\sum_{i=1}^{M} h_i f(X_i, c_i) = \sum_{i=1}^{M} h_i \pi_{c_i} p_{c_i}(X_i) \tag{3.29}$$

by determining the corresponding $\mu_{c_i}, \sigma_{c_i}, z_{c_i}$ (*i.e.*, $\mu_c, \sigma_c, z_c, c = 1, 2, \cdots, C$), taking discriminative weights h_i into account.

This 'discriminatively weighted parameter estimation' for each component falls well into the EM framework. The E-step for this iterative procedure is

$$w_i = (h_i f(X_i, c_i))/(\sum_{i=1}^{M} h_i f(X_i, c_i)) \tag{3.30}$$

and the M-step is essentially to maximize $\sum_{i=1}^{M} w_i \log f(X_i, c_i)$ w.r.t. μ_c, σ_c, z_c. The optimal μ_c here, is exactly the weighted Karcher mean with weights w_i introduced in Section 3.2. Therefore, to implement the M-step we need and only need to perform the parameter estimation presented in 3.2 . After f is determined in this way, a line search for the best ϵ is followed to achieve the maximum $\sum_{i=1}^{M} \log P^{J+1}(c_i|X_i)$. If no ϵ can improve the discriminativeness, the algorithm terminates and the final number of components is J.

The multi-modal density learning presented above follows the line of recent work on boosting non-discriminative density functions [153] and is also inspired by the discriminative boosting for sequence classification [154]. However, in this work we are investigating a multi-class multi-modal probabilistic model for classification on a nonlinear manifold rather than in the Euclidean space, and in particular, applying the method to group activity recognition.

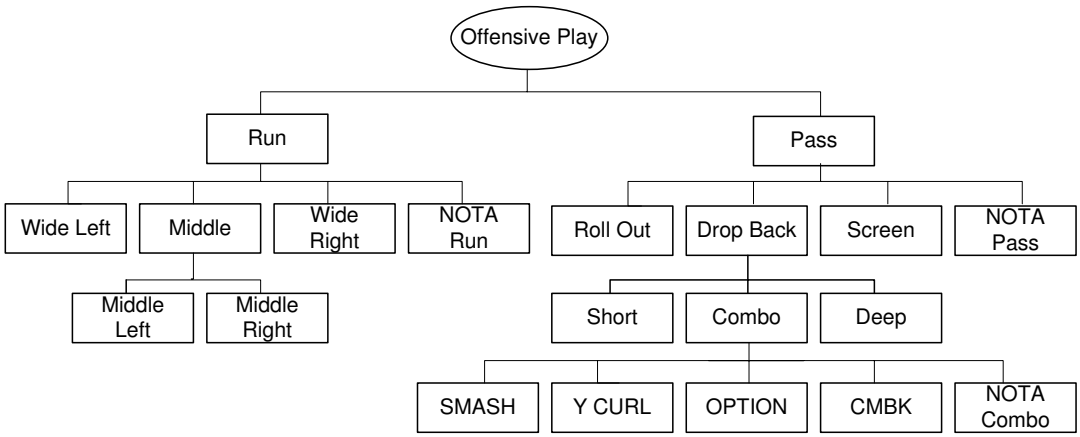


Figure 3.4: Play type hierarchy of GaTech Football Play Dataset.

3.5 Experiments

3.5.1 Experiments with Point Trajectories

The learning and recognition framework described above has been implemented on GaTech Football Play Dataset. The dataset is produced by Georgia Tech Athletic Association (who collect the videos) and the School of Interactive Computing, Georgia Tech (who process and annotate the videos). The GaTech Football Play Dataset consists of a collection of 155 NCAA football game videos. Each video is a segmented one which records an offensive play, *i.e.*, the play starts at the beginning of the video and terminates at the end of it. In our setting, the relevant activity type is defined as the offensive strategy in each play performed by the eleven offensive players, and the motion trajectories from the eleven defensive players are regarded as irrelevant ones. The offensive strategies in all videos have been annotated in a hierarchical manner, where each video is on the first level divided into either a 'RUN' play or a 'PASS' play, and on the subsequent levels divided into more specific play types. The play type hierarchy is shown in Figure 3.4.

Accompanied with each video is a complete ground-truth annotation of the object locations in each frame. Currently annotations for 56 videos are available for use. The annotation includes coordinates in the image plane of all the 22 players as well as field landmarks - the intersections of field lines. With the landmark information we can easily compute the plane homographies and convert motion coordinates into ground plane trajectories. We

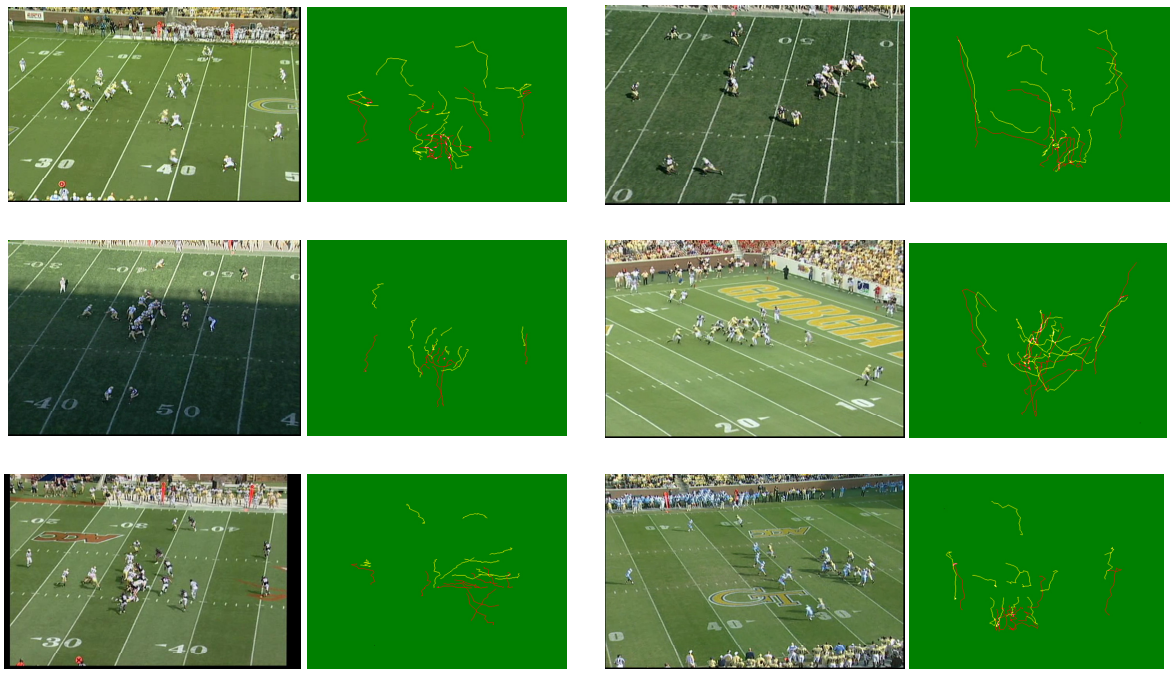


Figure 3.5: Samples of GaTech Football dataset: snapshots of plays with annotations.

show samples of snapshots together with corresponding ground truth trajectories (in the ground plane coordinates) in Figure 3.5, where red trajectories denote offensive players, yellow ones denote defensive ones, and the background is colored with green for better visualization.

A constant amount of time duration is used for all plays so as to maintain a constant size of temporal interaction matrix. In particular, for activity samples with varing lengths, we always normalize their time scales to T (We set $T = 10$), with trajectory interpolation and temporal resampling if necessary. To find the time span of the occurrence of an activity automatically, *i.e.*, group activity detection, is not considered in this work.

We first perform experiments on ground-truth data, where tracks for each player are manually labeled. The eleven trajectories for the eleven players on the offensive side are thus obtained for each play. Though view points vary among different plays, no geometric transformation is applied since view-stability will be enforced when learning the optimal tensor reduction kernel. However, we do put the origin at the center of the objects and normalize the distances between the objects and the center. From more than a hundred play

Table 3.1: The confusion matrix of play recognition using distances between objects: H,C,M,L, and R stand for HITCH Dropback, Combo Dropback, Middle Run, Wideleft Run, and Wideright Run respectively.

	C	H	M	L	R
C	93.1	0.2	5.6	0.4	0.7
H	0.1	71.0	6.5	15.2	7.2
M	1.6	1.3	77.2	12.4	7.5
L	0.4	0.6	0.6	95.4	3.0
R	0.8	0.1	0.1	2.5	96.5

samples we select five play types, including *Combo Dropback, HITCH Dropback, Middle Run, Wideleft Run* and *Wideright Run*, totaling a number of 56 play samples. Other play types with too few samples are not considered. To get a sufficient amount of training samples, we generate multiple new play samples from different views for each of the existing plays. To achieve this we apply view transformations to each of the 56 samples, with 12 typical views selected from the original dataset. The view transformations are simply locally affine ones whose parameters are determined by locating the landmark points of the football field. Learning and then classification runs a multiple of times independently, each of which uses a random division of sample collection into training (80%) and testing (20%) sets. Other free parameters (*e.g.*, α, β in Section 2) in the framework are determined by experimental evaluation.

The average confusion matrices are shown in Table 3.1 and 3.2, indicating the percentage by which a specific play type is recognized as itself/another. An average recognition rate of 87.9% and 85.5% is observed from the confusion matrices. The fully quantitative comparison with previous work, especially [6], is difficult due to a completely different framework as well as different datasets being used. Note that the previous work is based on Bayesian network modeling with explicit domain knowledge about the football game being incorporated. In contrast, the model in this chapter works in a data-driven manner and thus easily extendable to other general coordinated group activities.

To evaluate the effectiveness of optimal tensor reduction as well as probabilistic mod-

Table 3.2: The confusion matrix of play recognition using velocity correlations between objects: H,C,M,L, and R stand for HITCH Dropback, Combo Dropback, Middle Run, Wideleft Run, and Wideright Run respectively.

	C	H	M	L	R
C	88.4	2.6	5.1	1.2	2.7
H	0.4	68.5	8.7	15.2	7.2
M	0.5	0.8	77.8	13.4	7.5
L	3.5	2.7	0.5	92.8	0.5
R	0.3	2.1	0.4	3.9	93.3

eling on DTIM, a comparative study is carried out with a baseline descriptor and three baseline classifiers. The baseline descriptor to compare with is the 'nominal' temporal interaction matrix obtained from the trivial tensor reduction kernel - the identity matrix. The baseline classifiers are selected as two nearest neighbor (NN) classifiers and supporter vector machine (SVM) classifier. One of the two NN classifiers defines the distance between two temporal interaction matrices as the usual Euclidean distance (NN Euclidean). The other, instead, makes use of the intrinsic distance on DTIM (NN on DTIM). The SVM classifier is employed from libSVM [155] where the multi-class classifier is implemented as a set of one-to-one binary ones. In each of these SVMs a radial basis function kernel is used together with the default parameter settings of the software. Classification is performed by taking the majority of the votes from individual SVMs.

The overall correct recognition rates are shown in Table 3.3 and 3.4. In all cases the improvement brought by optimized tensor reduction is clear. On the other side, by comparing probabilistic modeling on DTIM with the other three baseline classifiers, we actually investigated its advantage over three typical philosophies besides a Bayesian network paradigm. The NN Euclidean classifier ignores the intrinsic geometry of DTIM but regards all temporal interaction matrices as elements in Euclidean spaces. The SVM takes into account the probable nonlinear phenomenon in the Euclidean space but bypasses it with the kernel trick to pursue linear seperability. NN on DTIM, meanwhile, exploits the essential geometry of the data space without a statistical point of view. The comparison among the

Table 3.3: Comparison of recognition performance using distances between objects(%).

	baseline	optimized
NN Euclidean	73.2	83.7
NN on DTIM	75.8	84.5
SVM	69.3	79.8
Probabilistic modeling on DTIM	76.3	87.9

Table 3.4: Comparison of recognition performance using velocity correlations between objects (%).

	baseline	optimized
NN Euclidean	74.7	81.7
NN on DTIM	78.1	85.2
SVM	67.5	77.4
Probabilistic modeling on DTIM	76.7	85.5

four demonstrates an empirical performance merit of the combination of both geometrical modeling and statistical modeling. Note that NN is only slightly weaker than the proposed framework due to the relative 'flatness' of DTIM. Geometrical and probabilistic modeling on more 'curved' manifold will potentially achieve more significant performance gain.

In the second round of experiment, we select a state-of-art multi-object tracker [9] to provide trajectories for the testing activity samples, due to its good performance on tracking soccer players. By this experiment we aim to investigate the robustness of our model toward non-ideal input data in a practical scenario. The tracking results are shown as well in Figure 3.6 for the same plays shown in Figure 3.5, with snapshots with bounding boxes and the tracks in the ground plane.

The experiment setting follows exactly as used in the first round of experiment, except for the testing phase where no ground-truth is used but we apply tracking. The eleven offensive trajectories are selected and identified from all trajectories by comparing

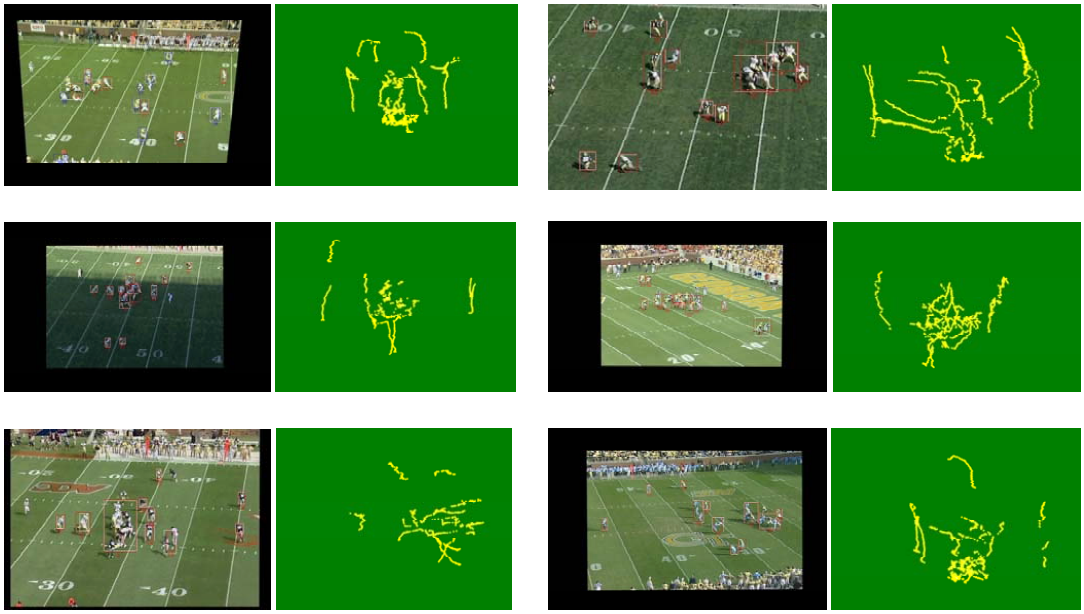


Figure 3.6: Tracks provided by multi-object tracking using [9]: snapshots and computed tracks in the ground plane coordinates.

their initial location coordinates with the annotation, and the complete trajectories are generated according to the method introduced in previous section for handling non-robust tracking. The confusion numbers and accuracy comparison with baselines are shown in Table 3.5, 3.6, 3.7, and 3.8 respectively. Robustness toward noisy tracking is demonstrated in the gentle performance degradation of geometric modeling in terms of nearest neighbor on DTIM and multi-modal density learning on DTIM, in contrast to a significant accuracy drop for Euclidean nearest neighbor as well as SVM.

3.5.2 Experiments with Group Articulated Human Actions

We then demonstrate the performance of our method on group activities using image features. The dataset we employ is the collective activity dataset previously used in [4, 7]. This dataset contains 74 sequences, each of which is in the length of several hundred frames. Every tenth frame in each sequence is annotated with image locations of the humans, the size of the bounding boxes, the pose direction, and the activity type. Seven activity types are provided including crossing, waiting, queuing, walking, talking, dancing, and jogging. We

Table 3.5: The confusion matrix of play recognition using distances between objects computed trajectories: H,C,M,L, and R stand for HITCH Dropback, Combo Dropback, Middle Run, Wideleft Run, and Wideright Run respectively.

	C	H	M	L	R
C	89.5	1.3	6.2	1.3	1.7
H	1.1	67.4	8.2	15.3	8.0
M	2.5	2.0	74.0	13.2	8.3
L	1.7	1.9	1.9	90.2	4.3
R	2.0	1.2	1.3	3.6	91.9

Table 3.6: The confusion matrix of play recognition using velocity correlations between objects on computed trajectories: H,C,M,L, and R stand for HITCH Dropback, Combo Dropback, Middle Run, Wideleft Run, and Wideright Run respectively.

	C	H	M	L	R
C	75.8	4.6	9.2	4.9	5.5
H	4.6	55.4	14.6	18.4	7.0
M	2.9	7.5	59.0	18.0	12.6
L	7.2	4.7	1.5	82.9	3.7
R	5.5	6.9	1.7	7.9	78.0

Table 3.7: Comparison of recognition performance using distances between objects on computed trajectories (%).

	baseline	optimized
NN Euclidean	61.6	75.4
NN on DTIM	74.1	81.8
SVM	59.0	63.9
Probabilistic modeling on DTIM	75.7	83.0

Table 3.8: Comparison of recognition performance using velocity correlations between objects on computed trajectories (%).

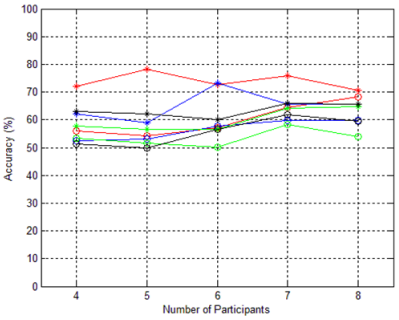
	baseline	optimized
NN Euclidean	63.4	69.1
NN on DTIM	62.4	72.2
SVM	55.4	65.9
Probabilistic modeling on DTIM	63.6	72.0

additionally label the group activity in each tenth frame by taking the dominant individual action type, and then look for a segment of consecutive ten annotations with a consistent group activity label. We keep those segments involving four to eight participants, as fewer people do not constitute a meaningful 'group' activity and segments with more than nine participants are rare. Moreover, as suggested in the dataset 'walking' is dropped out of consideration, and we also ignore 'dancing' with few segments available. Eventually, the dataset used in this part includes five activity categories, each of which consists of segments of equal length of ten sampled on the multiples of the tenth frames in the original sequence, and with the number of involved objects varying from four to eight.

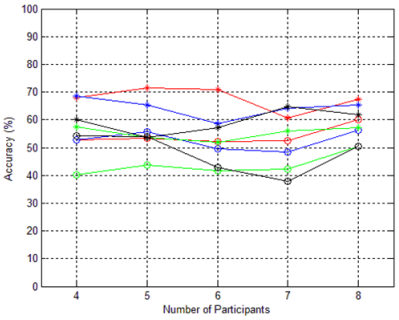
We learn a separate DTIM model for each number of participating objects. The order of the objects is simply taken from the leftmost of the image to the rightmost in the first frame. We test on six combinations of two local features and three distance measures. The local features include Histogram of Optical Flow (HOF) densely estimated and Histogram of Oriented Gradient (HOG) on Harris corners in the bounding boxes. The three distances between histograms are χ^2 distance, cosine distance, and Euclidean distance. All other experimental settings follow the same as in previous subsection, in the form of multiple runs of five fold cross validations on randomized division of all samples into training/testing.

We show the overall recognition accuracy in Figure 3.7, where the same set of baseline methods are also used for comparison. Best performance is observed using our proposed method for most cases, and manifold modeling or optimized tensor reduction generally gives improved accuracies over Euclidean modeling or non-optimized tensor reduction.

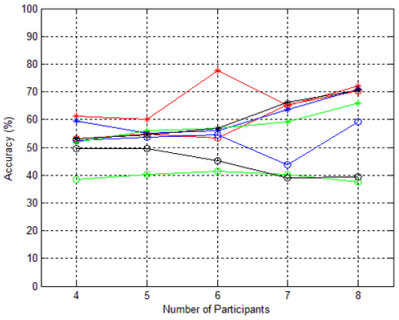
We also briefly discuss our proposed method qualitatively as compared to [4, 7] who also reported work on the same dataset. A fair quantitative comparison is difficult due to different assumptions made in respective methods, unavailability of implementations, as well as data usage protocols. In contrast to [4] which proposes a spatio-temporal correlation histogram descriptor, our method marginalizes the spatial and inter-personal correlations but mainly focuses on the temporal evolution of the overall interactions, and the resulting descriptor - temporal interaction matrix - is more compact. More importantly, we manipulate the descriptors geometrically by exploiting the intrinsic topology of the space of them, while no such effort is made in [4]. A direct read of the statistics reported in [4] shows that our recognition performance is better than what is presented in [4]. On the other hand, the probabilistic graphical model explicitly incorporating features, individual actions,



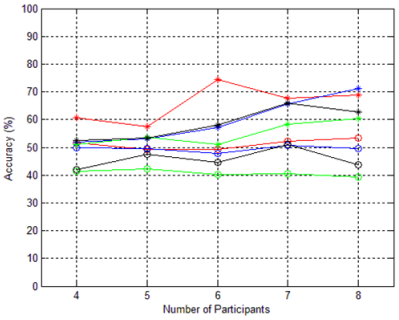
(a) HOF and Chi-Square distance



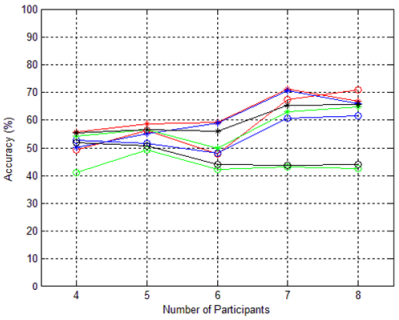
(d) HOG and Chi-Square distance



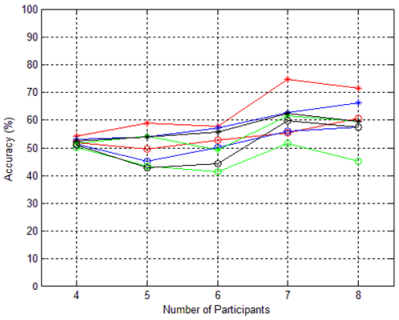
(b) HOF and Cosine distance



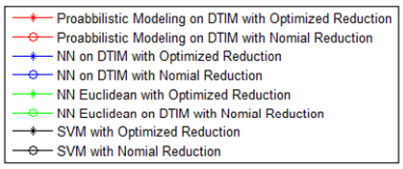
(e) HOG and Cosine distance



(c) HOF and Euclidean distance



(f) HOG and Euclidean distance



Legend

Figure 3.7: Recognition accuracy on collective activity dataset.

global behavior, and their relationships is presented in [7], while our method 'condenses' these information into a single measurement. The accuracy numbers achieved by us are gently lower than those in [7] while a significant drop in complexity is the main merit of our method.

3.6 Summary

In this chapter we investigated the modeling and recognition of coordinated multi-object activity in a data-driven manner. In particular, we proposed a temporal interaction matrix to characterize a group activity view-stably and discriminatively. We established the Riemannian geometry for the space of temporal interaction matrices, DTIM, and set up the 'intrinsic' probabilistic mechanism for random samples on DTIM. To better approximate the possibly complex distribution on DTIM, we further recursively built multi-component densities on DTIM in a way that inter-class separability is enhanced. We demonstrated the effectiveness of the proposed framework using football plays and articulated human actions as experimental data. We made little use of activity-specific domain knowledge and hope that the framework is more generally extensible to a larger body of categories of group activities.

3.7 Appendix: Derivation of (3.28)

By elementary calculus it is obvious that

$$\frac{\partial \log P^{J+1}(c_i|X_i)}{\partial P^J(X_i,c_i)} = \frac{\sum_{c=1,\cdots,C,c\neq c_i} P^J(X_i,c) + \epsilon \sum_{c=1,\cdots,C,c\neq c_i} f(X_i,c)}{(P^J(X_i,c_i)+\epsilon f(X_i,c_i))(P^J(X_i)+\epsilon f(X_i))}. \tag{3.31}$$

Since ϵf is a local deviation from P^J in Taylor's expansion, here we may assume $\epsilon f \ll P^J$. Hence we ignore the terms of ϵf and have the approximation

$$\begin{aligned} \frac{\partial \log P^{J+1}(c_i|X_i)}{\partial P^J(X_i,c_i)} &\doteq \frac{\sum_{c=1,\cdots,C,c\neq c_i} P^J(X_i,c)}{P^J(X_i,c_i)P^J(X_i)} \\ &= \frac{\sum_{c=1,\cdots,C,c\neq c_i} P^J(c|X_i)}{P^J(X_i,c_i)} = \frac{1-P^J(c_i|X_i)}{P^J(X_i,c_i)}. \end{aligned} \tag{3.32}$$

Note that the best ϵ is determined after f is learned.

Chapter 4

Group Motion Segmentation Using a Spatio-Temporal Driving Force Model

4.1 The Group Motion Segmentation Problem

In this chapter we turn our attention to the problem of *group motion segmentation*, and propose a solution for it. The group motion segmentation problem arises in a large body of video surveillance applications and sports video analysis. Specifically, we have in hand point trajectories from consecutive frames of a video sequence, and aim to group them into two or more clusters. While this may appear to be similar to the traditional motion segmentation problems [156, 157, 158, 159, 11, 160, 161, 162, 163], it is actually different. As has been discussed, during group motion the participating objects/people have distinctive and varying motions but the group itself collectively demonstrates an underlying activity of a particular pattern, while the non-participating group of objects/people does not demonstrate that pattern. Recent developments on analyzing these motion patterns of the participating group [1, 2, 109, 6, 110, 111, 5, 106, 107, 3, 113], to recognize the group motion pattern or detect a change or an anomaly assume that all objects are involved in the activity, which is far from realistic scenarios where only a portion of the objects/people contribute to the specific group motion. The group motion segmentation problem explored here attempts to divide the point trajectories into participating group and non-participating group, or into multiple participating groups, each corresponding to a coherent group motion pattern.

In a football play, the offensive players are the participants in the offensive motion, while the defensive players are non-participants. Different offensive participants will give rise to different moving trajectories, while the group will collaboratively demonstrate an ensemble motion pattern which can be identified as a semantic strategy represented as a play diagram in the playbook. This group motion pattern manifests itself as the spatial constraint and temporal co-occurrence among the interacting trajectories. Note that participants move under the guidance of a play diagram but significant spatio-temporal variations exist among different realizations. Also, as the participating and non-participating group are mixed within the same area, they cannot be separated by simply partitioning the spatial domain. For these reasons, we address the group motion segmentation problem in the context of

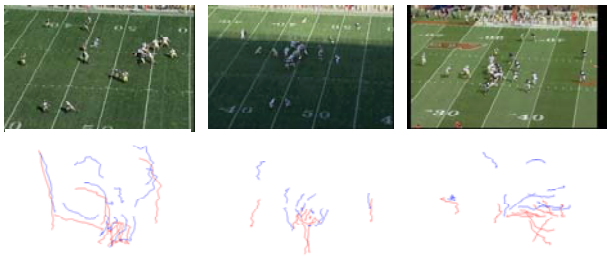


Figure 4.1: Examples of relevant motion mixed in irrelevant motion, sampled from GaTech Football Play Dataset. The top row gives snapshots of videos of the plays and bottom row contains corresponding trajectories. The red trajectories are offensive ones (participating ones) and the blue defensive (non-participating) ones.

offensive player identification in football plays. Examples of participating group motion mixed with unrelated or irrelevant motion under consideration in this chapter are given in Figure 4.1. Note that our segmentation is based on motion only: we do not make use of appearance based features, which may not be always available due to poor video quality or other reasons.

There are additional challenges beyond the aforementioned ones. Though the participating group of a football play consists of a constant number of objects, more generally the group motion pattern may be executed by a varying number of agents in different realizations, and the number may change during the activity due to a participant's departure or the arrival of a new member. Moreover, as the motion trajectories are generated by a tracking algorithm, they are noisy. The trajectories may be incomplete, fragmented or missing due to limitations of the tracking algorithm, strong occlusion among objects, and other issues such as video quality. Each of these challenges should be addressed by a vision algorithm and indeed our method is able to handle them.

Looking at the traditional motion segmentation problems we find the majority of them addressing trajectories of feature points from independent 3-D rigid objects [156, 157, 158, 159, 11, 160] and the problem eventually boils down to subspace clustering. Other works also exploit dependant/articulated rigid body motion [162] or a motion model leading to nonlinear manifold clustering [163]. The group motion segmentation problem considered here has little in common with them. On the other hand, the non-rigid Structure-from-

Motion problems [164, 165, 166] assume non-rigid shape to be linear combination of rigid ones, and non-rigid motion segmentation [161] makes use of local piecewise subspace model, while the group motion under our consideration does not belong to either of these cases. In this work we employ Lie group theory [167] and in particular establish a statistical model over Lie algebra. Lie group and Lie algebra based approaches play roles in invariant visual modeling and recognition [168, 169], robotics [170], 3-D rigid motion estimation [171, 172, 173], as well as dense flow field modeling [174]. In this work, we discuss a new application to group motion estimation.

Looking beyond group motions arising from multiple agents, it is in fact not difficult to find analogies between the scenario involving multiple agents and cases where an ensemble of trajectories is obtained from tracking local space-time interest points. Specifically, the model proposed in this chapter can also be used toward single human articulated motion, or other video dynamics of relevance, simply based on space-time interest points tracked throughout a sequence. In realistic natural scenes the human actions are captured under complex, cluttered and dynamic background, all of which contribute to space-time interest points, while those points from the human body constitute only a portion of them. Consequently, methods using the Bag-of-Words or Bag-of-Features framework tend to degrade when directly applying space-time interest points detector regardless of the irrelevant points from the background. Therefore, it is necessary that the points and associated trajectories from the relevant motion are identified before we apply any machine learning methods which assume clean background. This challenge falls right into the framework considered here and we will apply the model to this case as an extension.

The proposed model is detailed in Section 4.2, and its application to group motion segmentation is presented in Sections 4.3 and 4.4. Section 4.5 empirically demonstrates the application of the approach, followed by extension to local space-time interest points in Section 4.6. Finally discussions are presented in 4.7.

4.2 Spatio-Temporal Driving Force Model for A Group Motion Pattern

In this section we introduce a characterization for a group motion pattern, made of a collection of spatio-temporal constrained trajectories possibly noisy, of varying number, and undergoing spatio-temporal variation from realization to realization. The key idea is that we model the group motion as a dynamic process driven by a *spatio-temporal driving force*

densely distributed across the area where the motion occurs, instead of simply as a set of discrete point trajectories. To be precise, the driving force is denoted as a 3×3 real matrix $F(t_0, t_f, x, y)$ which moves an object located at $X(t_0) = (x(t_0), y(t_0), 1)^T$ in homogeneous coordinates at time t_0 to location $X(t_f) = (x(t_f), y(t_f), 1)^T$ at time t_f, by

$$X(t_f) = F(t_0, t_f, x, y)X(t_0). \tag{4.1}$$

Without loss of generality, we usually take $t_0 = 0$ to be the starting time. It is obvious that once we have learned F for all t_f, x, and y, then the group motion is completely characterized. To be able to learn F, we limit our attention to those parametric F's which have the following properties: 1) $F(t_1, t_2, x, y)F(t_2, t_3, x, y) = F(t_1, t_3, x, y)$; 2) $F(t_1, t_2, x, y)^{-1} = F(t_2, t_1, x, y)$; and 3)

$$F(t, t+1, x, y) \triangleq F(t, x, y) = \begin{bmatrix} F_{11}(t,x,y) & F_{12}(t,x,y) & F_{13}(t,x,y) \\ F_{21}(t,x,y) & F_{22}(t,x,y) & F_{23}(t,x,y) \\ 0 & 0 & 1 \end{bmatrix}. \tag{4.2}$$

The $F(t, x, y)$'s defined this way is in fact a Lie group and more specifically an affine group [167]. By making use of Lie group theory we may achieve both generality and flexibility in modeling complex motion patterns, as shown next.

$F(t, x, y)$ characterizes the motion potential at time t for an object located at (x, y). However, we may look into an alternative representation. Consider $F(t, t+\delta t, x, y)$ and $X(t+\delta t) = F(t, t+\delta t, x, y)X(t)$, and we then have $X(t+\delta t) - X(t) = (F(t, t+\delta t, x, y) - I)X(t)$ where I is the identity matrix. Dividing both sides by δt and letting $\delta t \to 0$, we get $X'(t) = \mathbf{f}(t, x, y)X(t)$, in which $X'(t) = (x'(t), y'(t), 0)^T$ is the speed of the object, and

$$\mathbf{f}(t,x,y) = \begin{bmatrix} f_{11}(t,x,y) & f_{12}(t,x,y) & f_{13}(t,x,y) \\ f_{21}(t,x,y) & f_{22}(t,x,y) & f_{23}(t,x,y) \\ 0 & 0 & 0 \end{bmatrix} \triangleq \lim_{\delta t \to 0} \begin{bmatrix} \frac{F_{11}(t,t+\delta t,x,y)-1}{\delta t} & \frac{F_{12}(t,t+\delta t,x,y)}{\delta t} & \frac{F_{13}(t,t+\delta t,x,y)}{\delta t} \\ \frac{F_{21}(t,t+\delta t,x,y)}{\delta t} & \frac{F_{22}(t,t+\delta t,x,y)-1}{\delta t} & \frac{F_{23}(t,t+\delta t,x,y)}{\delta t} \\ 0 & 0 & 0 \end{bmatrix}. \tag{4.3}$$

In fact, $\mathbf{f}(t, x, y)$ is the Lie algebraic representation of $F(t, x, y)$ and the two are related by the exponential map $F(t, x, y) = \exp(\mathbf{f}(t, x, y)) = \sum_{i=0}^{\infty} \frac{1}{i!}\mathbf{f}(t, x, y)^i$ and logarithmic map

$\mathbf{f}(t,x,y) = \log(F(t,x,y)) = \sum_{i=1}^{\infty} \frac{(-1)^{i-1}}{i}(F(t,x,y) - I)^i$ [167]. In other words, we may work equivalently with $\mathbf{f}$ instead of directly working with F for the driving force model. The advantage of using $\mathbf{f}$ lies in the fact that the space of all $\mathbf{f}$'s, the Lie algebra, is a linear one on which we may develop various statistical tools, while the Lie group of F's is a nonlinear manifold not easy to work with. More importantly, $X'(t) = \mathbf{f}(t,x,y)X(t)$ implies that the location and speed of the object are linearly related by $\mathbf{f}$, and both the location and the speed are low-level features obtainable from the motion trajectories of the objects, *i.e.*, learning a single driving force model $\mathbf{f}(t,x,y)$ will be straightforward.

4.2.1 Learning a Spatial Hybrid Driving Force Model at a Time Instant

Suppose we fix a specific time instant t. Intuitively, different driving forces $\mathbf{f}(t,x,y)$'s induce different 'affine' motions for different (x,y)'s, and learning for all (x,y)'s in the whole area of group motion is intractable. On the other hand, constant $\mathbf{f}(t,x,y) \equiv \mathbf{f}(t)$ induces a global 'affine' motion to all objects in the group, whose representative power is severely limited. For this reason, we propose a *spatial hybrid* model, in which we assume K driving forces in the area of group motion. The effective area of the kth ($k = 1, 2, \cdots, K$) force is Gaussian distributed as $(x,y)^T \sim \mathcal{N}(\mu^k, \mathbf{\Sigma}^k)$. (We drop the argument t in this subsection for simplicity.) In the effective area, there is a uniform 'affine' motion $\mathbf{f}^k$. For notational convenience we write

$$\mathbf{f}^k = \begin{bmatrix} A^k & \mathbf{b}^k \\ \mathbf{0}^T & 0 \end{bmatrix} \tag{4.4}$$

where A^k is the upper-left 2×2 block of $\mathbf{f}^k$.

Consider the feature vector $Y \triangleq (x, y, x', y')^T$ extracted from an object motion trajectory driven by the kth force, and it is obvious that

$$Y = \begin{bmatrix} I \\ A^k \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{b}^k \end{bmatrix}. \tag{4.5}$$

Taking into account the noise created due to approximating the speed (1st order derivative) of point trajectories, we represent the observed feature vector as

$$\mathbf{y} = Y + \begin{bmatrix} \mathbf{0} \\ \mathbf{n}^k \end{bmatrix} \tag{4.6}$$

where $\mathbf{n} \sim \mathcal{N}(\mathbf{0}, T^k)$. Then after some manipulations we find $\mathbf{y} \sim \mathcal{N}(\nu^k, \mathbf{\Gamma}^k)$, where

$$\nu^k = \begin{bmatrix} \mu^k \\ A^k \mu^k + \mathbf{b}^k \end{bmatrix} \tag{4.7}$$

and

$$\mathbf{\Gamma}^k = \begin{bmatrix} \mathbf{\Sigma}^k & \mathbf{\Sigma}^k A^{kT} \\ A^k \mathbf{\Sigma}^k & A^k \mathbf{\Sigma}^k A^{kT} + T^k \end{bmatrix}. \tag{4.8}$$

We are now in a position to learn the spatial hybrid driving force model for a particular time instant t, from a limited discrete number of objects in a group motion. Suppose we have observed t_M motion feature vectors $\{\mathbf{y}_m\}_{m=1}^{t_M}$, then the learning task boils down to fitting a Gaussian Mixture Model (GMM) of K component $\{(\nu^k, \mathbf{\Gamma}^k)\}_{k=1}^K$ (using uniform mixing coefficient $\frac{1}{K}$). An Expectation-Maximization procedure is employed to complete the inference. After successfully learning the GMM, the spatial hybrid driving model, *i.e.*, $\{(\mathbf{f}^k, \mu^k, \mathbf{\Sigma}^k)\}_{k=1}^K$, is recovered. Note that by Gaussian assumption the learned effective areas of different driving force components will overlap and cover the whole area; to eliminate this ambiguity we technically partition the area such that the distance between the component's center and (x, y) is minimized.

It may be useful to recap what has been achieved till now by using the spatial hybrid driving force model. In fact, at time t we partition the area into K subareas, and model the instant motions of all the objects in one subarea to be an uniform 'affine' motion. In this way we are actually establishing a dense 'motion potential field' across the area where the group motion happens. Though the field may be learned from the motion features of sparse objects, it exists everywhere, and any other group of objects giving rise to the same field model is regarded as the same group motion pattern at time t, though participating objects (which the motion features come from) may appear in different locations from one group pattern to another. Figure 4.2 gives two examples of the spatial hybrid driving force model, where the sparse objects for learning the model, the learned partition of the area, as well as the learned driving force in each partition are all shown for two group motion samples in the dataset.

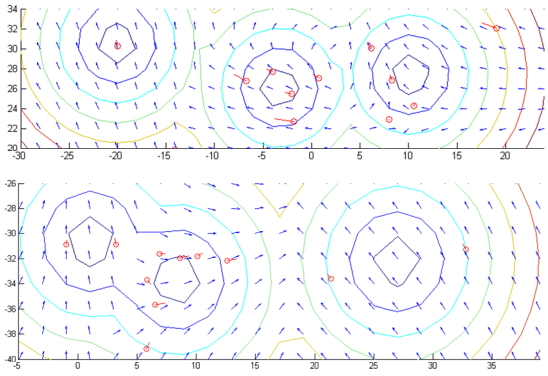


Figure 4.2: Two samples of 3-component driving force model at a time instant. The red circles denote the relative objects (offensive players) and the red bars attached to them denote the velocities of the objects. The blue arrow array gives the densely distributed driving force learned from the sparse object motion observations. The contour lines enclose the effective areas for the driving forces.

4.2.2 Learning Temporal Evolution of Driving Force Models for a Group Motion Pattern

Having obtained a hybrid driving force model at a particular time instant, we turn to the temporal evolution of the model, which eventually characterizes the complete group motion. Denote the driving force model at time t as $\mathfrak{M}(t) = \{\mathbf{m}^k(t)\}_{k=1}^K$ where $\mathbf{m}^k(t) = (\mathbf{f}^k(t), \mu^k(t), \mathbf{\Sigma}^k(t))$, and assume we have learned models at time $t = t_1, t_2, \cdots, t_T$ (which may not be continuous, but intermittent instants due to fragmented trajectories). We then learn a temporal sequence of these models in two steps: 1) component alignment between two consecutive instants, and 2) learning a parametric representation for the temporal sequence.

Component alignment is performed on $\mathfrak{M}(t_{i+1})$ with respect to aligned $\mathfrak{M}(t_i)$, starting from $\mathfrak{M}(t_2)$ with respect to $\mathfrak{M}(t_1)$. Mathematically, let the vector $(k_1, k_2, \cdots, k_K)^T$ denote the element-wise permutation of vector $(1, 2, \cdots, K)^T$, then we aim to find an optimal permutation such that $\sum_{j=1}^K D(\mathbf{m}^j(t_i), \mathbf{m}^{k_j}(t_{i+1}))$ is minimized, where $D(\mathbf{m}, \mathbf{m}')$ is a properly defined dissimilarity between model $\mathbf{m}$ and $\mathbf{m}'$. In other words, we are trying to associate each driving force component at time t_{i+1} uniquely with one at time t_i such that within each associated pair are they as similar as possible. We then give each component

at t_{i+1} a new component index which is nothing but that of the associated component at t_i. Obviously, after component alignment for a fixed k, $\mathbf{m}^k(t)$ become similar, or change smoothly, among all t_i's, which reduces the complexity of the parametric representation.

Note that the driving force model $\mathbf{m}$ includes the 'force' term $\mathbf{f}$ and the effective area $\mathcal{N}(\mu, \mathbf{\Sigma})$, and $\mathbf{f}$ lies on the Lie algebra which is a linear space. Therefore, we use

$$\begin{aligned} &D(\mathbf{m}, \mathbf{m}') = \|\mathbf{f} - \mathbf{f}'\| + \\ &\alpha(KL(\mathcal{N}(\mu, \mathbf{\Sigma}) \| \mathcal{N}(\mu', \mathbf{\Sigma}')) + KL(\mathcal{N}(\mu', \mathbf{\Sigma}') \| \mathcal{N}(\mu, \mathbf{\Sigma}))) \end{aligned} \tag{4.9}$$

where $KL(\cdot\|\cdot)$ denotes the Kullback-Leibler divergence. Then the optimal permutation can be solved using the classical Hungarian assignment algorithm [148].

Now we are looking for a parametric representation of the temporal sequence $\mathfrak{M}(t)$ for $t = 1, 2, \cdots$. Note that $\mathbf{m}^k(t)$ is essentially composed of $\mathbf{f}^k(t)$, the space of which is a Lie algebra (denoted as $\mathfrak{F}$), and $\mathcal{N}(\mu^k(t), \mathbf{\Sigma}^k(t))$, which is on the nonlinear manifold of 2×2 Gaussian's. As the nonlinearity brings analytical difficulty, we work with the parameters of $\mathcal{N}(\mu^k(t), \mathbf{\Sigma}^k(t))$, *i.e.*, $[\mu_1^k(t), \mu_2^k(t), \sigma_{11}^k(t), \sigma_{12}^k(t)(= \sigma_{21}^k(t)), \sigma_{22}^k(t)]^T \triangleq \mathbf{g}^k(t)$, rather than the Gaussian distribution itself, and regard the space of $\mathbf{g}^k(t)$'s (denoted as $\mathfrak{G}$) to be linear as well. (Though linearity does not rigorously hold, it is an effective approximation.)

We hence establish a parametric model for the temporal sequence $\mathfrak{M}(t), t = 1, 2, \cdots$ on the Cartesian product space $\mathfrak{F} \times \mathfrak{G}$. We propose the linear model $\{(\mathbf{W}^k t + \mathbf{w}^k + \mathbf{v}_1, \mathbf{U}^k t + \mathbf{u}^k + \mathbf{v}_2)\}_{k=1}^K$ for $\{(\mathbf{f}^k(t), \mathbf{g}^k(t))\}_{k=1}^K$, where

$$\mathbf{W}^k = \begin{bmatrix} W_{11}^k & W_{12}^k & W_{13}^k \\ W_{21}^k & W_{22}^k & W_{23}^k \\ 0 & 0 & 0 \end{bmatrix}, \tag{4.10}$$

$\mathbf{U}^k = [U_1^k, U_2^k, U_3^k, U_4^k, U_5^k]^T$, and $\mathbf{v}_1, \mathbf{v}_2$ are independent white Gaussian perturbation. With this model, $\mathbf{f}^k(t)$'s (resp. $\mathbf{g}^k(t)$'s) for each component k, when t varies, will approximately move in the one-dimensional subspace of $\mathfrak{F}$ (resp. $\mathfrak{G}$). In other words, components of the time-varying spatial hybrid driving force will evolve along straight lines in $\mathfrak{F} \times \mathfrak{G}$. A visual illustration for this idea is shown in Figure 4.3.

We use linear representations for the temporal sequence of driving forces basically for simplicity and effectiveness (to be demonstrated in the experiment). We may attempt advanced techniques but in this initial work on group motion segmentation we use linear ones to begin with. The straight lines $\{(\mathbf{W}^k t + \mathbf{w}^k, \mathbf{U}^k t + \mathbf{u}^k)\}_{k=1}^K$ in $\mathfrak{F} \times \mathfrak{G}$ are simply

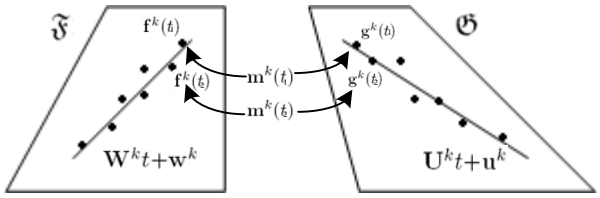


Figure 4.3: A pictorial illustration of the temporal evolution of driving force models. It only shows the kth component. In all, there should be K ones, *i.e.*, on left and right planes there should be K straight lines respectively.

fitted using previously obtained models $(\mathbf{f}^k(t), \mathbf{g}^k(t))$'s at time $t = t_1, t_2, \cdots, t_T$ in the least square sense. However, once these lines are available, we may re-sample the lines at other t's to generate new spatial hybrid driving forces at those time instants. In this way, we actually learn the group motion pattern within the whole time duration from only motion information at limited time instants.

Up to now, a group motion pattern has been fully captured by $\{(\mathbf{W}^k t + \mathbf{w}^k, \mathbf{U}^k t + \mathbf{u}^k)\}_{k=1}^K \triangleq GM$, by which the motions of participants of the group activity, at any location and any time, are condensed into and may be recovered from the corresponding GM.

4.3 DP-DFM: Accounting for Group Motion Variation

The variation of group motion patterns from video to video leads to the variation of GM's learned from video to video. To statistically model this variability among different GM's, we establish a Dirichlet Process (DP) [175] over $\mathfrak{F} \times \mathfrak{G}$, leading to a Dirichlet Process - Driving Force Model (DP-DFM). The DP-DFM is essentially a Bayesian mixture model good for handling an unknown number of mixing components. As we do not have prior knowledge about the variability of the group motion patterns (*i.e.*, GM's from different offensive plays), DP-DFM is a natural choice.

Specifically, we regard the GM as a long vector consisting of the elements of $\mathbf{W}^k$, $\mathbf{w}^k$, $\mathbf{U}^k$, $\mathbf{u}^k$, $k = 1, \cdots, K$, and suppose it comes in the following manner (called 'Stick Breaking'): 1) Let $v_t \sim Beta(1, \eta)$ and $\lambda_t = v_t \Pi_{l=1}^{t-1}(1 - v_l)$; 2) Draw a sample from $\Sigma_{t=1}^{\infty} \lambda_t \delta(\theta_t)$, where $\delta(\theta_t)$ is a point measure situated at parameter vector θ_t, and $\theta_t \sim \mathcal{G}_0$, which is a base measure (Gaussian-Wishart in this work); 3) Draw a GM from a Gaussian whose mean and

covariance are specified by θ_t. In this way the DP-DFM formulation has become a canonical DP mixture problem and we employ the standard procedure [175] to complete the inference of DP-DFM.

4.4 Probabilistic Segmentation

With a DP-DFM learned from training group motion patterns, we segment a new testing motion pattern by synthesizing a set of Monte Carlo samples (*i.e.*, spatio-temporal DFM's) from DP-DFM, matching the trajectories in the testing motion pattern with these simulated models, and voting for the best matching motion trajectories as the segmentation result. To simulate a Monte Carlo sample, we first draw a GM from the DP-DFM. Then we recover the temporal sequence of spatially hybrid driving forces $\mathfrak{M}(t) = \{\mathbf{m}^k(t)\}_{k=1}^K$, and consequently the time-varying densely distributed driving forces $F(t, x, y)$. As a result, at each time instant t and for each object(trajectory) at t in the testing motion pattern, we may predict its location at $t+1$ by (1), and measure the discrepancy between the predicted and actual locations (The discrepancy is simply measured as the distance between the two in this work). Those objects(trajectories) which accumulatively have the least discrepancies across all t's with all simulated driving force samples are finally determined as the participating objects.

4.5 Experiments

We perform group motion segmentation on the football play dataset used in previous chapter again, dividing players into participating (offensive) ones and non-participating (defensive) ones solely by their motion trajectories. We designed three rounds of experiments, the first of which employs ground-truth trajectories from training to testing. As in any practical system the input trajectories will be noisy, in the second round we generate noisy trajectories from the ground-truth and experiment with them. In the final round we test the learned framework on tracks computed from videos with a state-of-art multi-object tracker [9]. In each round of experiments, we carry out multiple passes of five-fold evaluations, *i.e.*, in each pass we randomly divide 4/5 of the samples into the training set and the remaining samples into testing set. The final statistics is aggregated from the average of all passes. Empirically we find $K = 5$ is a good selection for the total number of components. For

Gaussian-Wishart prior $\mathcal{G}_0$, we set the Wishart scale matrix to be the inverse of sample covariance of training GM's, and the Gaussian mean to be the sample mean of training GM's. The other free parameters in the framework are determined by experimental evaluation.

4.5.1 Experiment on Ground-Truth Trajectories

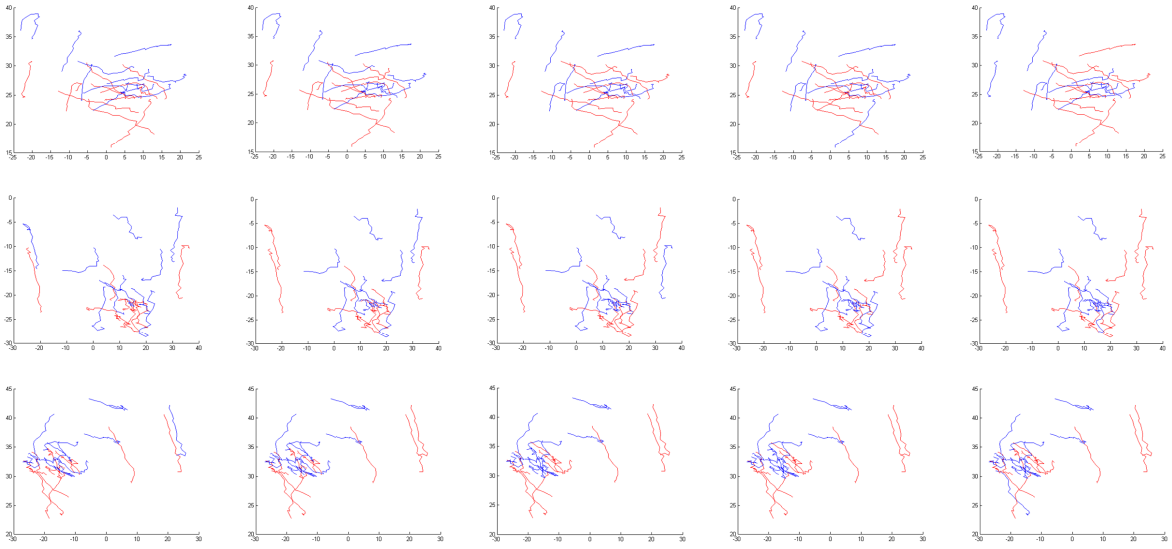
In the experiment with ground-truth trajectories, we have approximately $56 \times 4/5$ group motion samples, which may be captured in different views. For convenience and without loss of generality, we apply a homographic transform to each of them to work in a canonical view (ground plane in this work). To get sufficient exemplars to train the DP-DFM, we augment the training sets by generating new training samples from the original ones. For this purpose, we perturb each original trajectory by adding 2-D isotropic Gaussian perturbations on ground-plane coordinates at multiples of 20% of the whole motion duration, and polynomially interpolating the other time instants. In this way, we generate 25 new motion patterns from each original one. When learning a single hybrid driving force model within each (original or generated) motion pattern, we perform discriminative training, *i.e.*, we not only collect location/speed pairs from relevant (offensive) trajectories, but also take into account the irrelevant (defensive) trajectories away from the relevant ones, and include the inverse speeds from them into consideration. In addition, each speed vector is replicated a couple of times in the neighborhood of the corresponding location.

For comparison we set up three baselines. The first uses the homogeneous spatial model, *i.e.*, $K = 1$, and the second uses the time-invariant model, *i.e.*, we use a fixed hybrid driving force for all t's. Note that the second baseline is in principle similar to the model in [174]. In the third baseline, we simply regard the relevant trajectories as noisy observations of the states of a linear time-invariant dynamic system and use standard system identification approach [57] to learn the model.

We use the ratio of the correct segmented offensive players to the total offensive players, namely segmentation rate, as the numerical criterion to evaluate the performance, which is shown in Table 4.1. Samples of the segmentation results, are shown in Figure 4.4.

4.5.2 Experiment on Non-robust Trajectories

In this experiment, we simulate non-robustness by first randomly dropping a few trajectories (1, 2, 3 when training and 1, 2, 3, 4, 5 when testing) from the ground-truth, and



(a) Ground truth (b) Proposed DFM (c) Homogeneous Spatial model (d) Time-invariant model (similar to [21]) (e) Linear time-invariant system ([23])

Figure 4.4: Samples of segmentation results. In each row are a ground-truth group motion and corresponding segmentation results. Red trajectories denote the relevant objects and blue ones are irrelevant ones.

Table 4.1: The segmentation rates comparison (%).

Proposed Driving Force Model	79.7
Homogeneous Spatial Model	74.8
Time-Invariant Model	73.3
Linear Time-Invariant System Model	70.7

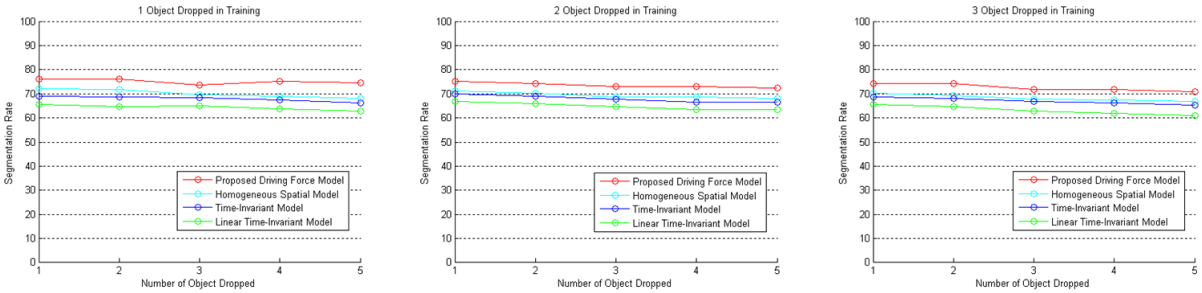


Figure 4.5: Segmentation statistics on non-robust trajectories.

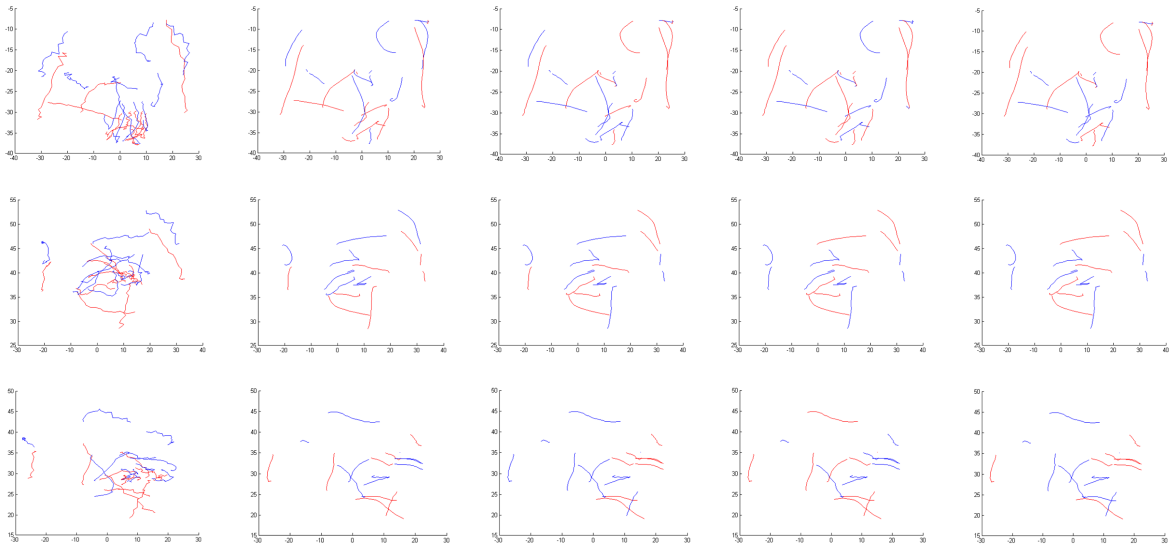
then for the remaining trajectories randomly locating the durations during which we let the trajectories disappear (using a 1/0 birth-death process model with $\lambda/\mu = 5$). The training samples are then augmented by perturbing trajectories in every continuous durations. The statistics are shown in Figure 4.5. It turns out that the segmentation performance is insensitive to the varying number of missing trajectories as well as interruptions, as expected from having a dense field and continuous sequence.

4.5.3 Experiment on Trajectories from Tracking

In this evaluation, we employ a multi-object tracker [9] rather than directly using the annotations. As before, the trajectories are then transformed into ground plane coordinates. The multi-object tracking algorithm is based on foreground detection and tends to merge multiple targets into a single one (thus loses objects) when objects are small, highly cluttered, or strongly occluded. Note that in this case no numerical statistics can be calculated due to difficulty in associating these non-robust tracks with the ground-truth. However, we show the results in Figure 4.6 for a qualitative demonstration of the performance.

4.6 Extension: Segmenting Relevant Space-Time Interest Points from Clutters

We demonstrate in this section that DP-DFM is applicable to trajectories of space-time interest points, for the purpose to segment a relevant group of points arising from relevant human actions in videos with complex cluttered background which produces irrelevant space-time interest points as well. To locate a noise-free bag of words is critical to properly



(a) Ground truth (b) Proposed DFM (c) Homogeneous Spatial model (d) Time-invariant model (similar to [21]) (e) Linear time-invariant system ([23])

Figure 4.6: Samples of segmentation results on trajectories from tracking. In each row are a ground-truth group motion and corresponding segmentation results on tracks. Red trajectories denote the relevant objects and blue ones are irrelevant ones.

employ the family of bag-of-words methods, and there has not been an effort devoted to the removal of noisy points, to the best of our knowledge. Our model, therefore, is able to fill in this gap. Technically, each point detected and tracked corresponds to a player in football plays and then all terminologies and algorithmic steps are straightforwardly transplanted, except for a global spatial warping effect of the DFM taken into account. Specifically, the group of relevant space-time interest points may only occupy a small portion of the whole area of view, and consequently we may need to shift, scale and warp the DFM globally so as to reach a best match. To achieve this, after generating Monte Carlo DFM samples in the segmentation algorithm introduced previously, we generate Monte Carlo samples of affine transformations, and warp the DFMs with them before matching against the motion extracted from the video. The mechanism of generating random affine transformations is discussed in the next Chapter under the section 'Spatial Misalignment Submanifold'.

We make use of the dataset in [176], in which five categories of articulated human actions are recorded in natural scenes with complex and dynamic background, including jumping jacks, one-handed waves, pickups, push buttons, and two-handed waves. For each category, template videos with clean background as well as testing videos with noisy background are both provided. The testing videos are partially annotated with bounding boxes and temporal durations of the relevant human actions of interest. We train a DP-DFM for each category, using the template videos and some of the testing videos with (manually inspected and selected) good boxes. Space-time interest points are extracted using the Harris corner detector and tracked by Kanade-Lucas-Tomasi tracker [177].

Sample results for segmenting relevant group of trajectories of local space-time interest points are shown in Figure 4.7, where the red color marks the relevant motion in response to the DP-DFM model learned from training videos with clean background or ground truth, while red plus green colors mark all the detected and tracked corner points. Obviously, our model effective eliminates the majority of background tracks and provides a significantly clean version of the bag of words, which is hopefully helpful in subsequent learning and inference.

4.7 Discussions

We briefly discuss a few related issues. The first is the fact that the group motion segmentation algorithm can be used for temporal detection of the group motion, *i.e.*, to

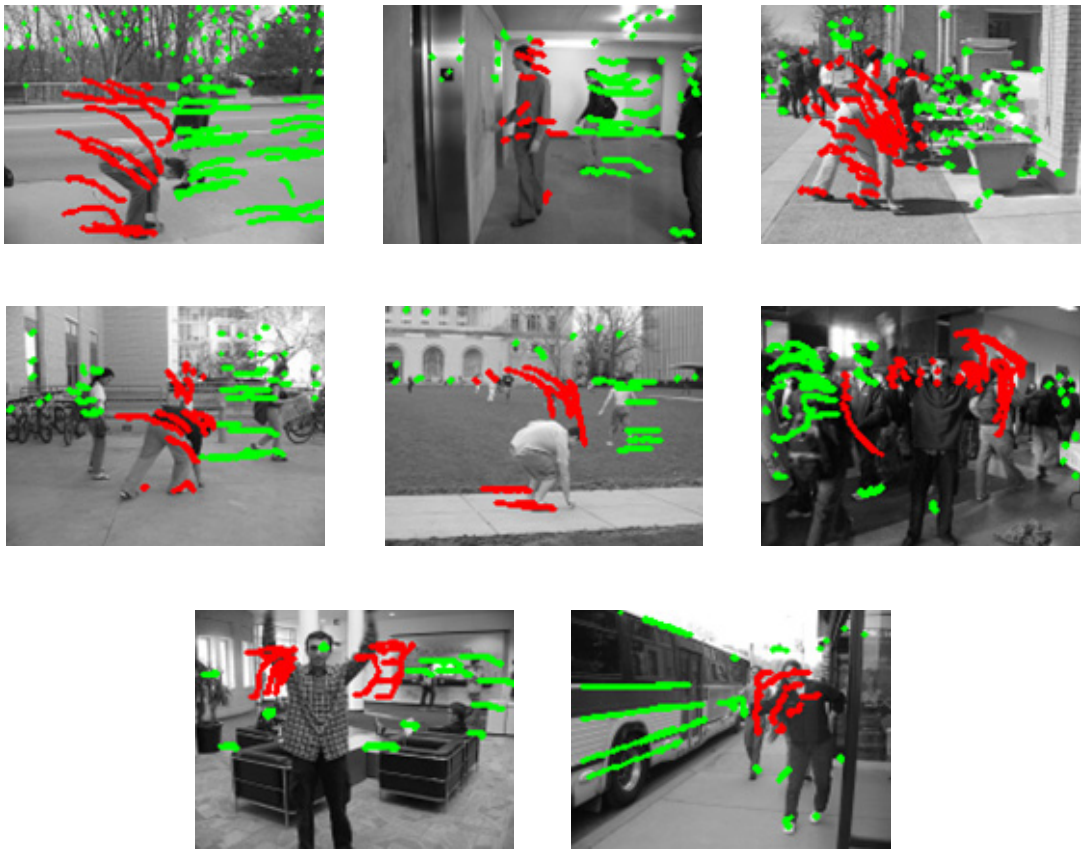


Figure 4.7: Samples of segmentation results on local space-time interest point trajectories.

determine the starting and ending location along the time axis. As the dataset used here only provides temporally segmented video clips containing a complete play, we are unable to empirically show this. However, for this purpose we simply initialize the segmentation algorithm from different time instants and identify the one(s) with the most likely match(es). Note that the algorithms can run in parallel.

A second issue is about estimating the spatial area of the group motion pattern. In football plays the participants and non-participants are homogeneously mixed all across the whole area of interest. However, in other applications the group motion pattern may only occupy a small portion of the whole area of view. In this case, we may also re-scale the field model into multiple scales and run the algorithm in parallel and in multiple scales. Within each scale we run the algorithm in dense partitions of the whole field. Note that scales and partitions with low matches in early stage can be eliminated from the candidate pool and the computational cost will keep decreasing. This strategy is similar to the one in the previous section where we explicitly model the global shift, scaling and deformation of the DFM within the generative process, and is computational expensive.

While the football play involves only one participating group, the method we presented can be extended to scenarios with multiple groups without much effort. To do so we learn a DFM per group and the testing motion pattern will be matched against every model. To get the final segmentation we simply vote for the best match.

The model is not view-invariant. We need to learn a separate model for each static view. However, static cameras are typical for surveillance and also commonly used for sports recordings. Also, the synthesis and voting based method is not computationally economical, and thus needs further improvement.

A final point is that though we designed methods in the context of group motion segmentation, the learned model, or compact features derived from it, can potentially serve as representatives of the underlying group motion. This implies a possibility that the proposed framework can be used toward the motivating application - group activity recognition addressed in previous chapter. Meanwhile, it is also expected that integration of the model into a multi-object tracker will help to improve the tracking quality due to its capability to predict potential motion. These open issues are under our further investigation.

Chapter 5

Spatio-Temporal Alignment of Two Motion Patterns (Signals)

5.1 Motivation

In this chapter we consider the problem of aligning two group motion ensemble, or more generally, two spatio-temporal signals (*i.e.*, videos, their filtered versions, or spatio-temporal features extracted from them.) which come from the same dynamic scene or the same category of dynamics. The misalignment between the two signals, captured by distinct cameras at the same time or by the same camera at different times, may result from various internal or external factors. Specifically, the internal factors include but are not limited to differences in view points, view angles, camera calibration parameters, as well as temporal shifts and scaling. Previous work on video sequence alignment mostly focused on this category of mismatch, and followed mainly two lines of approaches including feature-based approaches [178, 179, 11, 180, 12, 181, 10] and direct approaches [182, 183, 13]. In the former class, features like two-frame correspondences of interest points or trajectories of tracked objects were used as inputs to the alignment algorithm, while in the latter, intensity, color, or other pixel/patch level appearance attributes were employed. The spatial aspect of the misalignment, in most of the work, was naturally modeled as one of the parametric transforms including affine, homography, and perspective ones between the image plane coordinates of the two signals, based on different assumptions made regarding imaging conditions. The temporal misalignment, on the other hand, mainly took frame rate and shift synchronization into account, modeled as a 1-D affine transform along the time axis. The algorithms were exclusively designed for representations of the particular transforms exclusively to achieve optimal alignments. The warping parameters were then obtained using a numerical optimization method which is typically an exhaustive search or a greedy method such as gradient descent.

The first step taken in this chapter is to revisit the issue of temporal misalignment, which comes not only from the camera aspect (frame-rate and temporal shift), but also from the external factors from the observed dynamics. We look into semantically meaningful visual dynamics beyond plain spatio-temporal volumes: one of the examples of semantically

meaningful visual signals is videos recording human actions/activities. The same class of activities (*e.g.*, walking) may contain realizations executed at varying rates, though the essential characterization for that activity category should be rate independent. This rate change is in fact a temporal misalignments among realizations (signals) and is described by a non-affine time warping [139, 184]. Therefore, a complete description of the temporal misalignment regarding these signals should include time warping as well. A second concern is about the spatial aspect of the alignment algorithm, which usually pertains particularly to either feature-based methods or direct methods and is tuned to the assumed parametric spatial transform assumed. Existing algorithms are far from being scalable and flexible to easily adapt to different parametric model and different inputs. Moreover, it is always crucial to strike a balance between computational complexity and convergence towards global optimum. For example, exhaustive and greedy pursuit usually realize one of these goals at the cost of not achieving the other.

Taking all these factors into account, we reformulate the spatio-temporal alignment problem and provide a general framework and associated computational algorithms. To this end, we propose the concept of the *alignment manifold*, which is the nonlinear space of all possible spatio-temporal transformations with an intrinsic geometric characterization. We detail the construction of the alignment manifold and discuss the algebra of basic manipulations of the elements on it. The spatio-temporal signal alignment, consequently, becomes an optimization procedure on the manifold, regardless of whether the inputs are features, appearances, or other data instances from practice, provided that an objective function is properly defined to measure the misalignment of the two signal under a spatio-temporal transformation model. In particular, we present two Bayesian optimization algorithms on the manifold based on Sequential Importance Sampling (SIS) [185] , to achieve both efficiency and better convergence to the global optimum. The key idea is to regard the optimal alignment as a static state to be recursively estimated from the observed misalignment such that the posterior probability density of the estimated state reaches maximum at the true optimal alignment. In the basic SIS algorithm, an isotropic Gaussian diffusion is constructed for state particle propagation, which ignores the local first-order differential structure and leads to slow convergence. In the second algorithm, namely the Stochastic Gradient Sequential Importance Sampling (SG-SIS), we consider randomized gradient vector in the tangent space of a particular particle, which is equivalent to an alternative state transition

dynamics. Consequently, the search in the subsequent step is guided by that stochastic gradient, and results in a more efficient convergence.

In short, the contributions of this chapter are (1) we present a general framework for spatio-temporal alignment, incorporating temporal warping and various parametric spatial transforms as well as inputs; (2) we introduce the *alignment manifold*, a manifold tuned to the alignment task; and (3) a SIS algorithm and a SG-SIS algorithm specifically designed for the alignment manifold to realize a numerically optimal solution to the alignment problem.

The rest of this chapter is organized as follows. We formulate the alignment problem in Section 5.2 that includes typical examples in vision, followed by the construction of the alignment manifold in Section 5.3. We propose the SIS optimization procedure on the alignment manifold in Section 5.4, the SG-SIS algorithm in Section 5.5, and then validate their effectiveness in Section 5.6 with experiments on various typical spatio-temporal signals that arise in vision. Additional issues are discussed in Section 5.7.

5.2 The Framework of Alignment Problem

We investigate the two 3-dimensional spatio-temporal signals S^1 and S^2, whose elements are denoted as $S^1(x, y, t)$ and $S^2(x, y, t)$ respectively, where x and y represent the spatio coordinates, and t charts the temporal dimension. The spatio-temporal alignment problem aims to solve the following optimization problem

$$\min_{\mathbf{p}\in\mathfrak{M}} J(S^1, S^2, \mathbf{p}) \tag{5.1}$$

where $\mathbf{p}$ is the parameter vector specifying the alignment transform, $\mathfrak{M}$ is the alignment manifold, *i.e.*, the space of all feasible $\mathbf{p}$'s, and J is a measure of misalignment to be minimized by an optimal $\mathbf{p}$. As in previous efforts, we assume the relative calibrating parameters of the two cameras to be fixed but unknown, *i.e.*, the two cameras will remain both stationary, or move jointly. As a result, the spatial misalignment and temporal misalignment become decoupled. In other words, we may split $\mathbf{p}$ into two components as $\mathbf{p} = [\mathbf{p}_S^T, \mathbf{p}_T^T]^T$, so that the spatial and temporal misalignment can be independently handled. Note that under a much more complicated and challenging imaging condition, cameras may be under relative motion and hence the spatial misalignment and temporal misalignment are coupled. We will consider this general case in a future work. With decoupling of spatial and temporal factors, the alignment manifold $\mathfrak{M}$ is accordingly decomposed into the Cartesian product of

two submanifolds as $\mathfrak{M} = \mathfrak{M}_S \times \mathfrak{M}_T$, where $\mathbf{p}_S \in \mathfrak{M}_S$ and $\mathbf{p}_T \in \mathfrak{M}_T$. The explicit analytical form of J depends on the specific spatial and temporal transforms involved, the measure of misalignment, as well as the practical goal to be achieved. For a better illustration, we give three typical examples arising from representative vision applications.

Example 1 S^1 and S^2 are grey-level videos, the spatial displacement is 2-D affine, and temporal transform is 1-D affine. The misalignment is measured as the pixel-wise mean square error. In this case,

$$J(S^1, S^2, \mathbf{p}) = \sum_{x,y,t} (S^1(x, y, t) - S^2(x + u, y + v, t + w))^2, \tag{5.2}$$

and

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & 0 & b_1 \\ a_{21} & a_{22} & 0 & b_2 \\ 0 & 0 & a & b \end{bmatrix} \begin{bmatrix} x \\ y \\ t \\ 1 \end{bmatrix}. \tag{5.3}$$

The corresponding alignment parameter vectors are $\mathbf{p}_S = (a_{11}, a_{12}, a_{21}, a_{22}, b_1, b_2)^T$ and $\mathbf{p}_T = (a, b)^T$ with $\mathfrak{M}_S$ to be the 2-D affine group $\mathbb{A}(2)$ and $\mathfrak{M}_T$ to be $\mathbb{R}^+ \times \mathbb{R}$. ■

Example 2 S^1 and S^2 are color videos, *i.e.*, S^i contain three channels $S^i_j, j = 1, 2, 3$, spatial transform is 2-D homography, and the temporal transform is a nonlinear time warping. The misalignment is measured as the pixel-wise mean square error of the intensity. In this case,

$$J(S^1, S^2, \mathbf{p}) = \sum_j \alpha_j \sum_{x,y,t} (S^1_j(x, y, t) - S^2_j(x', y', t'))^2, \tag{5.4}$$

$$x' = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}}, \tag{5.5}$$

$$y' = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}}, \tag{5.6}$$

and $t' = W(t)$, where α_j's are the weights for the channels and $W(t)$ is the time warping function. If we denote $H = [h_{i,j}]_{3\times 3}$ to be the homography matrix with the constraint of unit determinant (*i.e.* $\det H = 1$, without loss of generality), then we have $\mathbf{p}_S = H$, $\mathbf{p}_T = W$, $\mathfrak{M}_S$ is the 3×3 special linear group $\mathbb{SL}(3)$, and $\mathfrak{M}_T$ is the set of all possible time warpings. ■

Example 3 S^1 and S^2 contain N spaces-time point trajectories respectively, *i.e.*, $S^i = \{T^i_j\}_{j=1,2,\cdots,N}$ and $T^i_j = \{(x^i_j(t), y^i_j(t))\}_t$, where $(x^1_j(t), y^1_j(t))$ $(x^2_j(t'), y^2_j(t'))$ are assumed to come from the jth tracked interest point corresponding to the same 3-D point,

captured by two pinhole cameras. Then considering perspective misalignment of the trajectories we have

$$J(S^1, S^2, \mathbf{p}) = \sum_j \sum_t \|[x_j^1(t), y_j^1(t), 1]\mathbf{F}[x_j^2(W(t)), y_j^2(W(t)), 1]^T\|^2. \tag{5.7}$$

Here $\mathbf{F}$ is the 3×3 fundamental matrix, and we may regard $\mathbf{p}_S = \mathbf{F}$ and $\mathfrak{M}_S$ to be the set of all possible fundamental matrices. ■

In practice, various objectives, besides these examples, are formulated depending on the applications to be addressed and problems to be solved, whereas most of them are essentially optimizations on the alignment manifold. As will be demonstrated, the perspective of aligning on the manifold enables generic non-linear (and possibly randomized) solutions for various formulations, in contrast to existing approaches that have been designed for a specific transform or a specific scenario. More importantly, the objectives involving realistic spatio-temporal signals are usually not analytical in the alignment parameters, which actually necessitates a generic modeling and randomized design.

5.3 The Alignment Manifold

In this section we look into the alignment manifold $\mathfrak{M} = \mathfrak{M}_S \times \mathfrak{M}_T$, whose elements characterize the alignment transforms under consideration. As the spatial and temporal factors are considered independently in this work, we are in a position to discuss them separately.

5.3.1 The Spatial Alignment Submanifold

The previous examples imply that the spatial alignment manifold $\mathfrak{M}_S$ is usually identical to a Riemannian manifold of the transformation/constraint matrices. Affine group $\mathbb{A}(2)$ and special linear group $\mathbb{SL}(3)$ both belong to the *matrix Lie group*, which possesses several intrinsic geometric properties. We list a few used in this work: The geodesic (intrinsic) distance between two elements $\mathbf{V}_1, \mathbf{V}_2$ on the matrix Lie group is

$$d(\mathbf{V}_1, \mathbf{V}_2) = ||\log(\mathbf{V}_1^{-1}\mathbf{V}_2)||. \tag{5.8}$$

The exponential map $\mathscr{E}_{v_m} : \mathscr{T}_{v_m} \to \mathbb{G}$, which maps v' in the tangent space $\mathscr{T}_{\mathbf{V}_m}$ at $\mathbf{V}_m$ onto the group $\mathbb{G}$, is given by

$$\mathscr{E}_{\mathbf{V}_m}(\mathbf{V}') = \mathbf{V}_m \exp(\mathbf{V}_m^{-1}\mathbf{V}'). \tag{5.9}$$

The logarithmic map $\mathscr{L}_{\mathbf{V}_m} : \mathbb{G} \to \mathscr{T}_{\mathbf{V}_m}$, meanwhile, is

$$\mathscr{L}_{\mathbf{V}_m}(\mathbf{V}) = \mathbf{V}_m \log(\mathbf{V}_m^{-1}\mathbf{V}). \tag{5.10}$$

Here, the matrix exponential and logarithmic operation used here are defined as

$$\exp(\mathbf{X}) = \sum_{i=0}^{\infty} \frac{1}{i!}\mathbf{X}^i \tag{5.11}$$

and

$$\log(\mathbf{X}) = \sum_{i=1}^{\infty} \frac{(-1)^{i-1}}{i}(\mathbf{X} - \mathbf{I})^i. \tag{5.12}$$

For more discussions about the geometry of matrix Lie group the reader is referred to [167].

The space of fundamental matrices - $\mathbf{F}$'s, as in Example 3, is the space of those matrices with rank 2. To get a parameterization for this manifold, we employ the singular value decomposition $\mathbf{F} = \mathbf{U}_1\mathbf{\Sigma}\mathbf{U}_2^T$, where $\mathbf{U}_1$ and $\mathbf{U}_2$ are both 3×2 orthogonal matrices and $\mathbf{\Sigma}$ is 2×2 diagonal positive. It is known that the spaces of all 3×2 orthogonal matrices is the Stiefel manifold $\mathfrak{V}_{2,3}$ [129] and thus the spatial alignment manifold $\mathfrak{M}_S = \mathfrak{V}_{2,3} \times \mathbb{R}^+ \times \mathbb{R}^+ \times \mathfrak{V}_{2,3}$. For two elements $\mathbf{V}_1, \mathbf{V}_2$ on $\mathfrak{V}_{2,3}$, an intrinsic distance is

$$d(\mathbf{V}_1, \mathbf{V}_2) = \sqrt{2 - tr(\mathbf{V}_1^T\mathbf{V}_2)}. \tag{5.13}$$

The tangent vectors at $\mathbf{V}_m$, denoted as $\mathbf{V}'$'s, can be represented as

$$\mathbf{V}' = \mathbf{V}_m\mathbf{A} + (\mathbf{I} - \mathbf{V}_m\mathbf{V}_m^T)\mathbf{B}, \tag{5.14}$$

where $\mathbf{A}$ is skew-symmetric and $\mathbf{B}$ is arbitrary. The exponential map from $\mathbf{V}'$ to $\mathbf{V}$, meanwhile, can be obtained as

$$\mathbf{V} = [\mathbf{V}_m, \mathbf{Q}] \exp\left(\begin{bmatrix} \mathbf{V}_m^T\mathbf{V}' & -\mathbf{R}^T \\ \mathbf{R} & \mathbf{0} \end{bmatrix}\right) \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \end{bmatrix} \tag{5.15}$$

where $\mathbf{Q}$ and $\mathbf{R}$ are the QR-decomposition of $(\mathbf{I} - \mathbf{V}_m\mathbf{V}_m^T)\mathbf{V}'$.

For each of the other types of spatial transformations, a proper representation for the spatial alignment submanifold should be established, and the geometry of the space should be well exploited so that computational manipulations on them are made feasible.

5.3.2 The Temporal Alignment Submanifold

As pointed out earlier, in this work we not only account for the temporal misalignment due to synchronization problem and differences in frame rates of the cameras, but also

Figure 5.1: Two realizations of the same activity 'coming to work' but with rate variations (temporal misalignment) along time axis.

exploit the rate variations among observed dynamic instances of the same category. As a toy example, let us assume that the cameras are well synchronized and share the same frame rate, while a global activity 'coming to work' is executed and observed twice during the time span [0,30]. The sub-activities in the cascade comprising the global one - opening the door, walking to the desk, and then sitting down - may occur in different time slots of [0,30] (See Figure 5.1 for visual illustration). In this case we need to non-linearly warp the sub-activities along temporal axis so that rate variation is removed among different realizations for the unambiguous specification of the global activity 'coming to work'.

Rate variation within a fixed time span, *i.e.*,[0,1], with global frame rate (scaling) and shift eliminated, is well modeled as a diffeomorphism γ from [0,1] to [0,1] with $\gamma(0) = 0$ and $\gamma(1) = 1$ [139]. Then, any time warping or misalignment $W(t)$ under consideration can be written as

$$W(t) = k_2\gamma(\frac{t - l_1}{k_1}) + l_2, \tag{5.16}$$

where k_1, k_2 are the positive global scaling factors and l_1, l_2 are the shift factors, defined for $l_1 \leq t \leq k_1 + l_1$. Obviously, when we take $\gamma(t) = t$, $W(t)$ reduces to the temporal affine transformation. Denoting the space of all possible γ's as $\mathfrak{d}$, we can now formally define the temporal alignment submanifold as $\mathfrak{M}_T = \mathbb{R}^+ \times \mathbb{R}^+ \times \mathbb{R} \times \mathbb{R} \times \mathfrak{d}$, where $\mathbb{R}^+ \times \mathbb{R}^+$ accounts for k_1, k_2 and $\mathbb{R} \times \mathbb{R}$ accounts for l_1, l_2.

If we let $\psi = \sqrt{\dot{\gamma}}$ and the space of all ψ's to be $\ominus$, then under Fisher-Rao metric (See

[149, 186]), the intrinsic distance between ψ_1 and ψ_2 are

$$d(\psi_1, \psi_2) = \cos^{-1}(< \psi_1, \psi_2 >) \tag{5.17}$$

where

$$< \psi_1, \psi_2 >= \int_0^1 \psi_1(t)\psi_2(t)dt. \tag{5.18}$$

The exponential map $\mathscr{E}_{\psi_m} : \mathscr{T}_{\psi_m} \to \ominus$ for $\psi' \in \mathscr{T}_{\psi_m}$ is defined as

$$\mathscr{E}_{\psi_m}(\psi') = \cos(< \psi', \psi' >^{\frac{1}{2}})\psi_m + \frac{\sin(< \psi', \psi' >^{\frac{1}{2}})}{< \psi', \psi' >^{\frac{1}{2}}}\psi'. \tag{5.19}$$

The logarithmic map $\mathscr{L}_{\psi_m} : \ominus \to \mathscr{T}_{\psi_m}$, which is actually the inverse map of exponential map, is then given by

$$\mathscr{L}_{\psi_m}(\psi) = \frac{\arccos(< \psi, \psi_m >)}{< \psi^*, \psi^* >^{\frac{1}{2}}}\psi^*, \tag{5.20}$$

where

$$\psi^* = \psi_m - < \psi, \psi_m > \psi. \tag{5.21}$$

Since we have used ψ instead of γ, the temporal alignment submanifold can also be equivalently represented as $\mathfrak{M}_T = \mathbb{R}^+ \times \mathbb{R}^+ \times \mathbb{R} \times \mathbb{R} \times \ominus$.

It should be noted that under this model, temporal warping of the sequence of sub-event components for global activity has been completely characterized. Changing the temporal order of the sub-event components gives rise to another global activity, and the cases with reordering effect are beyond the consideration of this work.

5.4 Sequential Importance Sampling on the Manifold for Optimal Alignment

It is now clear that the alignment problem (5.1) becomes an optimization problem on the alignment manifold $\mathfrak{M}$. This problem differs from several previous works, which have employed exhaustive or greedy strategies tuned to a specific spatio-temporal parameter space, usually Euclidean. Meanwhile, the gradient or Newton methods as used previously will tend to fall into local optimum as J defined on $\mathfrak{M}$ is normally non-convex and multi-modal, or even only locally but not globally analytical. In sum, it is desirable to find an algorithm that accounts for the non-linear manifold of the arguments, tends to converge at the global optimum, and has reasonable computational complexity.

Let us consider the following time-varying state-space model:

$$\begin{bmatrix} \mathbf{p}_{S,h} \\ \mathbf{p}_{T,h} \end{bmatrix} = \begin{bmatrix} \mathscr{E}_{\mathbf{p}_{S,h-1}}(\mathbf{u}_{S,h}) \\ \mathscr{E}_{\mathbf{p}_{T,h-1}}(\mathbf{u}_{T,h}) \end{bmatrix} \tag{5.22}$$

$$\mathbf{y}_h = J(S^1, S^2, \mathbf{p}_h) - v_h. \tag{5.23}$$

where $\mathbf{p}_h = [\mathbf{p}_{S,h}^T, \mathbf{p}_{T,h}^T]^T$ is the parameter state at step h. We assume that $\mathbf{p}^*$, the optimal alignment, is not directly observable, while at step t we observe $\mathbf{y}_t$. Moreover, we let

$$\mathbf{u}_{S,h} \sim \mathcal{N}(\mathbf{0}, (\sigma_S/h)^2\mathbf{I}), \tag{5.24}$$

$$\mathbf{u}_{T,h} \sim \mathcal{N}(\mathbf{0}, (\sigma_T/h)^2\mathbf{I}), \tag{5.25}$$

where σ_S^2 and σ_T^2 are both small numbers. By construction (details below) we may let v_h to be an non-negative random variable with an appropriate density function (*e.g.*, exponential $\mathcal{E}(\lambda)$ in this work). Equivalently, we may represent the state transition and observation model as

$$p(\mathbf{p}_{S,h}|\mathbf{p}_{S,h-1}) \sim \exp(-\frac{d^2(\mathbf{p}_{S,h}, \mathbf{p}_{S,h-1})}{2(\sigma_T/h)^2}), \tag{5.26}$$

$$p(\mathbf{p}_{T,h}|\mathbf{p}_{T,h-1}) \sim \exp(-\frac{d^2(\mathbf{p}_{T,h}, \mathbf{p}_{T,h-1})}{2(\sigma_T/h)^2}), \tag{5.27}$$

where

$$p(\mathbf{p}_h|\mathbf{p}_{h-1}) \sim p(\mathbf{p}_{S,h}|\mathbf{p}_{S,h-1})p(\mathbf{p}_{T,h}|\mathbf{p}_{T,h-1}), \tag{5.28}$$

and

$$p(\mathbf{y}_h|\mathbf{p}_h) \sim \exp(\lambda(\mathbf{y}_h - J(S^1, S^2, \mathbf{p}_h))). \tag{5.29}$$

The motivation as to why we formulate a state space model is to be able to recursively compute the Maximum A Posterior (MAP) estimate of the parameter state $p(\mathbf{p}_h|\mathbf{y}_h, \mathbf{y}_{h-1}, \cdots, \mathbf{y}_0)$. From the recursion

$$\begin{aligned} &p(\mathbf{p}_h|\mathbf{y}_h, \mathbf{y}_{h-1}, \cdots, \mathbf{y}_0) \propto (\mathbf{y}_h|\mathbf{p}_h)p(\mathbf{p}_h|\mathbf{y}_{h-1}, \mathbf{y}_{h-2}, \cdots, \mathbf{y}_0) \\ &= p(\mathbf{y}_h|\mathbf{p}_h) \int p(\mathbf{p}_h|\mathbf{p}_{h-1})p(\mathbf{p}_{h-1}|\mathbf{y}_{h-1}, \mathbf{y}_{h-2}, \cdots, \mathbf{y}_0)d\mathbf{p}_{h-1}, \end{aligned} \tag{5.30}$$

we know that the posterior probability of the alignment $p(\mathbf{p}_h|\mathbf{y}_h, \mathbf{y}_{h-1}, \cdots, \mathbf{y}_0)$ is equal to the posterior probability at the previous step $p(\mathbf{p}_{h-1}|\mathbf{y}_{h-1}, \mathbf{y}_{h-2}, \cdots, \mathbf{y}_0)$ smoothed by the state transition probability $p(\mathbf{p}_h|\mathbf{p}_{h-1})$ and weighted by the likelihood $p(\mathbf{y}_h|\mathbf{p}_h)$. Therefore, by constructing a decreasing sequence $\{\mathbf{y}_h\}_{h=0,1,\cdots}$ and letting σ_S, σ_T be small, $p(\mathbf{p}_h|\mathbf{y}_h, \mathbf{y}_{h-1}, \cdots, \mathbf{y}_0)$

is expected to be continuously increasing and peaking at the the optimal alignment $\mathbf{p}^*$. In other words, the MAP estimate of the parameter state will give the optimal alignment.

The above Bayesian recursive estimation is realized in a Monte Carlo manner. In particular, the construction of appropriate observation sequence $\{\mathbf{y}_h\}_{h=0,1,\cdots}$ come up naturally from the Monte Carlo samples. We propose the SIS algorithm on the alignment manifold as follows. Note that the proposed algorithm handles states evolving on the Riemannian manifold rather than the conventional Euclidean space, thus is different from most existing particle filters and their variations. Bayesian recursive filtering using particles has been proposed for specific manifolds in the context of tracking [187, 188, 189, 190], while the approach suggested is generally applicable for various alignment manifolds. Furthermore, we formulate the static optimization problem into a dynamic state space model, which provides insights on applications of SIS to new problems beyond tracking.

Algorithm 1 *SIS on the alignment manifold.*

1)Initialization. Specify an initial distribution p_0 defined on $\mathfrak{M}$ and draw i.i.d. samples $\{\mathbf{p}_0^k\}_{k=1}^K$ from p_0. Let $h = 1$.

2)Importance Sampling. Sample $\hat{\mathbf{p}}_h^k$ from $p(\mathbf{p}_h^k|\mathbf{p}_{h-1}^k)$. For this purpose, generate $\mathbf{u}_{S,h}^k$ from $\mathcal{N}(\mathbf{0}, (\sigma_S/h)^2\mathbf{I})$ and $\mathbf{u}_{T,h}^k$ from $\mathcal{N}(\mathbf{0}, (\sigma_T/h)^2\mathbf{I})$. Then apply exponential maps $\hat{\mathbf{p}}_{S,h}^k = \mathscr{E}_{\mathbf{p}_{S,h-1}^k}(\mathbf{u}_{S,h}^k)$ and $\hat{\mathbf{p}}_{T,h}^k = \mathscr{E}_{\mathbf{p}_{T,h-1}^k}(\mathbf{u}_{T,h}^k)$.

3)Constructing observation. Let

$$\mathbf{y}_h = \min_k J(S^1, S^2, \hat{\mathbf{p}}_h^k). \tag{5.31}$$

If $\mathbf{y}_h > \mathbf{y}_{h-1}$, $\mathbf{y}_h \longleftarrow \mathbf{y}_{h-1}$.

4)Weighting. Approximate the new posterior probability by

$$q_h(\mathbf{p}_h) = \sum_{k=1}^{K} w_h^k \delta(\mathbf{p}_h - \hat{\mathbf{p}}_h^k), \tag{5.32}$$

where δ is the Kronecker delta, $w_h^k \propto p(\mathbf{y}_h|\hat{\mathbf{p}}_h^k)$ and $\sum_{k=1}^K w_h^k = 1$.

5)Importance resampling. Draw i.i.d. samples $\{\mathbf{p}_h^k\}_{k=1}^K$ from $q_h(\mathbf{p}_h)$.

6)Stop if a stopping criteria is satisfied; Otherwise, $h \longleftarrow h+1$ and go to 2). ■

Step 3) follows from the observation equation in the proposed state-space model, and this construction of observation $\mathbf{y}_h$ plays an important role in the above algorithm. By letting $\mathbf{y}_h$ to be the minimum value of the alignment cost function, Monte Carlo samples

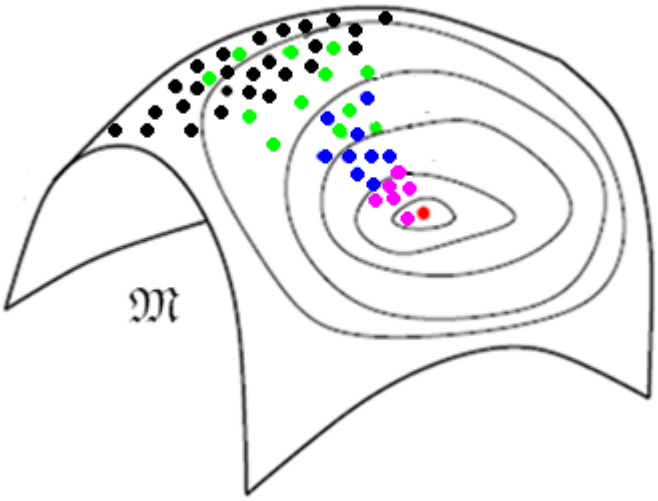


Figure 5.2: A visual illustration of SIS on the manifold. See the text for explanations.

that lead to a lower cost will receive higher importance weights when applying the weighting step. Consequently, the Monte Carlo samples (particles) will tend to concentrate around the minima of the alignment cost function, including the global minimum. With a proper initialization of samples over $\mathfrak{M}$, the optimal $\mathbf{p}^*$ will be located more and more accurately during the coarse-to-fine particle propagation. The operation $\mathbf{y}_h \longleftarrow \mathbf{y}_{h-1}$ when $\mathbf{y}_h > \mathbf{y}_{h-1}$ guarantees a non-increasing sequence of $\mathbf{y}_h$ and thus a continuously optimized solution.

A visualization of the above idea is shown in Figure 5.2, in reference to figures in [130]. The black dots represent the initial particles. The green, blue, and cyanic dots represent the particles after the first, second, and third iterations respectively. The particles converge to the minimum at the red dot. Only a single minimum is shown in this picture, while in practice particles will be concentrating around multiple local minima, and those propagating around the global one will receive the highest importance weights.

The initialization of the particles is case dependant. For the spatial alignment submanifold of $\mathbb{A}(2)$, we may generate independent, uniformly distributed samples over the corresponding Lie algebra $\eth(2)$ and exponentially map them onto $\mathbb{A}(2)$. For the temporal alignment submanifold, we may also generate uniform distributed samples over the tangent space at $\gamma(t) = t$ together with uniform samples from $\mathbb{R}^+ \times \mathbb{R}^+ \times \mathbb{R} \times \mathbb{R}$. The stopping criteria, meanwhile, can be flexible as well. $0 < \mathbf{y}_{h-1} - \mathbf{y}_h < \epsilon$ is a useful one. The final

MAP estimate of $\mathbf{p}^*$, can be simply taken as $\hat{\mathbf{p}}^* = \arg\min_k J(S^1, S^2, \hat{\mathbf{p}}_h^k)$ after the algorithm stops at step h.

5.5 Stochastic Gradient Sequential Importance Sampling for Efficient Alignment

A closer examination of the dynamical model in the previous section, especially the state transition dynamics (5.24) and (5.25), points out a drawback of the SIS algorithm presented in the previous section. By assuming $\mathbf{u}_{S,h}$ and $\mathbf{u}_{T,h}$ to be isotropic Gaussian vectors in the tangent space, we essentially assumed an isotropic diffusion pattern of the states. Consequently, the randomized search, or particle propagation, is homogenous in all 'directions' regardless of the likelihood of the observations, or values of the objectives. Apparently, if the cost function descents more steeply along a particular direction than the others, then more particles sampled toward that direction will probably lead to a lower cost and a faster convergence. Gradient descent methods, or randomized versions like stochastic gradient descent, are available for efficient search in the Euclidean space, but are not straightforward for Riemannian manifolds. An effort using stochastic gradient simulated annealing on the Grassmann manifold was attempted in [191]. We develop a stochastic gradient SIS (SG-SIS) algorithm by proposing new state transition models instead of (5.24) and (5.25), taking into account the first-order differential information. We will sequentially address the issues of 1) Gradient on the alignment manifold and its approximation due to the unavailability of an analytical form; 2) formulation of the stochastic gradient; and 3) the new SG-SIS algorithm.

5.5.1 Gradient on a Generic Manifold and Its Approximation

We commence by establishing the analogy from gradient descent in Euclidean case, and first look into gradient descent on the manifold. For $\mathbf{x} \in \mathbb{R}^D$ and an analytical function $f(\mathbf{x})$, a gradient $\nabla f(\mathbf{x})$ exists, and the steepest descent direction is given by $-\nabla f(\mathbf{x})$. To look for the next position from current location, we simply search along the straight line $\gamma(t) : t \longrightarrow (x) - t\nabla f(\mathbf{x})$. Let us denote the objective $J(S^1, S^2, \mathbf{p})$ defined at every $\mathbf{p} \in \mathfrak{M}$ as $J(\mathbf{p})$ for simplicity, we expect the a similar vector, denoted as $\nabla J(\mathbf{p})$, exists, though it can only be defined in the tangent space $T_{\mathbf{p}}\mathfrak{M}$. In this case, a geodesic $\gamma(t)$ exists, and

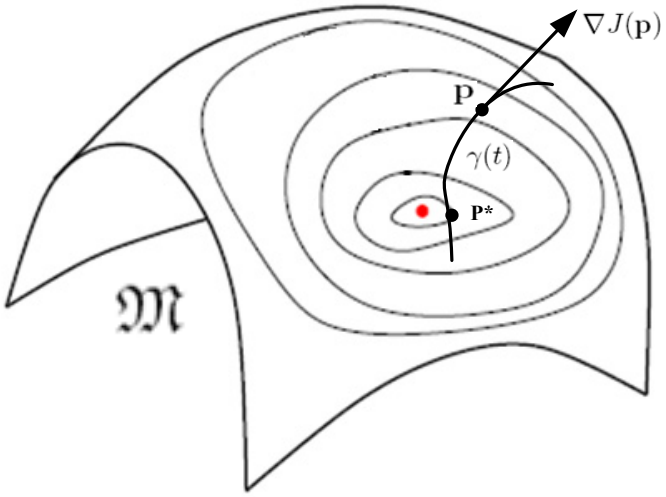


Figure 5.3: A visual illustration of gradient descent on the manifold. See the text for explanations.

satisfies $\gamma(0) = \mathbf{p}$ and $\dot{\gamma}(0) = -\nabla J(\mathbf{p})$. It is then obvious that a search along this geodesic serves as a steepest descent procedure on $\mathfrak{M}$.

A visualization of this idea is give in Figure 5.3 (partially modified from [130]). At the current point $\mathbf{p}$ the gradient $\nabla J(\mathbf{p})$ is computed and the next landing point is searched on the geodesics $\gamma(t)$ toward the direction $-\nabla J(\mathbf{p})$ until it is found at $\mathbf{p}*$ which is most close to the (local) minimum at the red dot.

It is shown in [130] that the gradient exists, and is the solution of the programming

$$\nabla J(\mathbf{p}) = \arg \max_{\xi \in T_{\mathbf{p}}\mathfrak{M}} D(J(\mathbf{p}))[\xi], \tag{5.33}$$

$$s.t. \|\xi\| = 1. \tag{5.34}$$

Here, $D(J(\mathbf{p}))[\xi]$, the directional derivative of J at $\mathbf{p}$ along the direction ξ, is defined by

$$D(J(\mathbf{p}))[\xi] \triangleq \frac{\partial}{\partial t} J(\gamma(t))|_{t=0}, \tag{5.35}$$

where the geodesic is induced by $\gamma(0) = \mathbf{p}$, and $\dot{\gamma(0)} = \xi$.

Note that the directional derivative $D(J(\mathbf{p}))[\xi]$ is not analytical in ξ for a general misalignment cost J and a generic manifold, and consequently we are in a position to approximate $D(J(\mathbf{p}))[\xi]$. Assume that the manifold under consideration is N-dimensional

(and so is the tangent space $T_{\mathbf{p}}\mathfrak{M}$), and we pick an orthonormal basis $\xi_1, \xi_2, \cdots, \xi_N$ ($\|\xi_n\| = 1, \forall n = 1, 2, \cdots, N$ and $< \xi_i, \xi_j >= 0, \forall i \neq j$) for $T_{\mathbf{p}}\mathfrak{M}$. Then any tangent vector ξ can be represented as $\xi = \sum_{n=1}^{N} a_n \xi_n$. The above programming for gradient correspondingly becomes

$$\nabla J(\mathbf{p}) = \arg \max_{a_n, n=1,2,\cdots,N} D(J(\mathbf{p}))[\sum_{n=1}^{N} a_n \xi_n], \tag{5.36}$$

$$s.t. \sum_{n=1}^{N} a_n^2 = 1. \tag{5.37}$$

The first approximation we are taking is to replace the directional derivative along the linear combination of basis vectors by the linear combination of the directional derivatives along the basis vectors, *i.e.*,

$$\nabla J(\mathbf{p}) \doteq \arg \max_{a_n, n=1,2,\cdots,N} \sum_{n=1}^{N} a_n D(J(\mathbf{p}))[\xi_n], \tag{5.38}$$

$$s.t. \sum_{n=1}^{N} a_n^2 = 1. \tag{5.39}$$

The solution to this programming, apparently, is

$$a_n = \frac{D(J(\mathbf{p}))[\xi_n]}{(\sum_{n'=1}^{N} D(J(\mathbf{p}))[\xi_{n'}])^{\frac{1}{2}}}. \tag{5.40}$$

The second approximation is to approximate $D(J(\mathbf{p}))[\xi_n]$, and we simply use

$$D(J(\mathbf{p}))[\xi_n] \doteq J(\mathscr{E}_{\mathbf{p}}(\xi_n)) - J(\mathbf{p}). \tag{5.41}$$

up to this point we have achieved an approximate expression for the gradient on a generic manifold.

5.5.2 Stochastic Gradient on a Generic Manifold

Deterministic gradient descent represents a greedy strategy and frequently converges to a locally optimal solution. On the other hand, we adopt two major approximations when computing the gradient on the manifold. These two facts necessitate a randomization of the approximate deterministic gradient, inspired by stochastic gradient methods in the Euclidean space, in the hope of converging to a global optimum. $D(J(\mathbf{p}))$ points to the direction of the maximal ascent, and we aim to allowing a directional perturbation. the

matrix Langevin distribution $L(N, 1, D(J(\mathbf{p})))$ [192], whose density function is given by

$$\frac{1}{{}_0H_1(\frac{1}{2}, \frac{1}{4}D(J(\mathbf{p}))^T D(J(\mathbf{p})))} \exp(tr(D(J(\mathbf{p}))^T X)), \tag{5.42}$$

characterizes a random unit-norm (directional) vector X whose mean (mode, median) is at $D(J(\mathbf{p}))$. Here ${}_0H_1$ is a hyper-geometric function. Therefore, we employ it as the generator of a stochastic direction around $D(J(\mathbf{p}))$. To generate a sample from $L(N, 1, D(J(\mathbf{p})))$ we adopt the acceptance-rejection method. Specifically, we generate a uniformly distributed unit-norm random vector X' in $T_{\mathbf{p}}\mathfrak{M}$ by simply normalizing an isotropic Gaussian random vector to unit length. Then, we generate a uniform random variable u from $(0, 1)$. If

$$u < \exp(tr(D(J(\mathbf{p}))^T X' - 1)), \tag{5.43}$$

we accept X' as X. Otherwise we reject and re-sample until we accept one.

After a random direction has been sampled, we are in a position to account for a random scaling. A positive continuous random variable, while the choice depends on the underlying scenario, suffices for this purpose. We use the exponential distribution in this work. The stochastic gradient thus generated from $D(J(\mathbf{p}))$ is denoted as $\tilde{D}(J(\mathbf{p}))$.

5.5.3 Stochastic Gradient Sequential Importance Sampling on the Alignment Manifold

We have been able to obtain a stochastic gradient for a generic manifold. Note that the alignment manifold $\mathfrak{M}$ is essentially the Cartesian product of $\mathfrak{M}_S$ and $\mathfrak{M}_T$, on which the stochastic gradients are $\tilde{D}_S(J(\mathbf{p}_S))$ and $\tilde{D}_T(J(\mathbf{p}_T))$ respectively. The stochastic gradient $\tilde{D}(J(\mathbf{p}))$, therefore, consists of the Cartesian product of the two components, *i.e.*,

$$\tilde{D}(J(\mathbf{p})) = (\tilde{D}_S(J(\mathbf{p}_S)), \tilde{D}_T(J(\mathbf{p}_T))). \tag{5.44}$$

At this point we are in the position to present the SG-SIS algorithm as below.

Algorithm 2 *SG-SIS on the alignment manifold.*

1)Initialization. Specify an initial distribution p_0 defined on $\mathfrak{M}$ and draw i.i.d. samples $\{\mathbf{p}_0^k\}_{k=1}^K$ from p_0. Let $h = 1$.

2)Importance Sampling. Sample $\hat{\mathbf{p}}_h^k$ from $p(\mathbf{p}_h^k|\mathbf{p}_{h-1}^k)$. For this purpose, generate stochastic gradients $\tilde{D}_S(J(\mathbf{p}_{S,h-1}^k))$ and $\tilde{D}_T(J(\mathbf{p}_{T,h-1}^k))$ as introduced above, and let

$\mathbf{u}^k_{S,h} = \tilde{D}_S(J(\mathbf{p}^k_{S,h-1}))$ and $\mathbf{u}^k_{T,h} = \tilde{D}_T(J(\mathbf{p}^k_{T,h-1}))$. Then apply exponential maps $\hat{\mathbf{p}}^k_{S,h} = \mathscr{E}_{\mathbf{p}^k_{S,h-1}}(\mathbf{u}^k_{S,h})$ and $\hat{\mathbf{p}}^k_{T,h} = \mathscr{E}_{\mathbf{p}^k_{T,h-1}}(\mathbf{u}^k_{T,h})$.

3)Constructing observation. Let

$$\mathbf{y}_h = \min_k J(S^1, S^2, \hat{\mathbf{p}}^k_h). \tag{5.45}$$

If $\mathbf{y}_h > \mathbf{y}_{h-1}$, $\mathbf{y}_h \longleftarrow \mathbf{y}_{h-1}$.

4)Weighting. Approximate the new posterior probability by

$$q_h(\mathbf{p}_h) = \sum_{k=1}^{K} w^k_h \delta(\mathbf{p}_h - \hat{\mathbf{p}}^k_h), \tag{5.46}$$

where δ is the Kronecker delta, $w^k_h \propto p(\mathbf{y}_h|\hat{\mathbf{p}}^k_h)$ and $\sum_{k=1}^{K} w^k_h = 1$.

5)Importance resampling. Draw i.i.d. samples $\{\mathbf{p}^k_h\}^K_{k=1}$ from $q_h(\mathbf{p}_h)$.

6)Stop if a stopping criteria is satisfied; Otherwise, $h \longleftarrow h+1$ and go to 2). ■

To compare Algorithm 2 with Algorithm 1, we notice no major differences from Algorithm 1, except for a 'directed' particle propagation with additional computations for pursuing a steeper descent, rather than a non-informative Gaussian diffusion. This implicitly implies a different state transition on the Riemannian manifold, though it is not straight forward to express explicitly the state transition equation. Note that the computation of a gradient expedites convergence, the approximation makes the computation of gradient possible, and the randomization of direction and scaling enable the pursuit out of potential local optimums. The extra effort spent on Algorithm 2 brings fast convergence in practice, which will be demonstrated in the following section.

5.6 Empirical Evaluation

We have applied the algorithm described above to three different datasets for the same purpose of spatial-temporal alignment, while these datasets represent different spatio-temporal signals originated from videos. Specifically, we looked into the alignment of point trajectories esembles, deforming shape sequences, as well as videos themselves. The alignment objectives and alignment manifolds corresponding to each datasets vary, while the SIS procedure is the same for all. In each experiment, we select appropriate state-of-art methods or design baseline(s) for comparison, while the purpose of these comparisons is simply to

show how the inclusion of temporal warping submanifold, formulation of the aligning procedure as a recursive estimation of the state-space model, and the Monte Carlo approach help advance the state-of-the-art performance on practical data.

5.6.1 Evaluation with Collaborative Group Motions

We first evaluate our method again on GaTech Football Dataset. Note that the GaTech Football Play Multi-Trajectory Dataset is organized into categories, each of which contains all realizations of the same play strategy (specified by the playbook). In other words, trajectory sets in the same category are samples of the same 'activity', thus resembling each other (on the ground plane) though intra-class variations exist. However, they are observed in different viewpoint and executed at different and varying rates. In each set the roles of players are annotated and thus the trajectory correspondence between two sets is available to us. This enables us to focus on the alignment algorithm rather than worry about possibly non-robust correspondence and tracking, and also provides a fair comparison. Note that though in this dataset the trajectories come from motion of objects (players), in general they can originate from local feature points extracted and tracked across consecutive frames as well.

We model the spatial misalignments to be a planar homography and thus the spatial alignment submanifold becomes $\mathbb{SL}(3)$. The misalignment cost J is simply taken as the average distances between point pairs from all trajectory pairs across the whole time span. We perform two types of experiments, in the first of which we select a set of trajectories and transform it with a typical view change (homography) and a specific time warping to get the other, and then we align the two. We do so on all 55 sets. In the second type, we randomly select a total of 40 pairs of sets, each pair being the samples of the same play type (activity), and then we align these pairs. For comparative purposes, we implemented two state-of-art methods [10, 11] that address similar task as ours. The approach in [10] assumes affine temporal misalignment only, and the strategy in [11] uses Dynamic Time Warping (DTW) to determine the non-linear temporal misalignment. The preprocessing modules of tracking and correspondence in the two methods are unnecessary as the dataset has provided trajectory and correspondence information, and thus a common basis is shared among all implementations for comparison. Note that [10] mainly focus on temporal alignment, and to add spatial alignment into it we simply estimate a planar homography with

Table 5.1: Average residual misalignments between the aligned trajectory pairs.

	Using [10]	Using [11]	SIS on manifold	SG-SIS on manifold
mean	15.9	13.1	10.0	9.8
standard deviation	8.6	6.8	3.6	4.0

the points from the temporally aligned trajectories. Meanwhile, when using [11] we take alternations between DTW and gradient-descent-based homography estimation (on all corresponding points collected from all temporally aligned frames) to get the final alignment parameters. (Note that though DTW is globally optimal in 1-D temporal dimension, when placed into alternations between spatial and temporal submanifolds the combined search may not necessarily be so, and thus the alternating process is a greedy search.) For our method, we get the initial particles by generating random samples in the tangent space at the homography estimated from the first pair frames and in the tangent space at $\gamma(t) = t$.

Samples of the results are shown in Figure 5.4, where in the first two rows are the two trajectory sets to be aligned toward each other, and the following two rows show alignment results using the state-of-art methods and our method. Each of the three columns, meanwhile, represents a typical experimental setting: in column (a) the target is a generated misaligned version of the reference, in column (b) the reference and target are a real pair with similar realization but undergoing significant misalignment, and in column (c) the two are a real pair with significant variation from each other.

To quantitatively understand the performance of the alignment methods, we recorded the average distance of point pairs from aligned trajectory pairs, and show the results in Figure 5.8 and Table 5.1. Note that the statistics is from the 40 real pairs rather than the generated ones.

It is also interesting to investigate the efficiency that the randomized gradient decent search brings. In Figure 5.6(a) we show the average convergence curves for the basic SIS and SG-SIS algorithms, where a faster descending residual is observed for the involved stochastic gradient.

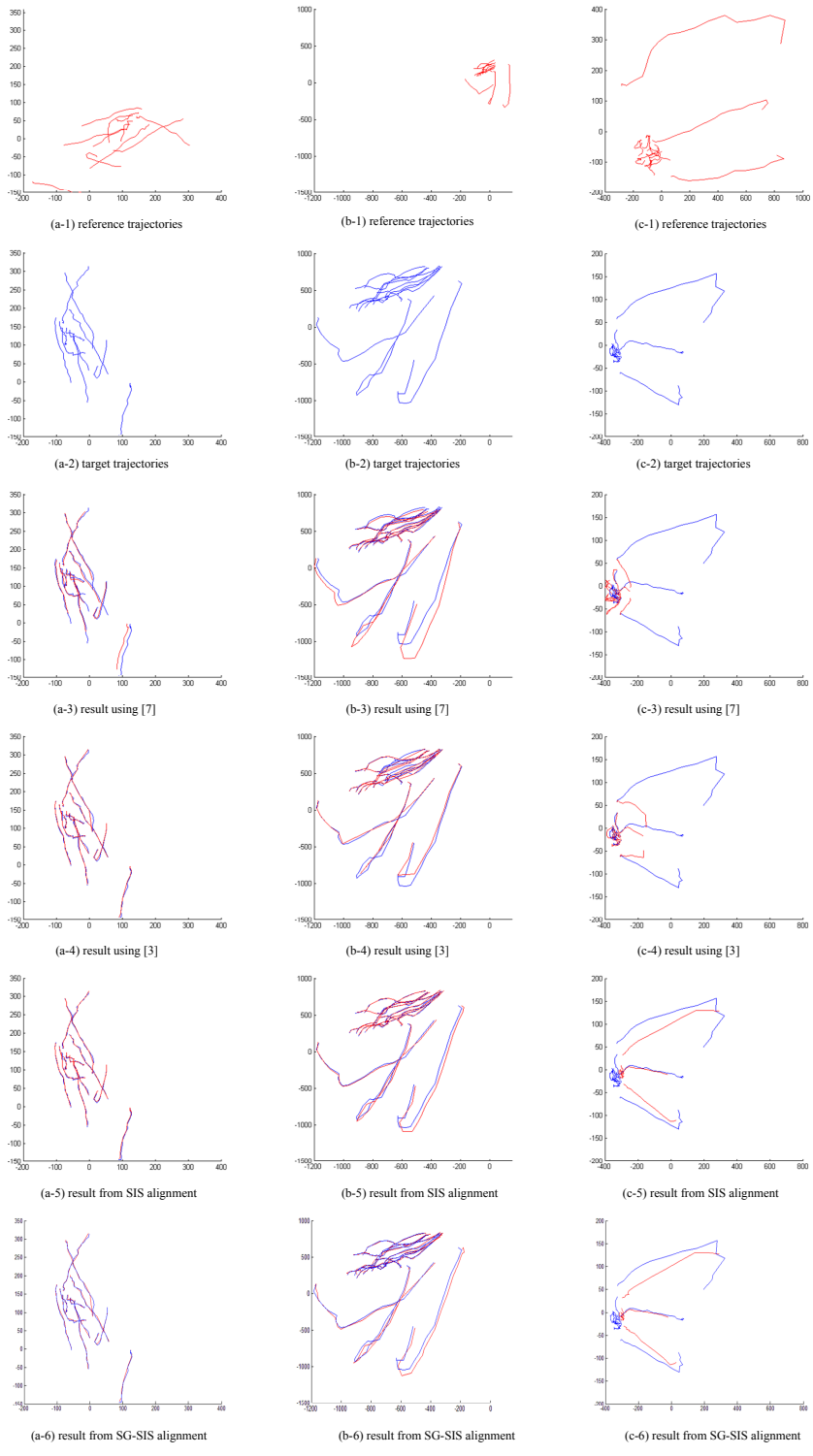


Figure 5.4: Samples of the alignment results on point trajectories using baselines and the two algorithms proposed in this paper.

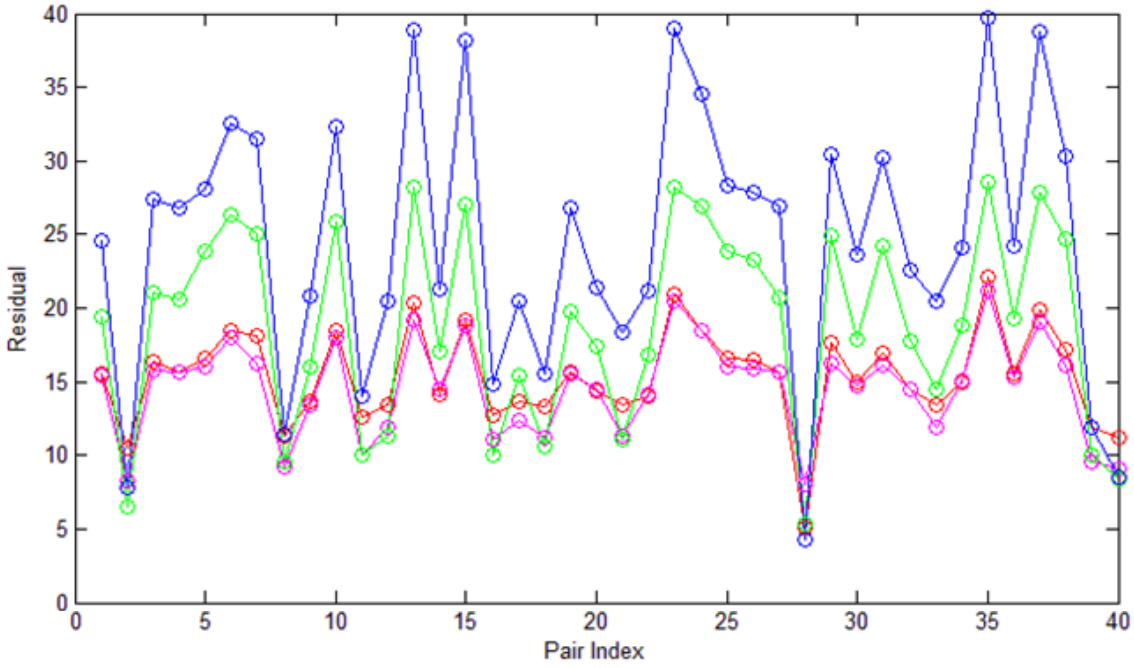


Figure 5.5: Residual misalignment of the 40 pairs of trajectory ensembles. Blue, green, red and megenta dots represent the results using [10], using [11], using the basic SIS algorithm, and using SG-SIS algorithm respectively.

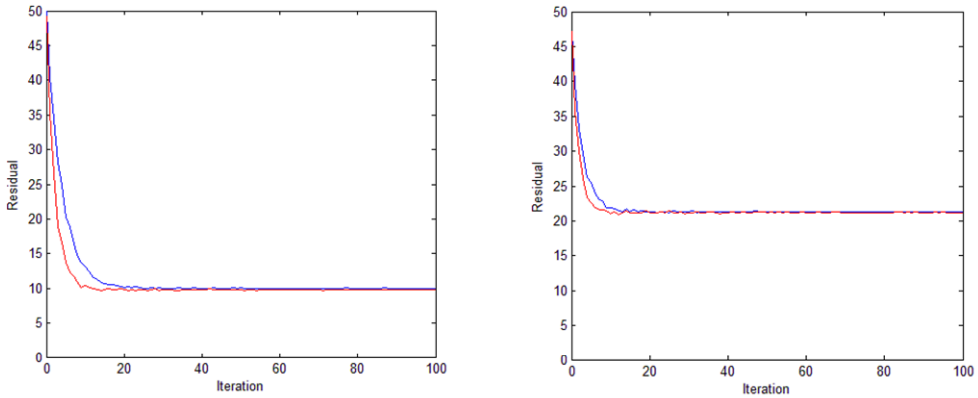


Figure 5.6: Average convergence curves for (a) 40 pairs from Gatech Multi-Trajectory Dataset, and (b)20 pairs from USF Gait Database. The blue curve corresponds to the residuals v.s. iterations for SIS algorithm, and the red curves corresponds to those for SG-SIS algorithm.

5.6.2 Evaluation with Deforming Shape Sequences

Sequences of deforming shapes are typical mid-level features extracted from original videos containing the deforming objects of interest. In this experiment we use silhouette sequences from the USF Gait Database [193] to demonstrate the performance of out method. We randomly select 20 sequence pairs, each with the same shoe types, carrying conditions, surface types, and walking directions, but observed at two different times. For efficiency, in each sequence we only consider the segment of frames of the first two walking circles. The spatial misalignment within each pair is modeled as affine and is actually less significant compared to the GaTech Football Multi-Trajectory Dataset, and the main focus is on the effect of taking non-linear rate warping into account in addition to linear scaling and shift. For comparison, we implemented the gradient descent algorithm presented in [12] and designed one more baseline. The designed baseline alternates between DTW (on all frames from spatial alignment) and gradient-descent-based affine estimation (on all frame pairs temporally aligned), and thus is a greedy search. The cost function is simply taken as the sum of pixel-wise absolute differences. For [12], the initial spatial parameter is estimated as a translation between the leading frames and the initial temporal parameter is taken as

Table 5.2: Average residual misalignments between the aligned pairs of shape sequences.

	Using [12]	Alternating DTW/affine	SIS on manifold	SG-SIS on manifold
mean	23.5	26.7	21.3	21.0
standard deviation	7.1	6.8	7.1	8.0

$\gamma(t) = t$. For our method, Monte Carlo samples are generated from Gaussians in the tangent spaces at the initial parameters.

We show three sample results in Figure 5.7, where each of the six rows for each sequence pair is explained in the caption of the figure. The average residual misalignment errors (in pixels) for all 20 pairs are shown in Table 5.2. Note that all three methods perform well due to mild spatial misalignment and near-affine temporal misalignment, while our method achieves improvement over [12] by allowing non-linear warping effect, and the improvement over alternating DTW and affine estimation should be credited to better global convergency. The average convergence curves for the basic SIS and SG-SIS algorithms are shown in Figure 5.6(b).

5.6.3 Evaluation with Human Action Videos

In the third set of experiments, we work with human action videos directly. We use the KTH database [194], in which the semantically meaningful signal is human motion. We randomly select 30 pairs of sequences, each pair performing the same action, but moderate variations in clothing, background, or view angle exist within the pair. For efficiency, again for each sequence we only keep a segment of frames including human motion but discard pure background frames. The spatial misalignment within each pair is affine [13], and the misalignment cost is the spatio-temporal correlation used by [13] but on optical flow extracted from consecutive frames. For comparison, we implemented [13] and the method that alternates between DTW and affine estimation as in previous section. The initial spatial parameter, when necessary, is estimated as the translation between the leading frames and the initial temporal parameter is taken as $\gamma(t) = t$. Meanwhile, Monte Carlo samples are generated from Gaussians in the tangent spaces at the initial parameters too.

We show three sample results in Figure 5.9, where each of the five rows for each

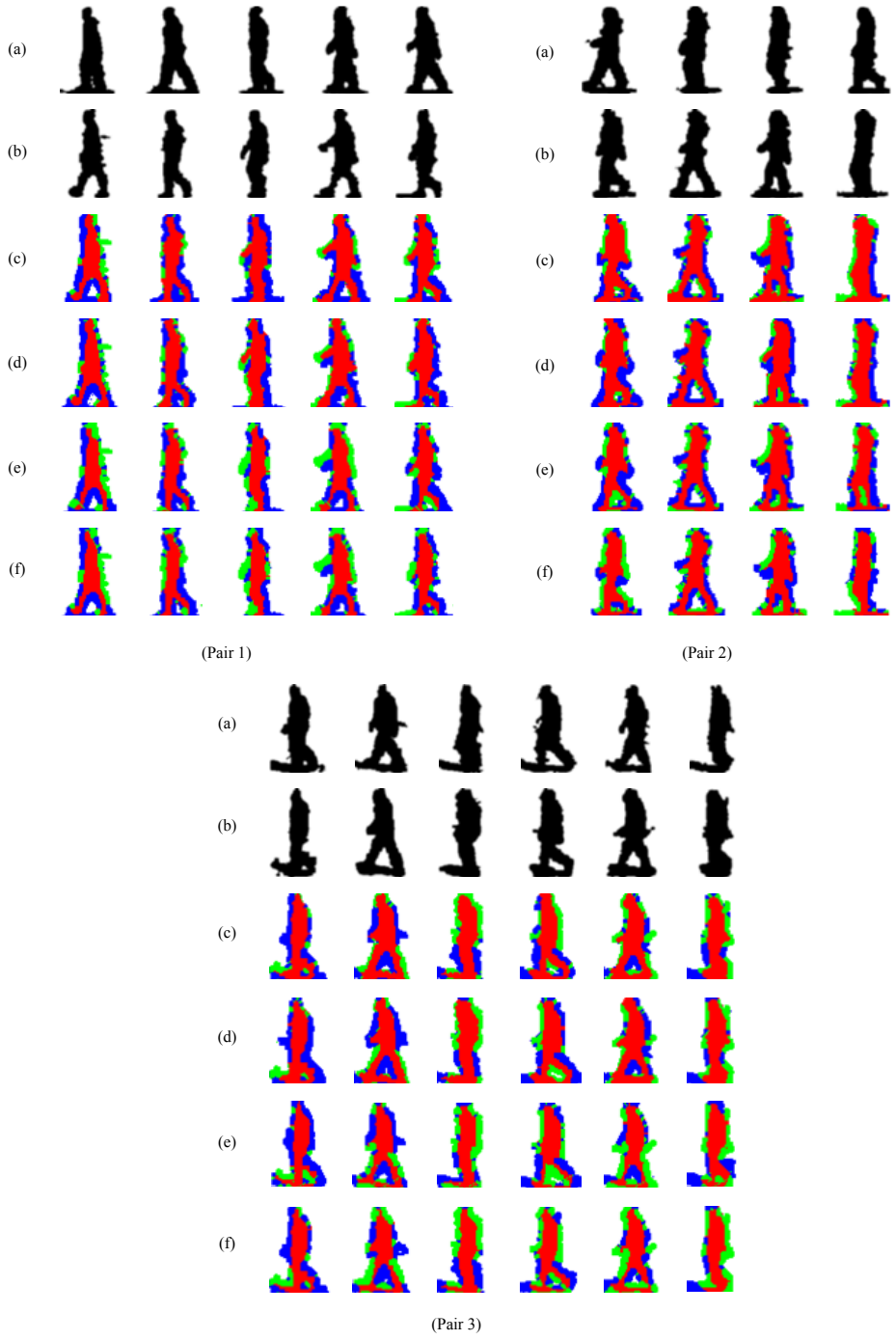


Figure 5.7: Samples of the alignment results on deforming shape sequences from the USF Gait Database. For each pair, (a) is the reference sequence and (b) is the target. (c), (d), (e) and (f) give the alignment results (transformed sequence overlaid onto target) using SIS, SG-SIS, the method in [12], and the method that alternates between DTW and spatial alignment. The red, green, blue, and white areas denote true positive, false negative, false positive and true negative respectively. In other words, a larger red area implies a better alignment.

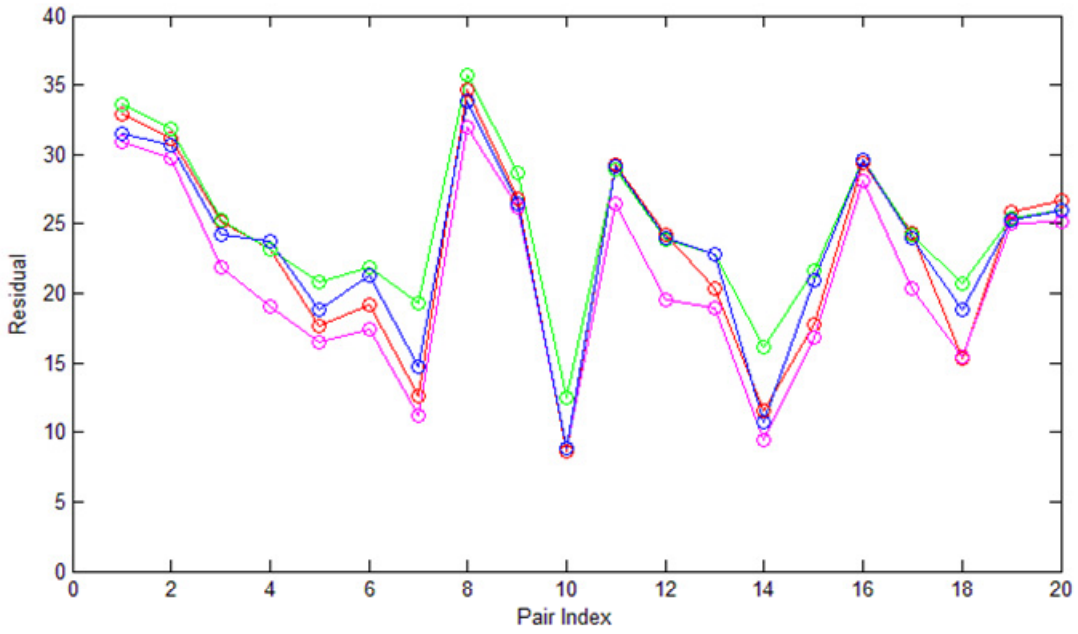


Figure 5.8: Residual misalignment of the 20 pairs of gait sequences. Blue, green, red and megenta dots represent the results using [12], using DTW/Affine alternation, using the basic SIS algorithm, and using SG-SIS algorithm respectively.

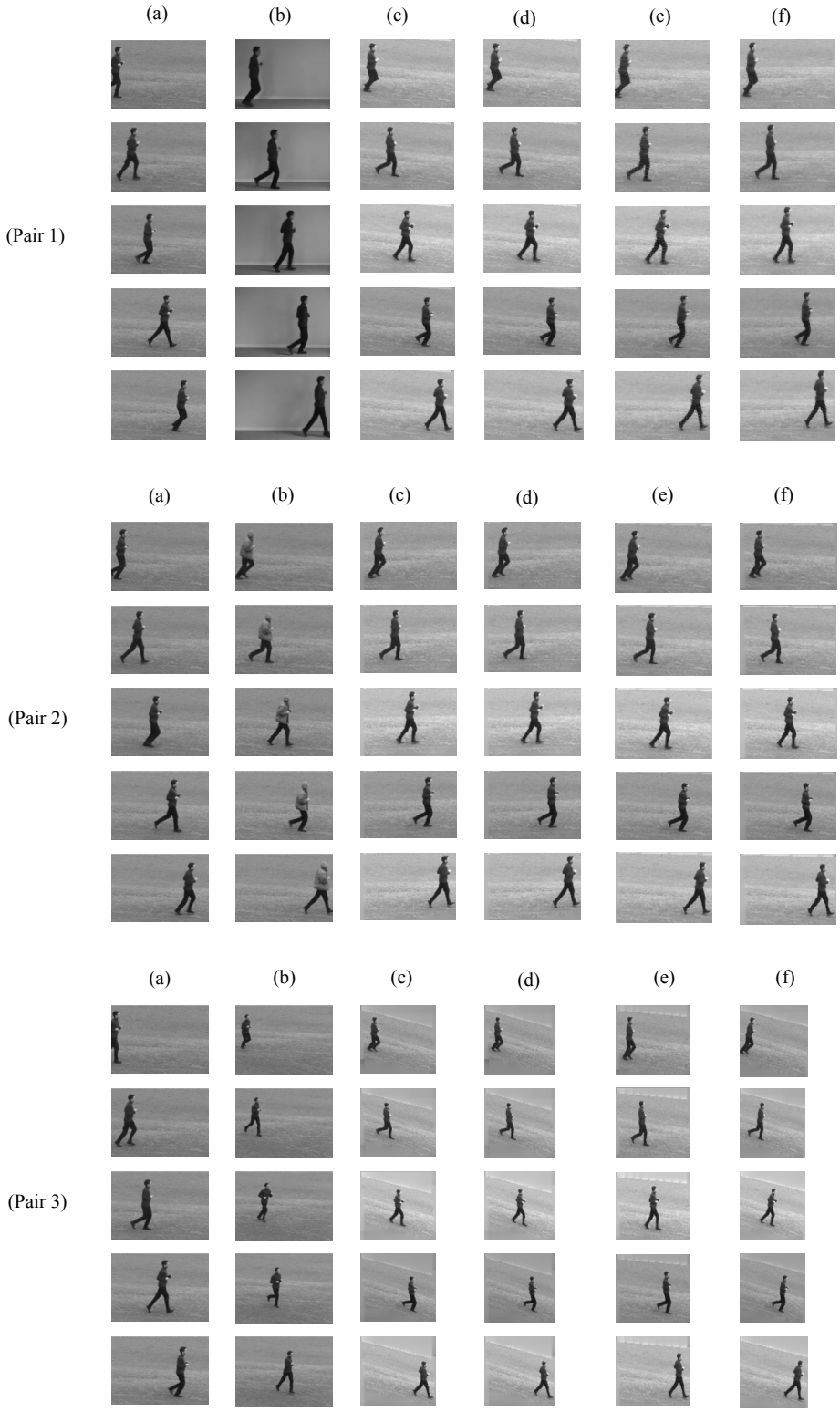


Figure 5.9: Samples of the alignment results on KTH dataset. For each pair, (a) is the reference sequence and (b) is the target. (c), (d), (e), and (f) give the alignment results using SIS, SG-SIS, the method in [13], and the method of alternation between DTW and spatial alignment.

sequence pair has the same interpretation as in the previous section. Substantial execution rate variations exist within every pair, and changes in clothing, background, or view angle also exist. There is not a numerical criterion to evaluate the performance on aligning real videos, and by qualitative observation the proposed method performs comparatively well as the baselines, and is visually more close to the target when undergoing a larger view change (pair 3).

As is true for many efforts involving particle filters, the proposed method is computationally more demanding than greedy search, but much less expensive than exhaustive approaches. This trade-off, however, leads to improved performance as demonstrated in the previous subsections. The time complexity depends on the number of particles used. The convergence, on the other hand, turns out to be fast. In this section all results are obtained with 1000 particles and less than twenty iterations. Another issue is that $\mathfrak{d}$ is by definition infinitely dimensional, while in all experiments we approximated the γ's with non-decreasing sequences valued from 0 to 1 of length 20.

5.7 Discussion

This chapter assumes that the parametric manifolds are known a priori; alignment problems without knowing the specific form of the manifolds deserve exploration as well. It is also desirable to remove the assumption regarding relative stationarity between cameras. Though we pursue global optimum in the algorithm and empirically observe improved solution, we have not theoretically proved any properties regarding asymptotic convergence. A theoretical study on geometrical SIS and SG-SIS methods will be important. We will also look into other efficient search schemes like stochastic gradient descent. By generalizing the considered manifolds and cost functions, we will extend the proposed strategy of stochastic optimization on geometric spaces for other problems (*e.g.* face alignment on Grassmann manifold [142]). We hope this can bring new insights and improved performance to a larger number of vision applications.

Chapter 6

A Generative Model for Joint Segmentation and Recognition

6.1 Motivation

In this chapter we jointly recognize and segment collaborative group motions involving multiple objects by proposing a generative model which describes the spatial coordinations among objects. By 'segmenting' a group motion we again mean to divide all objects in motion into two sets, one of which behaves according to a group-wise motion pattern. Recognition of group motion and Segmentation of relevant group motion have been individually addressed in Chapters 3 and 4, while simultaneous recognition and segmentation has not been addressed and is now attempted in this chapter. The technical framework to achieve this goal is straight-forward though not necessarily trivial: We build a generative model to incorporate all the semantic components contributing to a football play and execute Bayesian inference on it. It is worth noting that another recent effort has been devoted to recognizing and segmenting meaningful shapes from point cloud [16]. Our work can be seen as extending their 'static' problem of shape to a 'dynamic' version that includes motion. We emphasize that our segmentation is motion based only, *i.e.*, we do not use other discriminative features. In other words, we are not locating the offensive players by color patterns of their clothing.

As reviewed in Chapter 2, view-invariance is one of the desirable properties for human motion/activity recognition. View-invariant techniques have been proposed in various settings to different extents [195, 95, 39, 92, 196, 96, 197] where they are generally applied to single object actions only. Group motion modeling and recognition, meanwhile, has taken little view-invariance into account. In practice, many group motions occur in a restricted spatial domain like a ground plane. This is the case for football and soccer plays which take place on the field when we treat players as moving points. Meanwhile, the practically possible view changes are normally non-uniform but also spatially constrained: the locations of cameras observing football plays are not likely to spread all over the field. Under these considerations we explicitly characterize the view variation statistically for group motions, and consequently achieve view-invariant recognition and segmentation.

We establish the generative model from group motion in detail in Section 6.2, followed by the simultaneous recognition and segmentation algorithm in Section 6.3. Experimental evaluation is presented in Section 6.4 with discussion. The co-occurrence function, a component of the model, is also used as a discriminative descriptor to classify group group motion s in Section 6.5. Discussions are in Section 6.6.

6.2 Probabilistic Generative Model for an Collaborative Group Motion Observation

Given an group motion observation O, we would like to classify it into A, one of the group motion types of interest to us. We aim to achieve this by maximizing the posterior probability $P(A|O)$, which is usually evaluated as $P(O|A)P(A)$. Then it follows that the generative part $P(O|A)$, which describes the formation of observation from a group motion class, becomes crucial. The observed data O is essentially a collection of n motion trajectories in the image plane, among which $m(m < n)$ ones are those involved in the relevant group motion and the other $n - m$ are irrelevant ones. In football plays, for instance, we may take $n = 22$ and $m = 11$ to recognize the offensive play from the whole motion information of all players. To model the observation generating process from A to O, we decompose $P(O|A)$ into several components corresponding to sequential data-formation steps as follows

$$\begin{aligned} P(O|A) = \int_{f,D,T,v} & P(O|T,v)P(v)P(T|D) \\ & P(D|f,A)P(f|A)dfdDdTdv \end{aligned} \tag{6.1}$$

where the Markovian property is assumed as needed.

Now we explain each of the above quantities and the corresponding data generating step. 1) f is called the spatial co-occurrence function which is used to describe the spatial distribution of different types of motion trajectories involved in the relevant group activity, and $P(f|A)$ characterizes the possible variation of f. 2) D is the collection of m trajectories in the real ground area, and $P(D|f, A)$ tells us how to generate the true motion for the relevant activity in the field from a co-occurrence function. 3) T denotes the entire set of n trajectories, including those from irrelevant objects, and $P(T|D)$ provides the mechanism giving rise to the complete motion information in the ground plane. 4) v, finally, is the view transform which brings the trajectories from ground plane into image plane, while $P(v)$ is essentially the statistical characterization of the view change.

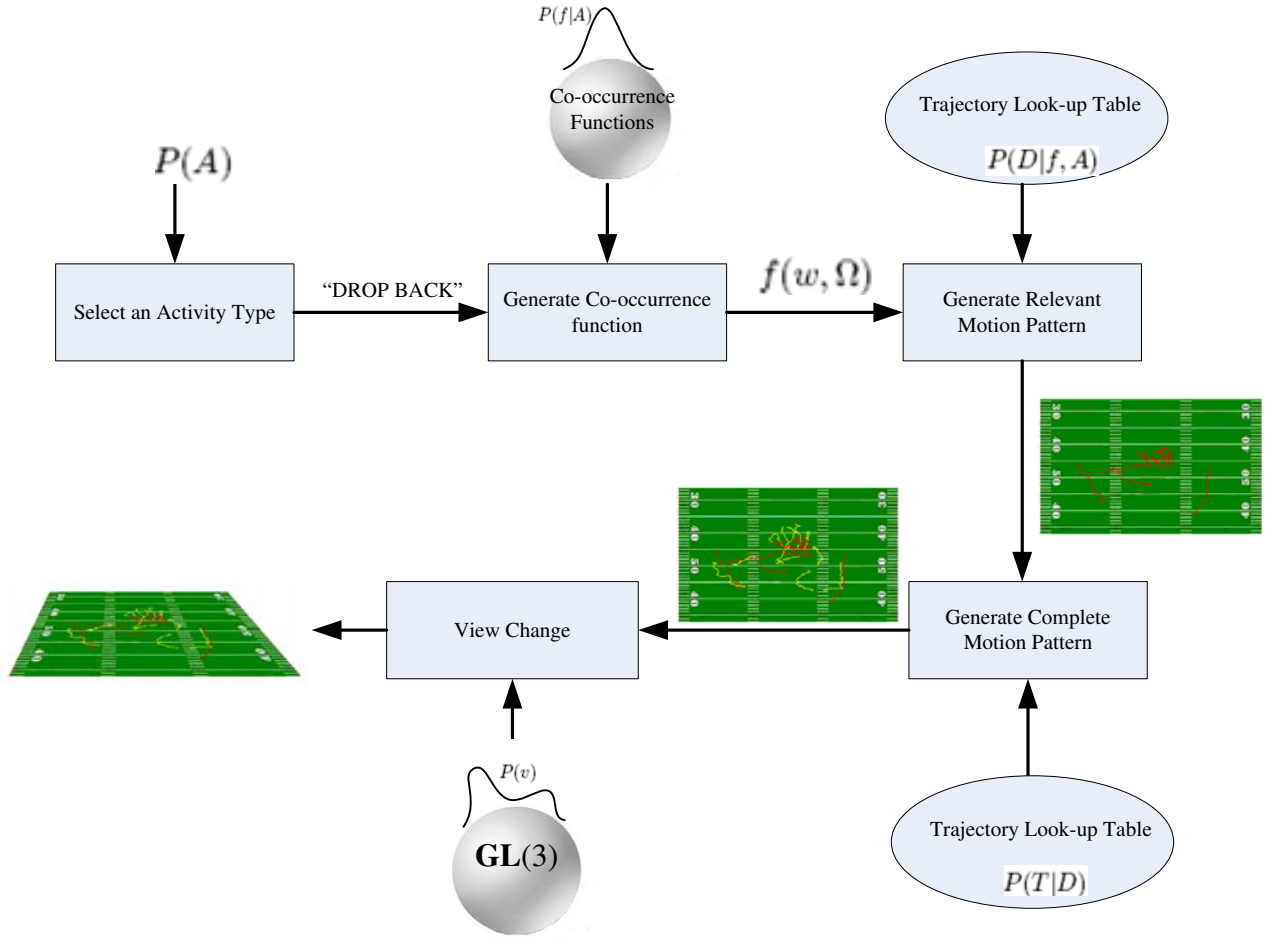


Figure 6.1: The statistical generative model (graphical model) for a group activity.

We elaborate on each of these factors now in the following subsections. For a visual illustration of the generative model, please see Figure 6.1.

6.2.1 From Motion Type to Co-occurrence Function

The spatial co-occurrence function $f(w, \Omega)$ is a two argument non-negative one with $w \in \mathscr{W}$ as the label of a trajectory type and $\Omega \in \Pi$ as the label of a spatial area/partition of the field where the motion occurs. If $F(w, \Omega)$ gives the number of occurrences of single-object motion w within the spatial area Ω, then f is defined as the square root of F. A simple toy example for football, just for illustration, is that $\mathscr{W} = \{$acceleration, deceleration, left turn,

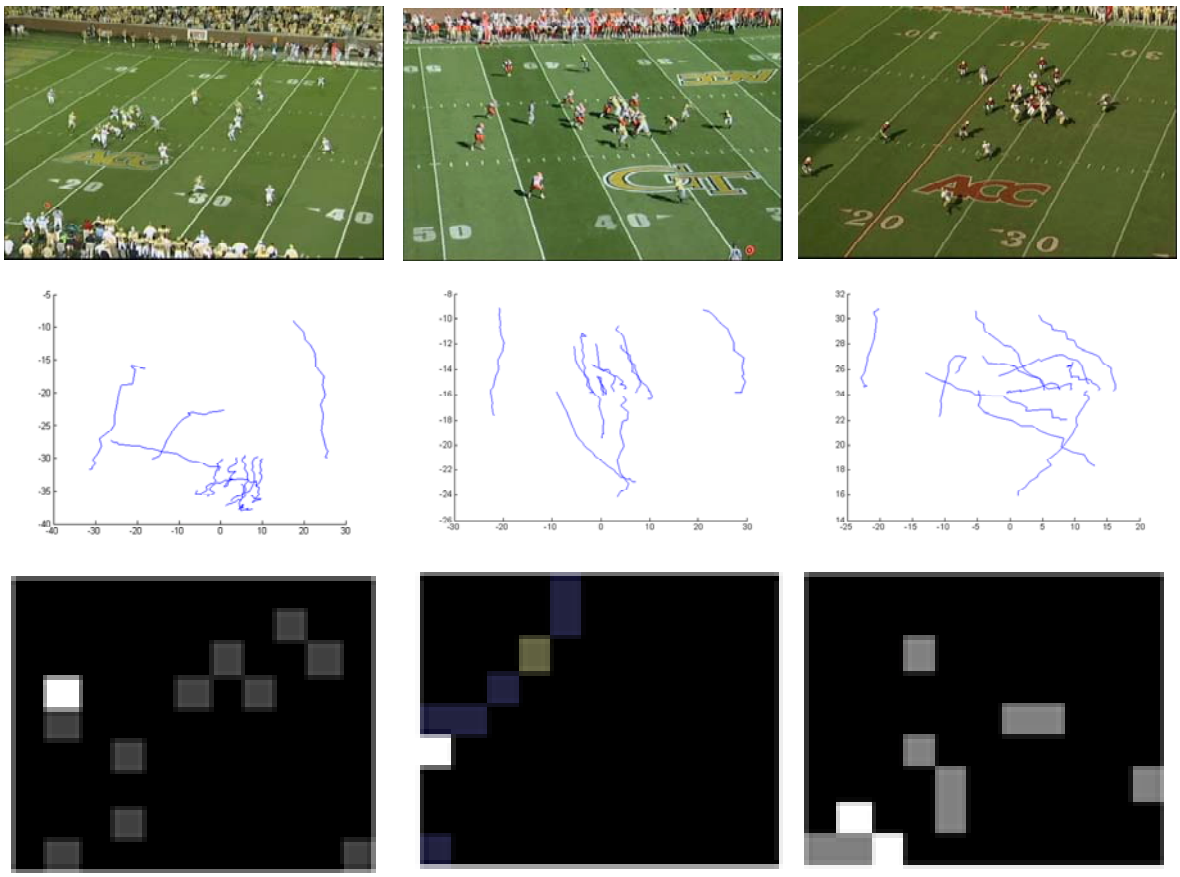


Figure 6.2: Top row: snapshots of plays; middle row: offensive trajectories in ground coordinate; bottom row: corresponding co-occurrence functions.

right turn} and Π = {middle of the field, side of the field}. Then F(acceleration, middle of the field) = 2, or f(acceleration, middle of the field) = $\sqrt{2}$, means there are two objects accelerating in the middle of the field during the group activity. In practice, we do not arbitrarily specify the trajectory types $\mathscr{W}$ as above, but learn a 'vocabulary' of all possible types in an unsupervised manner, as explained in subsection 6.2.1. To effectively describe the spatial distribution of each type of trajectories, we only take into account the minimum square area enclosing all trajectories, and partition it into *front, middle, rear* vertically and *left, middle, right* horizontally, totalling to nine possible spatial areas. Other partition schemes, obviously, can also be attempted to achieve different performance/complexity, while we use the above 3×3 pattern for simplicity.

A particular instance of a group motion, *i.e.*, m motion trajectories in the field, will correspond to a co-occurrence function, which in a sense serves as a 'descriptor' of that activity. Given a group motion type A and its instances, a collection of co-occurrence functions f's from these instances is available to characterize this type of activity. Figure 6.2 gives several samples of football plays and corresponding co-occurrence functions. To statistically capture the variability of f for a particular A, *i.e.*,$P(f|A)$, we propose to use a parametric model as

$$P(f|A) = p(f; \mu_A, \sigma_A, z_A) = \frac{1}{z_A} \exp(-\frac{d^2(f, \mu_A)}{2\sigma_A^2}), \tag{6.2}$$

which behaves as a 'Gaussian' distribution. The reason why it is not Gaussian is that the space of all f's for a fixed number of objects is not Euclidean but a curved manifold, on which the intrinsic distances between f_1 and f_2 is

$$\begin{aligned} d(f_1, f_2) &= \cos^{-1} < f_1, f_2 > \\ &\triangleq \cos^{-1}(\sum_{w \in \mathscr{W}} \sum_{\Omega \in \Pi} f_1(w, \Omega) f_2(w, \Omega)). \end{aligned} \tag{6.3}$$

However, the parameters μ_A, σ_A, z_A have similar physical meaning as the 'center', the 'scattering' as well as the normalizing factor respectively. Note that we use the compact parametric form above for its simplicity and effectiveness (as shown in the experiment), but more complex and elegant ones can also be considered whenever possible and necessary.

To learn the parameters μ_A, σ_A, z_A from a set of training co-occurrence functions $\{f_i\}_{i=1}^N$, we need more knowledge about the geometry of the manifold of co-occurrence functions, which are briefly listed in appendix. With them, the 'center' μ_A is obtained by the iteration

$$\mu_A'^{(g+1)} = \frac{\sum_{i=1}^N \mathscr{L}_{\mu_A^{(g)}}(f_i)}{N} \tag{6.4}$$

and

$$\mu_A^{(g+1)} = \mathscr{E}_{\mu_A^{(g)}}(\mu_A'^{(g+1)}) \tag{6.5}$$

where $\mathscr{L}$ and $\mathscr{E}$ are logarithmic and exponential maps respectively (see appendix). The 'scattering' σ_A, then, is calculated as

$$\sigma_A = (\frac{\sum_{i=1}^N d^2(f_i, \mu_A)}{N})^{1/2}. \tag{6.6}$$

The normalizing number z_A can be estimated by Monte Carlo simulation, which is however unnecessary in this chapter. It is worth noting that the co-occurrence functions investigated

here are closely related to and rooted in the theory of information geometry [150, 151], to which interested readers are referred for further study.

To get a new instance of co-occurrence function f of type A, we need to generate new samples from $P(f|A)$. For this purpose, we generate a function f' in $\mathscr{T}_{\mu_A}$, the tangent space at μ_A, such that $< f', f' >= 1$. Then we generate Gaussian random number $r \sim \mathscr{N}(0, \sigma_A^2)$ and obtain the new co-occurrence function as

$$f = \cos(r)\mu_A + \sin(r)f'. \tag{6.7}$$

One issue associated with this approach is that negative values of f may occur, and in this case we discard the generated sample. Another issue comes from the integer requirement of F which may not be satisfied by the generated function. To overcome it we change the generated co-occurrence function to the closest one of integer value.

6.2.1.1 Learning a Vocabulary of Trajectories

Now we explain how to learn a vocabulary of trajectories, *i.e.*, to find the set $\mathscr{W}$ in an automatic manner. We assume trajectories of individual objects are available, and denote a particular one as $X_t, t \in \mathfrak{T} = \{t_0, t_1, \cdots, t_f\}$.The collection of trajectories is evaluated in a pairwise manner to define a (dis)similarity index between every pair. The trajectories collected may take place with different starting times and locations, but the shape of the curves are actually the same, representing the same 'word' in the vocabulary. Therefore, a temporal and spatial alignment of the trajectories is necessary. With one trajectory X_t defined above and another trajectory $Y_s, s \in \mathfrak{S} = \{s_0, s_1, \cdots, s_f\}$ we define the aligned trajectories with respect to Y as

$$X'_t = X_{t+t_0} - X_{t_0}, t \in \mathfrak{T}' = \{0, 1, 2, \cdots, t_f - t_0\} \tag{6.8}$$

and

$$Y'_s = Y_{s+s_0} - Y_{s_0}, s \in \mathfrak{S}' = \{0, 1, 2, \cdots, s_f - s_0\}. \tag{6.9}$$

Note that essentially $\mathfrak{T}' = \mathfrak{S}'$. By this temporal and spatial shifting, we relocate X as well as Y at the spatial temporal origin. With the aligned trajectories X' and Y', the pairwise dissimilarity measure is taken as

$$dis(X, Y) = \sum_{r \in \mathfrak{T}'} ||X'_r - Y'_r|| \tag{6.10}$$

which will be fed to the unsupervised clustering procedure.

We only take into account the path distance in the above treatment. As mentioned before, the first and second order derivatives of X and Y may be strong features to describe the motion, and accordingly the above distance can be easily extended to include more features. Another possible approach is to incorporate a time warping when comparing two trajectories as reported in (e.g.[198]). Here we do not regard an trajectory as a stationary curve, but a *time-indexed* one, meaning that two events are different when they are executed with varied rate. An acceleration and a deceleration, for example, though along the same path across the same time span, are not regarded as same in our treatment.

To obtain a vocabulary, we cluster all training trajectories into subsets, where the intra-subset similarity of the trajectories is high but inter-subset similarity is low. Each of these subsets is identified as an 'word' and those in the subset are treated as instances of the same word. Once pairwise dissimilarity (*i.e.* similarity) is established, an obvious strategy for automatic clustering is spectral clustering (e.g. the popular Ncut method [199]) and its extensions. Nevertheless, majority of spectral clustering have their disadvantages in a word-mining setting as we have here. For a multiple-word discovery problem, the recursive bi-partition using Ncut may suffer from unstable eigenvector problem. Alternatively, the K-way simultaneous partition requires a pre-determined number k as the size of the vocabulary (i.e. the number of subsets), which is actually unavailable as we have little prior knowledge about the event primitives. In fact, a 'best' size of vocabulary can hardly be decided *a priori*. Therefore, we expect an 'automatic' vocabulary discovery scheme, by which the set of word clusters of a proper size is built up without subjective assistance.

To achieve this we use an iterated quadratic programming method. With dissimilarity measure $dis(X_i, X_j)$ between each pair of trajectories, we obtain a similarity value via

$$s_{i,j} = \begin{cases} \exp(-\frac{dis(X_i,X_j)}{\gamma}), & i \neq j \\ 0, & i = j \end{cases} \tag{6.11}$$

and define the similarity matrix

$$S = (s_{i,j})_{1 \leqslant i \leqslant N, 1 \leqslant j \leqslant N}. \tag{6.12}$$

Now the quadratic optimization problem w.r.t. $\mathbf{x} = (x_1, x_2, \cdots, x_N)^T$ is posed as

$$\begin{aligned} &\max_{\mathbf{x}} \mathbf{x}^T S \mathbf{x} \\ s.t. &\mathbf{x} \geqslant \mathbf{0}, \sum_{i=1}^{N} x_i = 1. \end{aligned} \tag{6.13}$$

It has been proved [200] that a maximum point $\mathbf{x}^*$ for the above programming exactly corresponds to a 'dominant' subset of the training set, and the non-zero items in $\mathbf{x}^*$ corresponds to the elements in the dominant subset. A dominant subset has the property that the intra-set similarity is high but similarity with the out-of-set part is much weaker. Therefore, the dominant subset obtained from the quadratic programming becomes naturally a trajectory 'word'.

Normally we should have more than one word in the vocabulary. Consequently after we find the dominant word by running the optimization, we remove them and re-run the optimization on the remaining samples in order to get the next word. This process is iterated until all training trajectories have been assigned to an event word category. It is clear that in this way we find one word by one pass of the quadratic optimization, and when we stop all instances in the training set have been assigned their word label. This process will neither be affected by the robustness of eigen-decomposition, nor incur an issue of pre-deciding or post-validating the best subset numbers.

Solving the quadratic programming does not require a complicated optimization procedure. Just as [200] suggests, a replicator dynamics iteration as

$$x_i(h+1) = x_i(h) \frac{(S\mathbf{x}(h))_i}{\mathbf{x}(h)^T S \mathbf{x}(h)} \tag{6.14}$$

will converge asymptotically to the maximum point.

6.2.2 From Co-occurrence Function to Ground Plane Motion Pattern

As mentioned, the complete motion trajectories in the ground plane consist of those involving the relevant group motion, as well as those irrelevant ones. We have modeled the generation of these motions in two steps: 1) generating the relevant ones from the co-occurrence function, *i.e.*, $P(D|f, A)$; and 2) generating irrelevant motions $P(T|D)$.

To generate the relevant group motion trajectories, we employ a look-up table approach. For each w in the index of the table, we maintain all exemplar trajectories in the

training set. Then for a specific $f(w, \Omega)$ we randomly pick out $F(w, \Omega)$ trajectories from w-labeled exemplar ones. Then we randomly distribute them over the area specified by Ω to obtain a D.

We assume that the irrelevant motion trajectories are uniformly spread across the whole involved area. For any D generated, we may randomly select a number of irrelevant motions from the set and uniformly distribute them all over in order to reach T. However, eventually we do not need to generate T in the algorithm, as will be explained in (6.18) and Section 6.3. Up to this point we have simulated a complete group motion pattern from a type A.

6.2.3 From Ground Plane to Image Plane: Statistical View Variability

The final step of our observation generating model is the map of the complete motion pattern to image plane. The issue of view-invariance comes up here where view variations usually happen among observed motions. We propose here to achieve view-invariance by explicitly and analytically exploiting the statistical view change, instead of reaching an 'invariant' in a 'brutal-force' manner. The practical spatial restraint of the cameras mounted in the real world, as mentioned above, provides us the feasibility to build a statistical model for possible view variations under consideration.

Under the pinhole camera model, different views correspond to different coordinate transforms between the ground plane and the image plane, which is exactly characterized by a 2-D planar homography. Therefore, modeling the view variation is equivalent to modeling the variation of homographies between image planes and the ground plane, *i.e.*, v's. Analytically, v's are 3×3 non-singular matrices which relate the homogenous coordinates of points in the two planes.

We are now building a statistical model $P(v)$ on the space of 3×3 non-singular matrices $\mathbb{GL}(3)$, *i.e.*, the generalized linear group. As $\mathbb{GL}(3)$ is again a nonlinear manifold, we exploit its intrinsic geometry. To account for possible complex distribution of v, we propose a 'Curved Gaussian Mixture Model' (CGMM) as a parametric distribution on $\mathbb{GL}(3)$. Specifically, probability density function for a K-component CGMM is

$$P(v) = \sum_{k=1}^{K} \pi_k p_C(v; \mu_k, \Sigma_k) \tag{6.15}$$

where π_k is the mixing probability, p_C is a single 'curved' Gaussian distribution defined on

$\mathbb{GL}(3)$, μ_k is the center of each Gaussian, and Σ_k the covariance defined in the *tangent space* at μ_k.

We compute the plane homographies from each of the training activities by locating landmark points of the ground plane in the image. For football videos they are field markers. The CGMM, from a collection of training views $\{v_j\}_{j=1}^{M}$, is then learned as follows. We first cluster v_j's into different components by computing a pairwise intrinsic metric between each pair v_1, v_2

$$d(v_1, v_2) = ||\log(v_1^{-1} v_2)|| \tag{6.16}$$

which is introduced in standard literature on Lie groups like [167]. From the pairwise similarity metric we may employ any suitable unsupervised clustering technique to cluster v_j's. Here we make use of the repeated quadratic programming in [200]. Once we have obtained K clusters (components), each of which contains M_k samples, we estimate the mixing probabilities as

$$\pi_k = \frac{M_k}{M}, k = 1, 2, \cdots, K. \tag{6.17}$$

Then the center of each component is estimated from the samples clustered into that component, using exactly the iterations between exponential map and logarithmic map as in (6.4)(6.5), but replacing the maps with those for Lie groups (see appendix). Finally, the covariance Σ_k is calculated as normally done in the tangent space at μ_k, which contains the logarithmically mapped component samples from $\mathbb{GL}(3)$.

To simulate a new view from the learned CGMM, we randomly select a component according to the mixing probability, locate the center, generate a Gaussian random matrix with the covariance in the tangent plane, and finally exponentially map it to $\mathbb{GL}(3)$. One possible issue is that the simulated homography may not be a practically valid one. However, the probabilistic scheme, to be presented in Section 6.3, will assign low likelihood to invalid views so that the simulation algorithm will work properly.

The last component involved in the observation generating model, $P(O|T, v)$, is trivial once a view transform v is given: The observation O will be determined by $O = v(T)$ or $T = v^{-1}(O)$. Therefore, $P(O|T, v) = \delta(O = v(T))$ or $P(T|O, v) = \delta(T = v^{-1}(O))$. Note that factors as 'observation noise' or 'disturbance' have been implicitly taken into account when modeling $P(D|f, A)$ and $P(T|D)$. Also note that the form of Kronecker delta for

$P(T|O,v)$ reduces (6.1) into

$$P(O|A) = \int_{f,D,v} P(v^{-1}(O)|D)P(v)P(D|f,A) \\ P(f|A)dfdDdv. \tag{6.18}$$

6.3 Recognition and Segmentation

With all components of the generating model ready in hand, it is straightforward now to employ Monte Carlo method to evaluate the posterior probability $P(A|O)$. Specifically, we pick up a motion class A_i according to class prior $P(A_i)$ (assumed known), simulate a co-occurrence function f_{ij} from $P(f|A_i)$, generate a relevant motion pattern D_{ij} according to $P(D|f_{ij}, A_i)$, and randomly select a view v_j from $P(v)$. Then the posterior probability can be approximated as

$$P(A_i|O) \doteq \frac{P(A_i)\sum_j P(v_j^{-1}(O)|D_{ij})}{\sum_i P(A_i)(\sum_j P(v_j^{-1}(O)|D_{ij}))} \tag{6.19}$$

by Monte Carlo principle. It is worth noting that view invariance has been realized implicitly by integrating all possible views.

Now it remains to evaluate $P(v^{-1}(O)|D)$. Based on the motion generating scheme discussed in 6.2, we achieve this as follows. We look for a one-to-one correspondence of the m trajectories $t_{D1}, t_{D2}, \cdots, t_{Dm}$ in D with m ones in $v^{-1}(O)$. In other words, we pick up m trajectories $t_1, t_2, \cdots, t_m$ from the n ones in $v^{-1}(O)$, such that the total distance between the two group $\sum_{i=1}^{m} d(t_{Di}, t_i)$ is minimized. Intuitively, we are finding a subset of $v^{-1}(O)$ such that motions in it are the most likely ones corresponding to the relevant activity. The distance between two trajectories $d(t_{Di}, t_i)$ is simply calculated by summation of the Euclidean distances of point pairs between the two trajectories. Then with all pair-wise distances between $v^{-1}(O)$ and D available, the desirable correspondence can be determined by running the classical Kuhn-Munkres assignment algorithm [148]. Once the best correspondence is found on $t_1^*, t_2^*, \cdots, t_m^*$, $P(v^{-1}(O)|D)$ is evaluated as

$$P(v^{-1}(O)|D) \triangleq \frac{1}{N}\exp(-\frac{\sum_{i=1}^{m} d^2(t_{Di}, t_i^*)}{2\alpha^2}). \tag{6.20}$$

After a Maximum A Posterior classification for the observed motion O into activity type A^*, we are in a position to segment the trajectories related to the relevant activity. This is now straightforwardly achieved by choosing the most likely correspondence between the generated D's from A^* and $v^{-1}(O)$.

6.4 Experiment

We perform experiments again on GaTech Football Play Dataset, employing full ground-truth trajectories from training to testing. As in a practical system the trajectories obtained from tracking will be noisy, we also propose possible modifications towards critical steps in the framework to handle non-robust input data.

6.4.1 Experiments with Ground-truth Annotation

We pre-process activity samples with varying temporal duration by normalizing their time scales to fixed value, with trajectory interpolation and temporal resampling done if necessary. The 56 play samples are organized into three play types, including *Dropback, Middle&Right Run* and *Wideleft Run*. The play type division takes into account both the play type hierarchy and the balance of sample amount. The learning and then recognition runs a multiple of times independently, each of which uses a random division of sample collection into training (approximately 80%) and testing (approximately 20%) sets. The homographies corresponding to the view changes are determined by locating the landmark points on the football field. The free parameter α^2 and γ is simply taken as the variation of all pairwise distances between trajectories and the normalizing factor N cancels out when calculating (6.19). Since the amounts of the training samples are limited, we augment the size of training set during each training process as follows. We first learn a trajectory vocabulary with the trajectories in the original training set. Then we perturb each original trajectory to get new ones and search for the new 'word' label for the new trajectories. The perturbation is realized by adding 2-D isotropic Gaussian variables on ground-plane coordinates at particular time instants (0%,20%,40%,60%,80%,100% of the video duration) and polynomially interpolating the other time instants. Moreover, we shift the location of each trajectory (original and generated) entirely with another Gaussian. In this way, for each of the original training play we get 20 new ones, and thus the eventual size of training set is 20 times more than the original one.

For each testing play, we generate 5×10^4 Mote Carlo samples when evaluating (6.19) as well as assume a uniform prior class probability. The confusion matrix for recognition is shown in Table 6.1, indicating the percentage by which a specific play type is recognized as itself/another. An average recognition rate of approximately 80% is observed from the

Table 6.1: The confusion matrix of play recognition: D,M, and W stand for Dropback, Middle&right Run, and Wideleft Run respectively.

	D	M	W
D	69.1	24.4	6.5
M	2.1	86.3	11.6
W	6.7	6.2	87.1

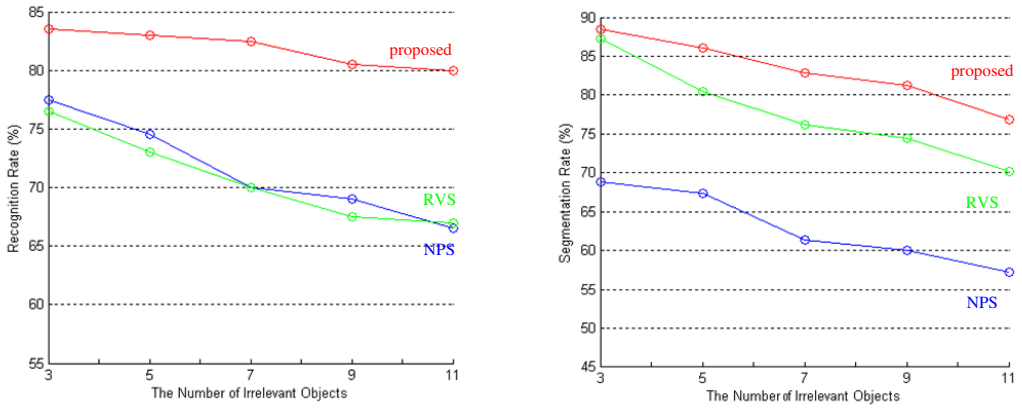
Table 6.2: The segmentation rate for each type of play.

Dropback	Middle&right Run	Wideleft Run
75.8	76.5	78.2

confusion matrix. The average segmentation rate is given in Table 6.2, where the percentages denote how many of the offensive players are correctly identified. We achieve more than 75% correct identification as a whole.

To evaluate the effectiveness of statistical view change modeling and optimal assignment based segmentation, we design two baselines, which comes more intuitively than the proposed probabilistic framework, for a comparative study. The first, namely 'random view selection' (RVS), does not simulate a homography from the learned CGMM, but randomly picks out one from all available training ones. The second baseline, denoted as 'nearest player selection' (NPS), picks out the spatially closest player in the testing play to match with each of the simulated relevant players, instead of performing the Kuhn-Munkres assignment. At the same time, to investigate the performances with varying amount of irrelevant objects, we randomly drop defensive players in the testing plays and re-run the recognition and segmentation. The play recognition rate and segmentation rate vs. the number of irrelevant players for RVS, NPS and proposed mechanism, are plotted in Figure 6.3 (a)(b) respectively. Several segmentation results with full eleven irrelevant objects are shown in Figure 6.4 for baselines and proposed framework. For better visualization all are shown on the ground plane with field markers removed. Among them the proposed full mechanism outperforms the other two in identifying the relevant motions.

The weakness of RVS approach comes mainly from a lack of generative capability.



(a) Recognition rate comparison.

(b) Segmentation rate comparison.

Figure 6.3: Performance comparisons of the proposed complete framework and two baselines.

When a testing view coincides with one of the training ones, RVS may yield comparative performance with the full scheme. However, CGMM modeling will allow more flexible view changes other than exemplars in the training set, thus giving rise to improved performance. NPS, on the other hand, greedily assigns each relevant motion in a simulated activity to the nearest one in the testing activity, probably leading to assignment of multiple relevant motions onto a single testing one. The trajectory correspondence scheme used in the proposed framework guarantees a one-to-one correspondence and thus achieves better segmentation.

Monte Carlo synthesis employed in our work is not computationally economical. However, an exhaustive approach, rather than probabilistic modeling, turns out to be much more prohibitive. For an activity involving n objects of which m are relevant ones, by exhaustive examination we have C_n^m possible segmentations. Then if we compare each of these candidates with all of the t training samples, each of which may undergo s possible view changes, the total amount of comparisons will be tsC_n^m. In our experimental setting, $m = 11, n = 12, t = s = 0.8 \times 56$ will lead to more than 10^9 comparisons for a single testing activity, which is in sharp contrast to 5×10^4 generated samples in our experiment.

6.4.2 Considerations to Handle Computed Tracks

In football plays, strong occlusions among players are more common, and motion of a single player may switch frequently between stationary and moving. As a result, the track

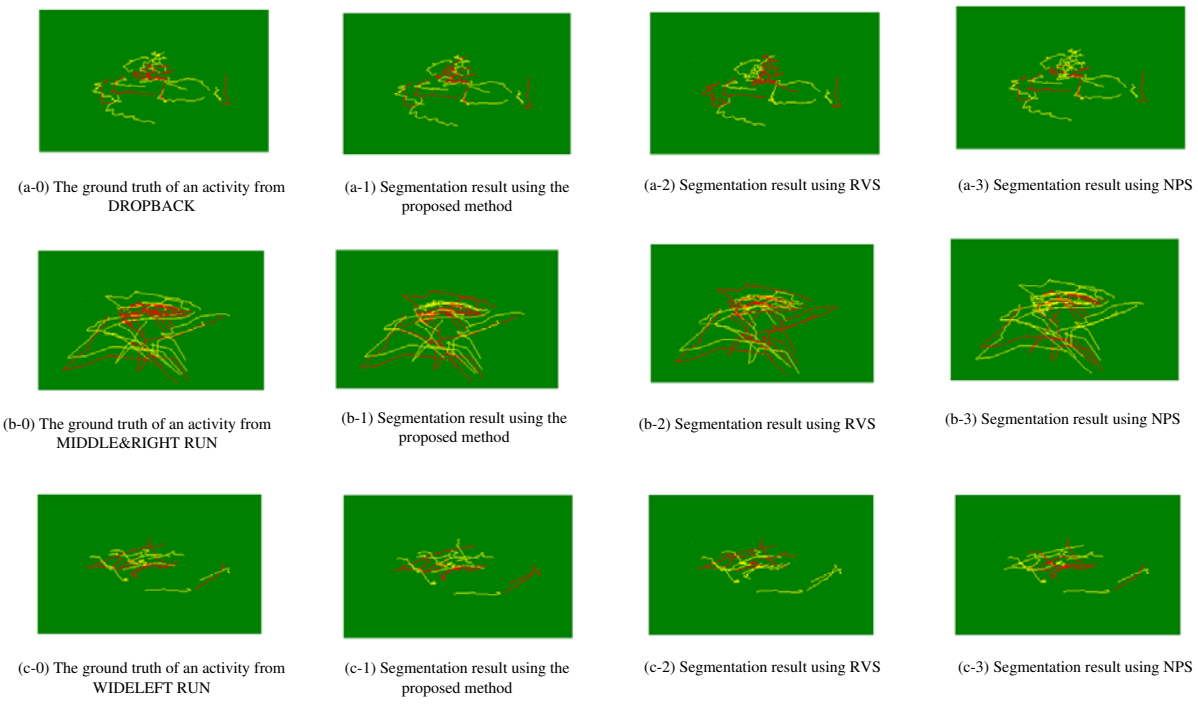


Figure 6.4: Activity examples and comparison of segmentation results. Red trajectories denote those of relevant activity (offensive side) and Yellow denote irrelevant ones (defensive side).

for a single object is typically significantly fragmented, and the effective number of objects (tracks) may be less than expected (twenty-two) at times and may change from time to time. We modify the scheme by which we evaluate $P(v^{-1}(O)|D)$ as follows to address the above issues.

First we identify continuous temporal durations, during each of which the tracker produces a constant number (greater than m) of non-fragmented tracks. We denote one of these duration as T_c and assume N_c tracks $t_{c,1}, t_{c,2}, \cdots, t_{c,N_c}$ are computed from the tracker during this period. Then from the Dm simulated trajectories we randomly select $n_c = \lceil \frac{Dm}{2} + 1 \rceil, \cdots, Dm$ trajectories $t_{D1}, t_{D2}, \cdots, t_{Dn_c}$ and run Kuhn-Munkres assignment between $\{t_{c,1}, t_{c,2}, \cdots, t_{c,N_c}\}$ and $\{t_{D1}, t_{D2}, \cdots, t_{Dn_c}\}$. The best assignment yielding the minimum average cost per trajectory-pair is recorded as $\{t^*_{c,1}, t^*_{c,2}, \cdots, t^*_{c,n^*_c}\}$ and $\{t^*_{D1}, t^*_{D2}, \cdots, t^*_{Dn^*_c}\}$.

For every duration T_c we will find a best assignment and the final evaluation of $P(v^{-1}(O)|D)$ is realized by

$$P(v^{-1}(O)|D) \triangleq \frac{1}{N} \exp(-\frac{\beta \sum_{\{T_c\}} (\frac{m}{n^*_c} \sum_{i=1}^{n^*_c} d^2(t^*_{c,i}, t^*_{Di}))}{2\alpha^2}). \tag{6.21}$$

Here the normalizing factor

$$\beta = \frac{l}{\sum_{\{T_c\}} l(T_c)} \tag{6.22}$$

where l is the length of the play and $l(T_c)$ is the length of the duration T_c.

6.5 Extension: Activity Characteristic Curve for Classification

In this section we deviate from the generative model for a while and show that the co-occurrence function, as a discriminative feature itself, can be used for group motion recognition as well by learning a 'Activity Characteristic Curve' from them.

6.5.1 Learning the Activity Characteristic Curve on Co-occurrence Manifold

A single spatial co-occurrence function is a 'holistic' but static description of a trajectory ensemble, not taking temporal evolution or time constraint into account. Similar descriptions have been invented for object recognition purpose in a still image [201, 202]. However, if we observe the group motion at different time-instants, each across a short time span, the coordinated group motion is essentially a dynamic ensemble of trajectory

segments. *i.e.*, at time t we will have a distinct $f(w, \Omega; t)$. This temporal process also critically determines the specific group motion pattern. Therefore, time-series modeling is also a natural attempt toward a group motion characterization. The time sequence of co-occurrence functions, nevertheless, is again not in Euclidean space, where we have rich and powerful tools on hand. Instead, the motion is a time sequence on the co-occurrence manifold. Evolution processes or time series on a Riemannian manifold, have received little attention. Among the limited efforts [203] is a systematic treatment for the manifold of directional data.

Let us denote $f(w, \Omega; t)$ by $f(t)$ from now on for simplicity, and keep in mind that as t varies $f(t)$ is an evolving process. However, beyond this we can hardly make any stronger assumptions. In this case, to find a proper quantitative feature for each group activity pattern we are going to use *activity characteristic curve* defined as

$$C(t) = \mathbb{E}(f(t)) \tag{6.23}$$

which is nothing but the mean value curve for the evolution process $f(t)$. With multiple training sequences (i.e. realizations of the process) for the same type of group activity, we are always able to find the mean sequence without further assumptions, though the mean should be obtained from an average on the manifold rather than in Euclidean space. It can be expected that different activity characteristic curve corresponding to different activities will be located distinctively on the co-occurrence manifold, therefore providing us the capability to classify a new sequence. In Figure 4.5 we visualize the above concepts.

The activity characteristic curve $C(t)$, i.e. the expectation curve on the manifold is explicitly defined as

$$C(t) = \arg\min_{f} \mathbb{E}(d^2(f, f(t))) \tag{6.24}$$

according to [152], where d is the intrinsic distance on the manifold induced by the Riemannian metric rather than the usual Euclidean distance $||f - f(t)||$. If there exists a probability density function $p(f(t))$ for $f(t)$, then we have

$$C(t) = \arg\min_{f} \int_{\mathscr{M}} d^2(f, f(t)) p(f(t)) d\mathscr{M} \tag{6.25}$$

where $d\mathscr{M}$ can be thought as a 'patch' of the manifold. Also, note that the integration is performed on the manifold only. Since we do not make more assumptions about p, practically we estimate the density function by kernel method once provided with a set of

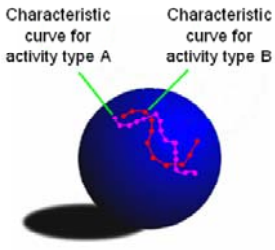


Figure 6.5: Activity Characteristic Curve on the co-occurrence manifold. The sphere represents the manifold and each solid dot represents an average co-occurrence function at a time instant. The Activity Characteristic Curve, therefore, is the temporally ordered dot sequence.

training samples $\{(f_i(t), t_i)\}_i$

$$p(f(t)) = \frac{\sum_i \mathbb{K}(d^2(f(t), f_i), t - t_i)}{k(t)} \tag{6.26}$$

where $k(t)$ is the normalizing factor. For simplicity we make use of the Nadaraya-Watson kernel[204, 205] so that $\mathbb{K}$ can be separated into a temporal factor and a spatial factor

$$p(f(t)) = \frac{\sum_i \mathbb{K}_{H_s}(d^2(f(t), f_i))\mathbb{K}_{H_t}(t - t_i)}{k(t)} \tag{6.27}$$

where H_s and H_t are the spatial and temporal 'bandwidths' respectively.

Performing above minimization to get $C(t)$ is equivalent to minimizing a 'variance' defined on the manifold. Indeed, the quantity $\mathbb{E}(d^2(f, f(t)))$ should be interpreted as a variance measure of $f(t)$ with respect to f. It has been proved [152] that the gradient of $\mathbb{E}(d^2(f, f(t)))$ with respect to f is

$$\nabla_f \mathbb{E}(d^2(f, f(t))) = -2\mathbb{E}(\mathscr{L}_f(f(t))) \tag{6.28}$$

Consequently an iterative gradient descent algorithm can be derived to conduct the minimization and eventually find the mean curve which again involves logarithmic and exponential maps as

$$\bar{f'}^{(h)} = \mathbb{E}(\mathscr{L}_{f^{(h)}}(f(t))) \tag{6.29}$$

and

$$f^{(h+1)} = \mathscr{E}_{f^{(h)}}(\bar{f'}^{(h)}). \tag{6.30}$$

Note the expectation here is performed in the tangent space $\mathscr{T}_{f^{(h)}}$, which is a vector space. Therefore, an ordinary expectation can be employed in this step.

The logarithmic and exponential maps are straightforward from the discussion of previous sections. The expectation in the tangent place, however, deserves a further investigation, since we only have finite training samples. Generally, in the Euclidean space the finite-sample expectation can be obtained from a weighted sum of all the available samples. In our case, it is

$$\bar{f'}^{(h)} = \frac{\sum_i w_i \mathscr{L}_{f^{(h)}}(f_i)}{\sum_i w_i} \tag{6.31}$$

Here we propose the following weighting strategy

$$w_i = \sum_j \mathbb{K}_{H_s}(d^2(f_j, f_i))\mathbb{K}_{H_t}(t - t_j) \tag{6.32}$$

By employing this weight involving both samples at close time instant and spatially neighboring samples, we not only take the temporal correlation into account, but also get a trade-off between the mean and the mode of the samples. In other words, we achieve both flexibility and robustness toward variation of the data.

If we keep a temporal bandwidth H_t but let the spatial bandwidth H_s go to zero, the weighting pattern reduces to a standard Nadaraya-Watson regressor [204, 205] as well as a manifold version[206]. On the other hand, a small temporal bandwidth and a large spatial bandwidth is equivalent to assuming that all samples at the same time are independently drawn from $p(f(t))$ and samples from different time instants are mutually independent. Thus our weighting pattern is a more general and flexible version which may be modified to adapt to various practical scenario.

6.5.2 NACC Classifier for New Group Motion

With c types of group activities represented as $C_1(t), C_2(t), \cdots, C_c(t)$, we are in a position to categorize a new incoming activity into one of these classes. The classification is achieved in two steps.

In the first step, we identify each of the trajectories T in the incoming video as one of the words. With W words $w = 1, 2, \cdots, W$ learned in training phase, for each of which word we have m_w primitives $X_1^w, X_2^w, \cdots, X_{m_w}^w$ as its instances in the training set, we evaluate the similarity between T and each word category $\{X_{i_w}^w\}$, and label the segment as the word with which it shares the largest similarity. Note that the trajectory vocabulary is

based on pairwise similarity of all training samples, and the clustering is realized recursively by finding the dominant clique of the similarity graph. Consequently, we do not hold a parametric or probabilistic model for each word category, and therefore a parametric or probabilistic classifier is not straightforward. We start from pairwise similarity again with the dissimilarities $dis(X_1^1, T), dis(X_2^1, T), \cdots, dis(X_{i_w}^w, T), \cdots, dis(X_{m_w}^W, T)$ evaluated. Then we simply apply a Maximum-Similarity classifier as following

$$Word(T) = \arg\min_{w} \min_{i_w} dis(X_{i_w}^w, T) \tag{6.33}$$

This is equivalent to the widely-used Nearest-Neighbor classifier. Though computationally intensive, Maximum-Similarity classifier is probably the most feasible for our task.

In the second step we construct the evolving co-occurrence function sequence and classify it into one of the activity types. The sequence of co-occurrence functions for a testing activity is denoted as $D(t)$. The Nearest Activity Characteristic Curve (NACC) classifier is nothing but the decision rule

$$Activity(D(t)) = \arg\min_{j} \sum_{t} d(D(t), C_j(t))$$

The classifier looks for the activity type of minimum manifold distance with the testing activity. Also, the classifier can be interpreted as a correlator, which picks up the maximum manifold correlation as the recognition result. In the language of signal processing, it can also be viewed as performing matched filtering on a nonlinear manifold.

6.5.3 Experiment

The learning and recognition framework described above has been implemented on GaTech Football Play Dataset once again. Similarly from all play samples we use five play types, including *Combo Dropback, HITCH Dropback, Middle Run, Wideleft Run* and *Wideright Run*, including a total of 56 play prototype samples. Learning and then classification runs a multiple of times independently, each of which uses a random division of sample collection into training (80%) and testing (20%) sets. For those prototypes falling into training set, we also generate new training samples by slightly disturbing the prototype trajectories both locally and globally using i.i.d. Gaussian. For the same play with different formations, a separate activity characteristic curve is learned for each formation. The spatial kernels include 16 overlapping blocks covering the activity area. The average

Table 6.3: Confusion matrix of play recognition on groundtruth data: H,C,M,L, and R stand for HITCH Dropback, Combo Dropback, Middle Run, Wideleft Run, and Wideright Run respectively.

	C	H	M	L	R
C	86.1	3.5	8.2	1.7	0.5
H	0.7	91.9	4.4	1.4	1.6
M	13.4	7.2	66.2	11.8	1.4
L	0.6	1.2	0.6	96.9	0.7
R	0.1	4.2	2.4	1.8	91.5

confusion matrix is shown in Table 6.3, indicating the percentage by which a specific play type is recognized as itself/another.

An average correctness ratio of 86% is observed from the confusion matrix. The fully quantitative comparison with previous work, especially [6], is still difficult due to a completely different framework, unavailability of implementation details, as well as different datasets being used. However, qualitatively it is seen from Figure 13 in [6] that we achieve a recognition performance better than [6]. Note that the previous work used parametric Bayesian network modeling with explicit domain knowledge about football game incorporated. In contrast, the event ensemble model works 'autonomously' and is directly extendable to more general coordinated group motions.

The main computational load comes from generating the pairwise similarity matrix together with new event categorization, which compares the new incoming activity with all event words available. The vocabulary discovery and activity characteristic curve learning converge both quickly in several iterations.

6.6 Discussion

In this chapter we investigate the problem of simultaneous recognition and segmentation of collaborative group motion pattern. Under a Bayesian Maximum A Posterior formulation we achieve our purpose in an 'analysis-by-synthesis' fashion. We explicitly

model every component of the motion generating process, during which we exploit the non-Euclidean geometry of the co-occurrence manifold in a statistical manner. In particular, we set up probabilistic CGMM to characterize view variation and achieve view-invariance by statistical marginalization. We demonstrate the effectiveness of the proposed framework using football plays as experimental data. However, note that we make little use of football-specific domain knowledge and the framework is expected to be extendable to other types of group activities.

Following this line, the following open issues may receive further investigation. Besides object segmentation, temporal detection of a particular group motion pattern, especially with a changing number of involved objects, is also of interest. It also deserves to model two or more groups of motions and interactions among groups (*e.g.*, taking defensive side into account as well). We may also consider incorporating articulated motion features, besides simply point motion paths, to establish a 'panoramic' characterization for a group motion, which may hopefully bring benefit for more accurate recognition. Finally, it will be useful to look into group motions in broader domains, beyond football plays, to be unified under this or another updated framework.

6.7 Appendix

We list the geometries of the co-occurrence function manifold and the matrix Lie group. Under the defined Riemannian metric, the co-occurrence function manifold in fact shares the same geometry as DTIM as discussed in Chapter 3, and the matrix Lie group has been discussed in Chapter 5 as well. We reiterate these facts here for clarity and the readers' convenience.

6.7.1 Geometry of the Co-occurrence Function Manifold

The space of all spatial co-occurrence functions $\mathscr{F}$ is a Riemannian manifold with the Riemannian metric defined as

$$< f_1', f_2' > \triangleq \sum_{w \in \mathscr{W}} \sum_{\Omega \in \Pi} f_1'(w, \Omega) f_2'(w, \Omega) \tag{6.34}$$

where f_1', f_2' are the elements in the tangent spaces $\mathscr{T}_f$ at f. With the above defined Riemannian metric, the intrinsic distance between two co-occurrence functions f_1, f_2 is

given by (4.3). The geodesic, *i.e.* the curve of minimum length connecting two co-occurrence functions f_1, f_2 , is given by

$$G(\lambda) = \frac{(1-\lambda)f_1 + \lambda f_2}{\lambda^2 + (1-\lambda)^2 + 2\lambda(1-\lambda) < f_1, f_2 >} \tag{6.35}$$

where λ is the real parameter between 0 and 1. The exponential map $\mathscr{E}_{f_m} : \mathscr{T}_{f_m} \to \mathscr{F}$ for $f' \in \mathscr{T}_{f_m}$ is defined as

$$\mathscr{E}_{f_m}(f') = \cos(< f', f' >^{\frac{1}{2}}) f_m + \frac{\sin(< f', f' >^{\frac{1}{2}})}{< f', f' >^{\frac{1}{2}}} f' \tag{6.36}$$

The logarithmic map $\mathscr{L}_F : \mathscr{F} \to \mathscr{T}_{f_m}$, which is actually the inverse map of exponential map, is then given by

$$\mathscr{L}_{f_m}(f) = \frac{\arccos(< f, f_m >)}{< f^*, f^* >^{\frac{1}{2}}} f^* \tag{6.37}$$

where

$$f^* = f_m - < f, f_m > f. \tag{6.38}$$

6.7.2 Geometry of Matrix Lie Group

The space $\mathbb{GL}(3)$ is a matrix Lie group and thus a Riemannian manifold, with the intrinsic distances between two elements defined as (9). The exponential map $\mathscr{E}_{v_m} : \mathscr{T}_{v_m} \to \mathbb{GL}(3)$ for $v' \in \mathscr{T}_{v_m}$ is given by

$$\mathscr{E}_{v_m}(v') = v_m \exp(v_m^{-1} v'). \tag{6.39}$$

The logarithmic map $\mathscr{L}_{v_m} : \mathbb{GL}(3) \to \mathscr{T}_{v_m}$, meanwhile, is

$$\mathscr{L}_{v_m}(v) = v_m \log(v_m^{-1} v). \tag{6.40}$$

The matrix exponential and logarithmic operation used here are defined as

$$\exp(X) = \sum_{i=0}^{\infty} \frac{1}{i!} X^i \tag{6.41}$$

and

$$\log(X) = \sum_{i=1}^{\infty} \frac{(-1)^{i-1}}{i} (X - I)^i. \tag{6.42}$$

For more discussion about the geometry of matrix Lie group the reader is referred to [167].

Chapter 7

Directions for Future Work

We have looked into collaborative group activities represented as an ensemble of motions, with the goal of inferring 1) the individuals relevant to a collaborative pattern of interest (Segmentation); 2) the underlying group activity pattern (Recognition); 3) how two motion ensembles are similar and how we can 'optimally' transform one to the other (Matching). Throughout the work we took football plays as a test case with the corresponding problems 1) who are offensive players; 2) what are the offensive strategy they are using; 3) whether two plays are using the same strategy and how we can remove the spatio-temporal misalignment between them due to imaging conditions. We exploit the geometry of the space of the involving features/descriptors/quantities and develop statistical models/tools on these non-linear manifolds. These efforts motivate us both from an applied point of view and a mathematical point of view to continue with a sequence of new interesting problems, which will form the future directions of the research.

7.1 Other Research Topics on Complex Motion/Activities

Following the line of extracting meaningful temporal information to process multi-object motion patterns, the problems discussed below would be also interesting for further study. Firstly, the temporal interaction matrix relies on correctly obtaining point trajectories, which brings the need for a more robust descriptor from incomplete/errorneous trajectories, since the latter will be the most probable outcome of a tracking module of a complete vision system. Though the temporal interaction matrices turn out to be view-stable, a strict view-invariant descriptor, beyond an empirically view-stable one, remains a challenge just as for other vision applications. As has been mentioned, detection and segmentation of a particular group activity pattern, especially with a changing number of involved objects, is also of interest. This requires us to locate the starting and ending positions on the temporal axis as precisely as possible, and to provide an adaptive scheme by which the system can smoothly alter the current status when participating objects changes. Although the interactions among objects are currently captured in terms of numerical quantities in arrays/matrices,

one can introduce syntactic approaches. The topological or geometrical characterization of the space of 'syntactic matrices', if exists, will probably brings new insight to describe the ensemble motion patterns.

On the other hand, the probabilistic generative model proposed for group activities naturally brings up the following open issues. The defensive team in a football play, as has been pointed out, actually follows another group motion pattern distinct from the offensive one, instead of random and irrelevant movements. This convinces us the necessity to model two or more groups of activities and interactions between groups, in addition to the interactions between objects within a single group. In this case, group segmentation becomes even more challenging due to the increased ambiguity of a local motion feature belonging to multiple potential global patterns. We may also consider incorporating articulated motion features of a single object/human being, besides simply point motion paths, to establish a 'panoramic' characterization for a group activity, which may hopefully contribute to more accurate recognition. With a standard Bayesian network formulation and continuous advance in research on features/models, traditional methods may achieve improved performance than early work reported in [6].

It will also be interesting to look into activities from other domains other than football games: in surveillance scenario, the crowd motion and flow, for example, may also need a concise descriptor for certain types of analysis. In indoor environments, interactive behaviors include meetings and class sessions, where an expressive visual feature is the face of every participant, together with the expressions and pose changes represented by local facial feature tracking. How can we utilize facial expression, pose, and other dynamics to assist analyzing the interactions among multiple objects? Classical and new tasks, including temporal segmentation, automatic indexing, participant grouping, discriminant descriptor extraction, as well as probabilistic network to model the mutual influence, will arise again but under a different background of multiple interactive faces.

7.1.1 Example: Initial Considerations on Indoor Interactive Behaviors of Multiple Faces

As an example of other domains involving collaborative behaviors, we may consider an indoor environment where participants mainly remain still with sporadic but gentle head motion. A typical scenario is a class session, where the students are seated with limited

(a) (b) (c)

Figure 7.1: Illustration of Low-level processing for a classroom multi-face interactive motion: (a) face detection and tracking; (b) KLT tracking; (c) face pose estimation.

face pose changes. Is it possible to infer meaningful group motion patterns from the motion features at this level, and are there any state-of-the-art learning tools for analyzing these patterns? We may apply the Viola-Jones face detector [207] to initially locate the faces, and then apply a tracker such as in [208]. Within the boxes we apply Kanade-Lucas-Tomasi feature tracker [177] to capture the face pose changes, and fit a global affine motion, denoted by a single motion vector, to describe the overall orientation of the faces with respect to the camera. An illustration is given in Figure 7.1.

Visual Interaction from Pose, Embedding, and Participant Grouping As a first attempt, we may consider the vector $\overrightarrow{u}_{p\to q}$ pointing from person p to person q and the accumulated pose direction $\overrightarrow{v}_q$ of person q (which is computed from the affine parameters), we may get 'visual' influence of person q receiving from person p as

$$\sigma_{p\to q} = \frac{\exp(-\sigma_1 \|\overrightarrow{u}_{p\to q}\|)\exp(-\sigma_2\langle \overrightarrow{u}_{p\to q}, \overrightarrow{v}_q\rangle)}{\sum_{p'}\exp(-\sigma_1 \|\overrightarrow{u}_{p'\to q}\|)\exp(-\sigma_2\langle \overrightarrow{u}_{p'\to q}, \overrightarrow{v}_q\rangle)} \tag{7.1}$$

where the notation $\langle\cdot,\cdot\rangle$ means the angle between the two vectors. Basically we believe that when person q is looking at person p then q is receiving a strong influence from p, and vice versa.

Given $\Sigma = (\sigma_{p\to q})_{P\times P}$ computed from the previous step, we find the 1-D embedding $x_p, p = 1, 2, \cdots, P$ of the participants by minimizing $\sum_p \omega_p \sum_q \sigma_{p\to q}(x_p - x_q)^2$ where $\omega_p = \frac{\sigma_{p\to q}}{\sum_q \sigma_{p\to q}}$. This number implies the 'strength' of influence that person p gives to all other people. Intuitively, we want the pairs interactive with each other to stay close in the embedded space, and the individual influential to others to be close to the influenced people.

The embedding problem is essentially

$$\min_x x^T L x \tag{7.2}$$

where

$$L = \Omega - \frac{\Omega\Sigma + \Sigma^T\Omega}{2} \tag{7.3}$$

and $\Omega = diag(\omega_1, \omega_2, \cdots, \omega_P)$. Proper regularization should be imposed such as $x^T\Omega x = 1$. Finally, a hard partition of the real axis gives the grouping of the participants.

Joint Temporal Segmentation and Indexing of Coherent Activities The next task for consideration is unsupervised discovery of meaningful activities. Given a descriptor matrix $Y_{M\times N} = [Y_{m,n}]$, each column of which is a $m \times 1$ descriptor (static 1st or 2nd order one) for all people or a subgroup involved in an interaction in frame n, we optimize the following objective

$$\min_{F,G} \|Y - FG\|_2 + \alpha\|GS\|_0 + \beta\|G\mathbf{1}\|_0 \tag{7.4}$$

$$s.t. F \in \mathbb{R}^{M\times K}, \tag{7.5}$$

$$G \in \{0,1\}^{K\times N}, \tag{7.6}$$

$$\mathbf{1}^T G = \mathbf{1}, \tag{7.7}$$

$$S_{N\times(N-1)} = \begin{pmatrix} -1 & 0 & 0 & 0 & \cdots \\ 1 & -1 & 0 & 0 & \cdots \\ 0 & 1 & -1 & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}. \tag{7.8}$$

We are interested in G, whose nth column is a sparse code for time n, and the fact of $G(k,n) = 1$ means that at time n the group motion is of type k. $\|GS\|_0$ prevents frequent changes between different types and $\|G\mathbf{1}\|_0$ leads to limited activity types.

The objective above can be relaxed as

$$\min_{F,G} \|Y - FG\|_2 + \|G[R; \gamma I_{N\times N}]\|_1 \tag{7.9}$$

where

$$R_{N\times N} = \begin{pmatrix} -\alpha & 0 & 0 & \cdots & \beta \\ \alpha & -\alpha & 0 & \cdots & \beta \\ 0 & \alpha & -\alpha & \cdots & \beta \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \beta \end{pmatrix}. \tag{7.10}$$

is non-singular. Denote $A = [R; \gamma I_{N\times N}]$, and then we aim to solve

$$\min_{F,G} \|Y - FG\|_2 + \|GA\|_1 \tag{7.11}$$

Consequently, the optimization can be iterations of the following steps: (1) Solve $F^{(J)} = \arg\min_F \|Y - FG^{(J-1)}\|_2$ using a proper dictionary learning method; (2) Solve $G^{(J)} = \arg\min_G \|Y - F^{(J)}G\|_2 + \|GA\|_1$. To do this, we solve the following equivalent optimization problem

$$\min_{G,T} \|Y - F^{(J)}G\|_2 + tr(\mathbf{1}^T T) \tag{7.12}$$

$$s.t. -T \leq GA \leq T. \tag{7.13}$$

Learning Group Activity Ranking Function and Pairwise Interactive Structure From the previous step we obtained group activity indices $y \in \{1, 2, \cdots, Y\}$ and the temporal durations for each of these segments by checking the dominant elements in G. Using them we may extract a dynamic descriptor per person per segment $d_p \in \{1, 2, \cdots, D\}, p \in \{1, 2, \cdots, P\}$, where P is the total number of people considered. The dynamic descriptor consists of two parts: the average C_p and the motion A_p (i.e. $d_p = (C_p, A_p)$. The VQ of the dynamic descriptor into D words can be realized in an appropriate manner (e.g., K-means). Consequently, for the ordered person pair (p, q) we may obtain a dynamic influence descriptor $d_{p\to q} = (C_{p\to q}, A_{p\to q}) \in \{1, 2, \cdots, G\}$, in which the two components can be described as the 'transformations' from d_p to d_q.

We consider pairwise interactions between involving people and quantitatively characterize them into a weighted directed graph $Z = (z_{p\to q})_{P\times P}$, which is the 'influence' of person p toward person q. Our goal is to learn an activity-specific parameter vector W_y for each activity, where $W_y = (W_{Ind,y}^T, W_{Pair,y}^T)^T$. Specifically, $W_{Ind,y}$ is a $D \times 1$ vector and $W_{Pair,y}$ is a $G \times 1$ vector corresponding to d_p an $d_{p\to q}$ respectively.

Suppose we get N indexed segments among which the nth belongs to type $y(n)$ with corresponding descriptor $d_p(n), d_{p\to q}(n)$. To find Z and W, we optimize the following discriminative objective

$$\max_{Z,W_y} \sum_y \Phi_y(W_y, Z, d(1), d(2), \cdots, d(N)) \tag{7.14}$$

where $\Phi_y(W_y, Z, \cdot) = \phi(W_y, Z, \cdot) - \phi_y(W_y, Z, \cdot) - \phi_{\setminus y}(W_y, Z, \cdot)$. More specifically, $\phi(W_y, Z, \cdot) = \sum_n (\varphi(W_y, Z(n), d(n)) - \overline{\varphi(W_y, Z(n), d(n))})^2 - \gamma tr((Z(n) - \Sigma(n))^T (Z(n) - \Sigma(n)))$ and $\varphi(W_y, Z(n), d(n)) =$

$W_{Ind,y}^T E(n)\mathbf{1} + W_{Pair,y}^T F(n) diag(vec(Z(n)))\mathbf{1}$, where $diag(vec(Z(n)))$ is a $P^2 \times P^2$ diagonal matrix whose diagonal elements are the elements of $Z(n)$,$E(n) = (\delta(d_p(n) = d))_{D\times P}$, $F(n) = (\delta(d_{p\to q}(n) = g))_{G\times(P\times P)}$, and $\Sigma(n)$ is a a $P \times P$ matrix whose (p, q) element is $\sigma_{p\to q}$ to be introduced soon.

Similarly $\phi_y(W_y, Z(n), \cdot)$ and $\phi_{\backslash y}(W_y, Z(n), \cdot)$ can be defined as $\phi_y(W_y, Z(n), \cdot) = \sum_{n,y(n)=y}(\varphi(W_y, Z(n), d(n)) - \overline{\varphi(W_y, Z(n), d(n))})^2$ and $\phi_{\backslash y}(W_y, Z(n), \cdot) = \sum_{n,y(n)\neq y}(\varphi(W_y, Z(n), d(n)) - \overline{\varphi(W_y, Z(n), d(n))})^2$.

Intuitively, we are looking for a best parameter vector W_y of the ranking function $\varphi(W_y, Z(n), d(n))$ which is supposed to report large (resp. small) score for type y and small (resp. large) score for non-type y activity $d(n)$. When learning the vector, we use the following optimal criteria: 1) The scores for samples of type y (resp. not of type y) are close to each other as much as possible; 2) Scores for all samples are scattered as much as possible. See In other words, we are looking for a discriminative signature W_y for each activity type y. Note that we have introduced a 'latent' pairwise interaction quantity $Z(n)$. Practically we weight the contributions of interactions from different pairs using $Z(n)$, and expect strong interactions (large elements in $Z(n)$) may contribute more to the score. However, we do not explicitly obtain Z but learn it in companion with W, while we do enforce the constraint that it should not deviate from the 'visual' (heuristic) interaction Σ too much.

Though the notations appear messy, the overall cost is bilinear in W_y and $Z(n)$ and an alternating coordinate ascent in W_y and $Z(n)$ will suffice. With proper regularization (e.g., norm constraint for W_y, $\|W_y\| = 1$, and degree constraint for $Z(n)$,$\mathbf{1}^T Z(n) = \mathbf{1}^T$), either step will become eigen decomposition problem or quadratic programming, which can be solved after standard derivation. For an incoming segment d_{test} together with Σ_{test}, the ranking is straightforward using $\varphi(W_y, Z_{test}, d_{test})$ with constraint $tr((Z_{test} - \Sigma_{test})^T (Z_{test} - \Sigma_{test})) < \epsilon$.

7.2 From Complex Human Motions to Complex Dynamics in Videos

Complex human activities can be regarded as special cases of complex dynamics in videos, and thus the study on the former brings new inspirations toward the latter. Activities mainly refer to, or are related to, motions, i.e., dynamics of the geometric/spatial locations, while dynamic appearances, in parallel with dynamic locations, are another major family of dynamics emerging from videos. A typical sub-area of dynamic appearances

is dynamic texture [55]. Video dynamics, apparently, is a coupling of both motion and appearance variation, which we hereby call 'complex dynamics'. This brings more challenges to the current mathematical models available, as we normally expect long-term and non-stationary dynamics, of both motion and appearance variation, in real-world videos. Even constrained within motions, a spatio-temporal generator or descriptor for long-term non-stationary ensemble motions remains unsolved problem. Note that in Chapter 3 we condense information onto temporal correlation and in Chapter 4 we condense them largely on to the spatial distribution. We have not obtained a rigorous spatio-temporal approach.

Lie group and Lie algebra [167], exploited in Chapter 4 in the context of describing relevant particle movements, bring potentially new insights. A linear time-invariant dynamic system closely relates to a single element in the Lie group, and a mixture of them simply becomes a set of points on the Lie group manifold. Moreover, a time-varying linear system, under this formulation, simply becomes a path on the Lie group as well. Any combination of these cases, will correspond to collection of points or paths on the manifold, which has been well studied. Lie algebra is a direct linearization of Lie group, while the points on Lie group will find their images under particular canonical mappings in the linear space of Lie algebra , for which we already have a variety of powerful tools. The Lie theory has not played an active role in video related problems and we believe new insight will be drawn from further exploration of it.

Instead of making use of hybrid time-varying linear dynamics to model complex events in video, we can also directly explore a nonlinear dynamic model. To overcome the analytical difficulties that may potentially exist in analyzing a general nonlinear dynamic process, we may focus on special cases, e.g., periodic or quasi-periodic motions, as well as deterministically/stochastically mixed motion (chaos). In fact, these dynamics have broad spectrum of applications in videos, such as repetitive human activities, fluctuated optical flows induced by moving crowds, as well as dynamic textures. Last but not least, non-parametric time-series will also probably find it new applications. For instance, it may be interesting to develop a dynamic version of compressive-sensing-based methods, which enforce sparsity in the joint spatio-temporal-exemplar space.

7.3 Statistics and Geometry

We have devoted efforts on establishing statistical inference methods on geometrically distributed data. Boosting discriminative densities on DTIM, learning principal curve on co-occurrence manifold, and constructing CGMM all fall into this category. With analytical intrinsic geometry or sample-based imbedded topology on hand, one can develop more statistical tools for new applications whenever necessary.

We have also proposed two stochastic optimization algorithms on a geometric space. To make use of the idea of importance sampling, we established state evolution models on the manifold and derived corresponding local/global search strategies. Model-based optimization with geometric constraint will be useful in a generic sense, not limited to the application of spatio-temporal alignment. To make the sampling-based stochastic search more efficient, improvement may be possibly achieved by introducing other greedy ingredients in.

In companion with statistics on geometric space, the geometric space of statistical models has received little attention from vision community. The study on geometric space of statistical models belongs to the scope of information geometry [150, 151], which we believe is worth trying for understanding video dynamics as well.

Bibliography

[1] S. M. Khan and M. Shah, "Detecting group activities using rigidity of formation," in *ACM Multimedia 2005*, 2005.

[2] N. Vaswani, A. Roy-Chowdhury, and R. Chellappa, "Shape activity: A continuous-state hmm for moving/deforming shapes with application to abnormal activity detection," *IEEE Transactions on Image Processing*, vol. 14, pp. 1603 – 1616, 2005.

[3] B. Ni, S. Yan, and A. Kassim, "Recognizing human group activities by localized causalities," in *CVPR*, 2009.

[4] W. Choi, K. Shahid, and S. Savarese., "What are they doing?: Collective activity classification using spatio-temporal relationship among people," in *9th International Workshop on Visual Surveillance*, 2009.

[5] S. Gong and T. Xiang, "Recognition of group activities using dynamic probabilistic networks," in *ICCV*, 2003.

[6] S. Intille and A. Bobick, "Recognizing planned, multiperson action," *Computer Vision and Image Understanding*, vol. 81, pp. 414 – 445, 2001.

[7] T. Lan, Y. Wang, W. Yang, and G. Mori, "Beyond actions: Discriminative models for contextual group activities," in *NIPS*, 2010.

[8] V. Morariu and L. Davis, "Multi-agent event recognition in structured scenarios," in *CVPR*, 2011.

[9] S. Joo and R. Chellappa, "A multiple-hypothesis approach for multiobject visual tracking," *IEEE Transactions on Image Processing*, vol. 16, no. 11, pp. 2849 – 2854, 2007.

[10] F. Padua, R. Carceroni, G. Santos, and K. Kutulakos, "Linear sequence-to-sequence alignment," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 2, pp. 304–320, 2010.

[11] S. R. Rao, R. Tron, R. Vidal, and Y. Ma, "Motion segmentation via robust subspace separation in the presence of outlying, incomplete, or corrupted trajectories," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[12] Y. Caspi, D. Simakov, and M. Irani, "Feature-based sequence-to-sequence matching," *International Journal of Computer Vision*, vol. 68, no. 1, pp. 53–64, 2006.

[13] Y. Ukrainitz and M. Irani, "Aligning sequences and actions by maximizing space-time correlations," *Proceedings of IEEE European Conference on Computer Vision*, 2006.

[14] I. L. Dryden and K. V. Mardia, *Statistical Shape Analysis*. John Wiley and Sons, 1998.

[15] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Journal of Computing Surveys*, vol. 38, no. 4, December 2006.

[16] A. Srivastava and I. H. Jermyn, "Looking for shapes in two-dimensional point clouds," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 9, pp. 1616–1629, 2009.

[17] P. Turaga, R. Chellappa, V. Subrahmanian, and O. Udrea, "Machine recognition of human activities: A survey." *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, pp. 1473 – 1488, 2008.

[18] J. Aggarwal and Q. Cai, "Human motion analysis: a review," *Computer Vision and Image Understanding*, vol. 73, no. 3, pp. 428–440, 1999.

[19] C. Cedras and M. Shah, "Motion-based recognition: A survey," *Image and Vision Computing*, vol. 13, no. 2, pp. 129–155, 1995.

[20] D. M. Gavrila, "The visual analysis of human movement: a survey," *Computer Vision and Image Understanding*, vol. 73, no. 1, pp. 82–98, 1999.

[21] T. B. Moeslund, A. Hilton, and V. Krüger, "A survey of advances in vision-based human motion capture and analysis," *Computer Vision and Image Understanding*, vol. 104, no. 2, pp. 90–126, 2006.

[22] R. Poppe, "A survey on vision-based human action recognition," *Image and Vision Computing*, vol. 28, no. 6, pp. 976–990, 2010.

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

[24] T. Huang, D. Koller, J. Malik, G. H. Ogasawara, B. Rao, S. J. Russell, and J. Weber, "Automatic symbolic traffic scene analysis using belief networks," *National Conference on Artificial Intelligence*, pp. 966–972, 1994.

[25] A. M. Elgammal, D. Harwood, and L. S. Davis, "Non-parametric model for background subtraction," *Proceedings of IEEE European Conference on Computer Vision*, pp. 751–767, 2000.

[26] M.-K. Hu, "Visual pattern recognition by moment invariants," *IEEE Transactions on Information Theory*, vol. 8, no. 2, pp. 179–187, 1962.

[27] H. Freeman, "On the encoding of arbitrary geometric configurations," *IRE Transactions on Electronic Computers*, vol. 10, no. 2, pp. 260–268, 1961.

[28] D. G. Kendall, "Shape manifolds, procrustean metrics and complex projective spaces," *Bulletin of London Mathematical society*, vol. 16, pp. 81–121, 1984.

[29] H. Blum and R. N. Nagel, "Shape description using weighted symmetric axis features," *Pattern Recognition*, vol. 10, no. 3, pp. 167–180, 1978.

[30] A. Bissacco, P. Saisan, and S. Soatto, "Gait recognition using dynamic affine invariants," *International Symposium on Mathematical Theory of Networks and Systems*, 2004.

[31] R. Polana and R. Nelson, "Low level recognition of human motion (or how to get your man without finding his body parts)," *Proceedings of the IEEE Workshop on Motion of Non-Rigid and Articulated Objects*, pp. 77–82, 1994.

[32] S. M. Seitz and C. R. Dyer, "View-invariant analysis of cyclic motion," *International Journal of Computer Vision*, vol. 25, no. 3, pp. 231–251, 1997.

[33] A. F. Bobick and J. W. Davis, "The recognition of human movement using temporal templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 3, pp. 257–267, 2001.

[34] O. Chomat and J. L. Crowley, "Probabilistic recognition of activity using local appearance," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 02, pp. 104–109, 1999.

[35] L. Zelnik-Manor and M. Irani, "Event-based analysis of video," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 123–130, 2001.

[36] I. Laptev and T. Lindeberg, "Space-time interest points," *Proceedings of IEEE International Conference on Computer Vision*, 2003.

[37] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," *IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, 2005.

[38] J. C. Niebles, H. Wang, and L. Fei Fei, "Unsupervised learning of human action categories using spatial-temporal words," *British Machine Vision Conference*, 2006.

[39] T. F. Syeda-Mahmood, M. Vasilescu, and S. Sethi, "Recognizing action events from multiple viewpoints," *IEEE Workshop on Detection and Recognition of Events in Video*, 2001.

[40] A. Yilmaz and M. Shah, "Actions sketch: A novel action representation," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 984–989, 2005.

[41] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 12, pp. 2247–2253, 2007.

[42] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[43] J. Yamato, J. Ohya, and K. Ishii, "Recognizing human action in time-sequential images using hidden markov model," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 379–385, 1992.

[44] A. D. Wilson and A. F. Bobick, "Learning visual behavior for gesture analysis," *Proceedings of the International Symposium on Computer Vision*, pp. 229–234, 1995.

[45] T. Starner, J. Weaver, and A. Pentland, "Real-time american sign language recognition using desk and wearable computer based video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 12, pp. 1371–1375, 1998.

[46] A. Kale, A. Sundaresan, A. N. Rajagopalan, N. P. Cuntoor, A. K. Roy-Chowdhury, V. Kruger, and R. Chellappa, "Identification of humans using gait," *IEEE Transactions on Image Processing*, vol. 13, no. 9, pp. 1163–1173, 2004.

[47] Z. Liu and S. Sarkar, "Improved gait recognition by gait dynamics normalization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 6, pp. 863–876, 2006.

[48] M. Brand, N. Oliver, and A. Pentland, "Coupled hidden markov models for complex action recognition," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 994–999, 1997.

[49] D. J. Moore, I. A. Essa, and M. H. Hayes, "Exploiting human actions and object context for recognition tasks," *Proceedings of IEEE International Conference on Computer Vision*, pp. 80–86, 1999.

[50] S. Hongeng and R. Nevatia, "Large-scale event detection using semi-hidden markov models," *Proceedings of IEEE International Conference on Computer Vision*, pp. 1455–1462, 2003.

[51] N. P. Cuntoor and R. Chellappa, "Mixed-state models for nonstationary multiobject activities," *EURASIP Journal of Applied Signal Processing*, vol. 2007, no. 1, pp. 106–119, 2007.

[52] A. Bissacco, A. Chiuso, Y. Ma, and S. Soatto, "Recognition of human gaits," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 52–57, 2001.

[53] A. Veeraraghavan, A. Roy-Chowdhury, and R. Chellappa, "Matching shape sequences in video with an application to human movement analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 12, pp. 1896–1909, 2005.

[54] M. C. Mazzaro, M. Sznaier, and O. Camps, "A model (in)validation approach to gait classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 11, pp. 1820–1825, 2005.

[55] G. Doretto, A. Chiuso, Y. N. Wu, and S. Soatto, "Dynamic textures," *International Journal of Computer Vision*, vol. 51, no. 2, pp. 91–109, 2003.

[56] A. B. Chan and N. Vasconcelos, "Classifying video with kernel dynamic textures," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[57] L. Ljung, Ed., *System identification (2nd ed.): theory for the user*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 1999.

[58] K. D. Cock and B. D. Moor, "Subspace angles between arma models," *Systems and Control Letters*, vol. 46, pp. 265–270, 2002.

[59] R. Vidal and P. Favaro, "Dynamicboost: Boosting time series generated by dynamical systems," *Proceedings of IEEE International Conference on Computer Vision*, 2007.

[60] S. V. N. Vishwanathan, A. J. Smola, and R. Vidal, "Binet-cauchy kernels on dynamical systems and its application to the analysis of dynamic scenes," *International Journal of Computer Vision*, vol. 73, no. 1, pp. 95–119, 2007.

[61] A. Bissacco and S. Soatto, "On the blind classification of time series," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[62] C. Bregler, "Learning and recognizing human dynamics in video sequences," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, p. 568, 1997.

[63] B. North, A. Blake, M. Isard, and J. Rittscher, "Learning and classification of complex dynamics," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 9, pp. 1016–1034, 2000.

[64] V. Pavlovic, J. M. Rehg, and J. MacCormick, "Learning switching linear models of human motion," *Advances in Neural Information Processing Systems*, pp. 981–987, 2000.

[65] S. M. Oh, J. M. Rehg, T. R. Balch, and F. Dellaert, "Data-driven mcmc for learning and inference in switching linear dynamic systems," *National Conference on Artificial Intelligence*, pp. 944–949, 2005.

[66] R. Vidal, A. Chiuso, and S. Soatto, "Observability and identifiability of jump linear systems," *Proceedings of IEEE Conference on Decision and Control*, pp. 3614–3619, 2002.

[67] M. I. Jordan, *Learning in Graphical Models*. The MIT Press, 1998.

[68] F. R. Kschischang, B. J. Frey, and H. A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 498–519, 2001.

[69] N. Friedman and D. Koller, "Being Bayesian about Bayesian network structure: A Bayesian approach to structure discovery in Bayesian networks." *Machine Learning*, vol. 50, no. 1–2, pp. 95–125, 2003.

[70] H. Buxton and S. Gong, "Visual surveillance in a dynamic and uncertain world," *Artificial Intelligence*, vol. 78, no. 1-2, pp. 431–459, 1995.

[71] P. Remagnino, T. Tan, and K. Baker, "Agent orientated annotation in model based visual surveillance," *Proceedings of IEEE International Conference on Computer Vision*, pp. 857–862, 1998.

[72] S. Park and J. K. Aggarwal, "Recognition of two-person interactions using a hierarchical bayesian network," *ACM Journal of Multimedia Systems, Special Issue on Video Surveillance*, vol. 10, no. 2, pp. 164–179, 2004.

[73] C. Castel, L. Chaudron, and C. Tessier, "What is going on? A High-Level Interpretation of a Sequence of Images," *ECCV Workshop on Conceptual Descriptions from Images*, 1996.

[74] N. Ghanem, D. DeMenthon, D. Doermann, and L. Davis, "Representation and Recognition of Events in Surveillance Video Using Petri Nets," *Second IEEE Workshop on Event Mining 2004, CVPR2004*, 2004.

[75] M. Albanese, R. Chellappa, V. Moscato, A. Picariello, V. S. Subrahmanian, P. Turaga, and O. Udrea, "A constrained probabilistic petri net framework for human activity detection in video," *Submitted to IEEE Transactions on Multimedia*.

[76] M. Brand, "Understanding manipulation in video," *Proceedings of the 2nd International Conference on Automatic Face and Gesture Recognition*, p. 94, 1996.

[77] M. S. Ryoo and J. K. Aggarwal, "Recognition of composite human activities through context-free grammar based representation," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1709–1718, 2006.

[78] Y. A. Ivanov and A. F. Bobick, "Recognition of visual activities and interactions by stochastic parsing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 852–872, 2000.

[79] D. Moore and I. Essa, "Recognizing multitasked activities from video using stochastic context-free grammar," *Eighteenth national conference on Artificial intelligence*, pp. 770–776, 2002.

[80] A. S. Ogale, A. Karapurkar, and Y. Aloimonos, "View-invariant modeling and recognition of human actions using grammars," *ECCV Workshop on Dynamical Vision*, pp. 115–126, 2006.

[81] S. W. Joo and R. Chellappa, "Recognition of multi-object events using attribute grammars," *International Conference on Image Processing*, pp. 2897–2900, 2006.

[82] G. Medioni, I. Cohen, F. Brémond, S. Hongeng, and R. Nevatia, "Event detection and analysis from video streams," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 8, pp. 873–889, 2001.

[83] S. Hongeng, R. Nevatia, and F. Bremond, "Video-based event recognition: activity representation and probabilistic recognition methods," *Computer Vision and Image Understanding*, vol. 96, no. 2, pp. 129–162, 2004.

[84] V. D. Shet, D. Harwood, and L. S. Davis, "Vidmap: video monitoring of activity with prolog," *Proceedings of IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 224–229, 2005.

[85] D. Chen, J. Yang, and H. D. Wactlar, "Towards Automatic Analysis of Social Interaction Patterns in a Nursing Home Environment from Video," *MIR 04: Proceedings of the 6th ACM SIGMM international workshop on Multimedia information retrieval*, pp. 283–290, 2004.

[86] A. Hakeem and M. Shah, "Ontology and Taxonomy Collaborated Framework for Meeting Classification." *International Conference on Pattern Recognition*, pp. 219–222, 2004.

[87] B. Georis, M. Maziere, F. Bremond, and M. Thonnat, "A Video Interpretation Platform Applied to Bank Agency Monitoring," *2nd Workshop on Intelligent Distributed Surveillance Systems (IDSS)*, 2004.

[88] F. Bremond and M. Thonnat, "Analysis of Human Activities Described by Image Sequences," *Intl. Florida AI Research Symposium*, 1997.

[89] Event Ontology Workshop. http://www.ai.sri.com/∼burns/EventOntology.

[90] J. Hobbs, R. Nevatia, and B. Bolles, "An Ontology for Video Event Representation," *IEEE Workshop on Event Detection and Recognition*, 2004.

[91] Y. Sheikh, M. Sheikh, and M. Shah, "Exploring the space of a human action," *Proceedings of IEEE International Conference on Computer Vision*, pp. 144–149, 2005.

[92] T. Darrell, I. Essa, and A. Pentland, "Task-specific gesture analysis in real-time using interpolated views," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, pp. 1236 – 1242, 1996.

[93] C. Rao and M. Shah, "View-invariance in action recognition," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 316–322, 2001.

[94] V. Parameswaran and R. Chellappa, "View invariants for human action recognition," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 613–619, 2003.

[95] ——, "View invariance for human action recognition." *International Journal of Computer Vision*, vol. 66, pp. 83 – 101, 2006.

[96] D. Weinland, R. Ronfard, and E. Boyer, "Free viewpoint action recognition using motion history volumes." *Computer Vision and Image Understanding*, vol. 104, pp. 249 – 257, 2006.

[97] A. F. Bobick and A. D. Wilson, "A state-based technique for the summarization and recognition of gesture," *Proceedings of IEEE International Conference on Computer Vision*, pp. 382–388, 1995.

[98] J. Hoey and J. J. Little, "Representation and recognition of complex human motion," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1752–1759, 2000.

[99] P. K. Turaga, A. Veeraraghavan, and R. Chellappa, "From videos to verbs: Mining videos for activities using a cascade of dynamical systems," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[100] K. Takahashi, S. Seki, E. Kojima, and R. Oka, "Recognition of dexterous manipulations from time-varying images," *Proceedings IEEE Workshop on Motion of Non-Rigid and Articulated Objects*, pp. 23–28, 1994.

[101] M. A. Giese and T. Poggio, "Morphable models for the analysis and synthesis of complex motion patterns," *International Journal of Computer Vision*, vol. 38, no. 1, pp. 59–73, 2000.

[102] C. Rao, M. Shah, and T. Syeda-Mahmood, "Invariance in motion analysis of videos," *Proceedings of the eleventh ACM International Conference on Multimedia*, pp. 518–527, 2003.

[103] A. Veeraraghavan, R. Chellappa, and A. K. Roy-Chowdhury, "The function space of an activity," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 959–968, 2006.

[104] A. Gritai, Y. Sheikh, and M. Shah, "On the use of anthropometry in the invariant analysis of human actions," *International Conference on Pattern Recognition*, pp. 923–926, 2004.

[105] A. Veeraraghavan, R. Chellappa, and M. Srinivasan, "Shape and behavior encoded tracking of bee dances," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 3, pp. 463 – 476, 2008.

[106] Y. Zhou, S. Yan, and T. S. Huang, "Pair-activity classification by bi-trajectories analysis," in *CVPR*, 2008.

[107] X. Ma, F. Bashir, A. Khokhar, and D. Schonfeld, "Event analysis based on multiple interactive motion trajectories," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, pp. 397 – 406, 2009.

[108] A. Hoogs, S. Bush, G. Brooksby, A. Perera, M. Dausch, and N. Krahnstoever, "Detecting semantic group activities using relational clustering," in *IEEE Workshop on Motion and Video Computing*, 2008.

[109] S. Hongeng and R. Nevatia, "Multi-agent event recognition," in *ICCV*, 2001.

[110] X. Liu and C. Chua, "Multi-agent activity recognition using observation decomposed-hidden markov models," *Image and Vision Computing*, vol. 24, no. 2, pp. 166 – 175, 2006.

[111] A. Hakeem and M. Shah, "Learning, detection and representation of multi-agent events in videos," *Artificial Intelligence*, vol. 171, pp. 586 – 605, 2007.

[112] I. McCowan, D. Gatica-Perez, S. Bengio, G. Lathoud, M. Barnard, and D. Zhang, "Automatic analysis of multimodal group actions in meetings," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 305 – 317, 2005.

[113] M. S. Ryoo and J. K. Aggarwal, "Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities," in *ICCV*, 2009.

[114] M. Lazarescu and S. Venkatesh, "Using camera motion to identify different types of american football plays," *International Conference on Multimedia and Expo*, pp. 181 – 184, 2003.

[115] T. Liu, W. Ma, and H. Zhang, "Effective feature extraction for play detection in american football video," *ACM Multimedia Modeling*, 2005.

[116] C. Huang, H. Shih, and C. Chao, "Semantic analysis of soccer video using dynamic bayesian network," *IEEE Transactions on Multimedia*, vol. 8, no. 4, pp. 749 – 760, 2006.

[117] E. Swears and A. Hoogs, "Learning and recognizing American football plays," in *Snowbird Learning Workshop*, 2009.

[118] M. Perse, M. Kristan, S. Kovacic, G. Vuckovic, and J. Pers, "A trajectory-based analysis of coordinated team activity in a basketball game," *Computer Vision and Image Understanding*, vol. 113, no. 5, pp. 612 – 621, 2009.

[119] P. Turaga, A. Veeraraghavan, A. Srivastava, and R. Chellappa, "Statistical analysis on manifolds and its applications to video analysis," *Video Search and Mining, Studies in Computational Intelligence*, 2010.

[120] R. Brockett, "System theory on group manifolds and coset spaces," *SIAM Journal on Control*, vol. 10, pp. 265–284, 1972.

[121] ——, "Notes on stochastic processes on manifolds," *Systems and Control in the Twenty-First Century: Progress in Systems and Control*, 1997.

[122] U. Grenander, *Probabilities on Algebraic Structures*. Wiley, Chichester, 1963.

[123] ——, *General Pattern Theory*. Oxford University Press, 1993.

[124] U. Grenander and M. Miller, "Computational anatomy: An emerging discipline," *Quarterly of Applied Mathematics*, vol. LVI, pp. 617–694, 1998.

[125] M. Miller and L. Younes, "Group actions, homeomorphisms, and matching: A general framework," *International Journal of Computer Vision*, vol. 41, pp. 61–84, 2001.

[126] U. Grenander, M. Miller, and A. Srivastava, "Hilbert-schmidt lower bounds for estimators on matrix lie groups for atr," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 790–802, 1998.

[127] A. Srivastava and E. Klassen, "Monte carlo extrinsic estimators for manifold-valued parameters," *Special issue of IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 299–308, 2001.

[128] ——, "Bayesian geometric subspace tracking," *Journal for Advances in Applied Probability*, vol. 36, pp. 43–56, 2004.

[129] A. Edelman, T. A. Arias, and S. T. Smith, "The geometry of algorithms with orthogonality constraints," *SIAM Journal Matrix Analysis and Application*, vol. 20, no. 2, pp. 303–353, 1999.

[130] P. Absil, "Optimization on manifolds: methods and applications," *Technical Report UCL-INMA-2009.043*, 2009.

[131] D. G. Kendall and H. L. Le, "The riemannian structure of euclidean shape spaces: a novel environment for statistics," *Annals of Statistics*, vol. 21, pp. 1225–1271, 1993.

[132] C. G. Small, *The Statistical Theory of Shape*. Springer, Heidelberg, 1996.

[133] E. Klassen, A. Srivastava, W. Mio, and S. Joshi, "Analysis of planar shapes using geodesic paths on shape spaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, pp. 372–383, 2004.

[134] A. Srivastava, S. H. Joshi, W. Mio, and X. Liu, "Statistical shape analysis: Clustering, learning, and testing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 4, 2005.

[135] W. Mio, A. Srivastava, and S. Joshi, "On shape of plane elastic curves," *International Journal of Computer Vision*, vol. 73, pp. 307–324, 2007.

[136] R. Bhattacharya and V. Patrangenaru, "Nonparametric estimation of location and dispersion on riemannian manifolds," *Journal for Statistical Planning and Inference*, vol. 108, pp. 23–36, 2002.

[137] ——, "Large sample theory of intrinsic and extrinsic sample means on manifolds-I," *Annals of Statistics*, vol. 31, no. 1, pp. 1–29, 2003.

[138] X. Pennec and N. Ayache, "Uniform distribution, distance and expectation problems for geometric features processing," *Journal of Mathematical Imaging and Vision*, vol. 9, pp. 49–67, 1998.

[139] A. Veeraraghavan, A. Srivastava, A. Roy-Chowdhury, and R. Chellappa, "Rate-invariant recognition of humans and their activities," *IEEE Transactions on Image Processing*, vol. 18, no. 6, pp. 1326–1339, 2009.

[140] E. Begelfor and M. Werman, "Affine invariance revisited," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[141] P. Turaga, A. Veeraraghavan, and R. Chellappa, "Statistical analysis on stiefel and grassmann manifolds with applications in computer vision," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[142] Y. M. Lui and J. R. Beveridge, "Grassmann registration manifolds for face recognition," *Proceedings of IEEE European Conference on Computer Vision*, 2008.

[143] O. Tuzel, F. Porikli, and P. Meer, "Region covariance: A fast descriptor for detection and classification," *Proceedings of IEEE European Conference on Computer Vision*, 2006.

[144] ——, "Human detection via classification on riemannian manifolds." *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[145] R. Subbarao and P. Meer, "Nonlinear mean shift for clustering over analytic manifolds," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 1168–1175, 2006.

[146] R. Cutler and L. Davis, "Robust real-time periodic motion detection, analysis, and applications." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 781 – 796, 2000.

[147] I. N. Junejo, E. Dexter, I. Laptev, and P. Perez, "Cross-view action recognition from temporal self-similarities," *Proceedings of IEEE European Conference on Computer Vision*, 2008.

[148] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, no. 2, pp. 83 – 97, 1955.

[149] A. Srivastava, I. Jermyn, and S. Joshi, "Riemannian analysis of probability density functions with applications in vision," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[150] S. Amari and H. Nagaoka, *Methods of Information Geometry.* Oxford University Press, 2000.

[151] R. Kass and P. Vos, *Geometric Foundations of Asymptotic Inference.* John Wiley and Sons, 1997.

[152] X. Pennec, "Intrinsic statistics on riemannian manifolds: Basic tools for geometric measurements," *Journal of Mathematical Imaging and Vision*, vol. 25, no. 1, pp. 127 – 154, 2006.

[153] S. Rosset and E. Segal, "Boosting density estimation," *Advances in Neural Information Processing Systems*, 2002.

[154] M. Kim and V. Pavlovic, "Discriminative learning of mixture of bayesian network classifiers for sequence classification," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[155] LibSVM. http://www.csie.ntu.edu.tw/ cjlin/libsvm/.

[156] J. Costeira and T. Kande, "A multibody factorization method for independently moving objects," *International Journal of Computer Vision*, vol. 29, pp. 159 – 179, 1998.

[157] C. Gear, "Multibody grouping from motion images," *International Journal of Computer Vision*, no. 29, pp. 133 – 150, 1998.

[158] N. Ichimura, "Motion segmentation based on factorization method and discriminant criterion," *Proceedings of IEEE International Conference on Computer Vision*, 1999.

[159] K. Kanatani, "Motion segmentation by subspace separation and model selection," *Proceedings of IEEE International Conference on Computer Vision*, 2001.

[160] R. Vidal, Y. Ma, and S. Sastry, "Generalized principal component analysis (GPCA)," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 1945 – 1959, 2005.

[161] J. Yan and M. Pollefeys, "A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate," *Proceedings of IEEE European Conference on Computer Vision*, 2006.

[162] L. Zelnik-Manor and M. Irani, "Degeneracies, dependencies and their implications in multi-body and multi-sequence factorizations," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2003.

[163] R. Vidal and Y. Ma, "A unified algebraic approach to 2-d and 3-d motion segmentation," *Journal of Mathematical Imaging and Vision*, vol. 25, pp. 403 – 421, 2006.

[164] C. Bregler, A. Hertzmann, and H. Biermann, "Recovering non-rigid 3d shape from image streams," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2000.

[165] M. Brand, "Morphable 3d models from video," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

[166] J. Xiao, J. Chai, and T. Kanade, "A closed-form solution to non-rigid shape and motion recovery," *Proceedings of IEEE European Conference on Computer Vision*, 2004.

[167] W. Rossmann, *Lie Groups: An Introduction through Linear Groups.* Oxford University Press, 2003.

[168] L. V. Gool, T. Moons, E. Pauwels, and A. Oosterlinck, "Vision and lies approach to invariance," *Image and Vision Computing*, vol. 13, pp. 259 – 277, 1995.

[169] R. P. N. Rao and D. L. Ruderman, "Learning lie groups for invariant visual perception," *Advances in Neural Information Processing Systems*, 1999.

[170] T. Drummond and R. Cipolla, "Application of lie algebras to visual servoing," *International Journal of Computer Vision*, vol. 37, pp. 21 – 41, 2000.

[171] V. M. Govindu, "Lie-algebraic averaging for globally consistent motion estimation," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2004.

[172] E. Bayro-Corrochano and J. Ortegon-Aguilar, "Lie algebra approach for tracking and 3d motion estimation using monocular vision," *Image and Vision Computing*, vol. 25, pp. 907 – 921, 2007.

[173] O. Tuzel, R. Subbarao, and P. Meer, "Simultaneous multiple 3d motion estimation via mode finding on lie groups," *Proceedings of IEEE International Conference on Computer Vision*, 2005.

[174] D. Lin, E. Grimson, and J. Fisher, "Learning visual flows: A lie algebraic approach," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[175] D. Blei and M. Jordan, "Variational inference for dirichlet process mixtures," *Journal of Bayesian Analysis*, vol. 1, pp. 121 – 144, 2006.

[176] Y. Ke, R. Sukthankar, and M. Hebert, "Event detection in crowded videos," *Proceedings of IEEE International Conference on Computer Vision*, 2007.

[177] J. Shi and C. Tomasi, "Good features to track," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 1994.

[178] L. Lee, R. Romano, and G. Stein, "Monitoring activities from multiple video streams: Establishing a common coordinate frame," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 758–767, 2000.

[179] L. Wolf and A. Zomet, "Sequence to sequence self calibration," *Proceedings of IEEE European Conference on Computer Vision*, 2002.

[180] I. Laptev, S. Belongie, P. Perez, and J. Wills, "Periodic motion detection and segmentation via approximate sequence alignment," *Proceedings of IEEE International Conference on Computer Vision*, 2005.

[181] L. Wolf and A. Zomet, "Wide baseline matching between unsynchronized video sequences," *International Journal of Computer Vision*, vol. 68, no. 1, pp. 43–52, 2006.

[182] Y. Caspi and M. Irani, "A step towards sequence-to-sequence alignment," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2000.

[183] ——, "Alignment of non-overlapping sequences," *Proceedings of IEEE International Conference on Computer Vision*, 2001.

[184] F. Zhou and F. de la Torre, "Canonical time warping for alignment of human behavior," *Advances in Neural Information Processing Systems*, 2009.

[185] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," *IEE Proceedings F on RAdar and Signal Processing*, vol. 140, pp. 107–113, 1993.

[186] S. Maybank, "The Fisher-Rao metric for projective transformations of the line," *International Journal of Computer Vision*, vol. 63, pp. 191–206, 2005.

[187] A. Srivasatava and E. Klassen, "Bayesian geometric subspace tracking," *Advances in Applied Probability*, vol. 36(1), pp. 43–56, March 2004.

[188] Y. Wu, B. Wu, J. Liu, and H. Lu, "Probabilistic tracking on riemannian manifolds," *International Conference on Pattern Recognition*, 2008.

[189] J. Kwon, K. M. Lee, and F. C. Park, "Visual tracking via geometric particle filtering on the affine group with optimal importance functions," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[190] F. Porikli and P. Pan, "Refressed importance sampling on manifolds for efficient object tracking," *IEEE International Conference on Advanced Video and Signal Based Surveillance*, 2009.

[191] X. Liu, A. Srivastava, and K. Gallivan, "Optimal linear representations of images for object recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, pp. 662–666, 2004.

[192] Y. Chikuse, *Statistics on special manifolds, Lecture Notes in Statistics.* Springer, New York., 2003.

[193] S. Sarkar, P. J. Phillips, Z. Liu, I. Robledo, P. Grother, and K. W. Bowyer, "The human id gait challenge problem: Data sets, performance, and analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 162–177, 2005.

[194] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local svm approach," *International Conference on Pattern Recognition*, 2004.

[195] C. Rao, A. Yilmaz, and M. Shah, "View-invariant representation and recognition of actions," *International Journal of Computer Vision*, vol. 50, no. 2, pp. 203 – 226, 2002.

[196] S. Seitz and C. Dyer, "View-invariant analysis of cyclic motion." *International Journal of Computer Vision*, vol. 25, pp. 231 – 251, 1997.

[197] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," *Proceedings of IEEE International Conference on Computer Vision*, 2005.

[198] X. Wang, K. Tieu, and E. Grimson, "Learning semantic scene models by trajectory analysis," *Proceedings of IEEE European Conference on Computer Vision*, 2006.

[199] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888 – 905, 2000.

[200] M. Pavan and M. Pelillo, "Dominant sets and pairwise clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 167 – 172, 2007.

[201] S. Lazebnik, C. Schmid, and J. Ponce., "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[202] S. Savarese, J. M. Winn, and A. Criminisi, "Discriminative object class models of appearance and shape by correlatons," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[203] K. Mardia and P. Jupp, "Directional statistics," 2000.

[204] E. A. Nadaraya, "On estimating regression." *Theory of Probability and its Applications*, vol. 25, pp. 186 – 190, 1964.

[205] G. S.Watson, "Smooth regression analysis," *Sankhya*, vol. 26, pp. 101 – 116, 1964.

[206] B. Davis, P. Fletcher, E. Bullitt, and S. Joshi, "Population shape regression from random design data," *Proceedings of IEEE International Conference on Computer Vision*, 2007.

[207] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

[208] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 603–619, 2002.