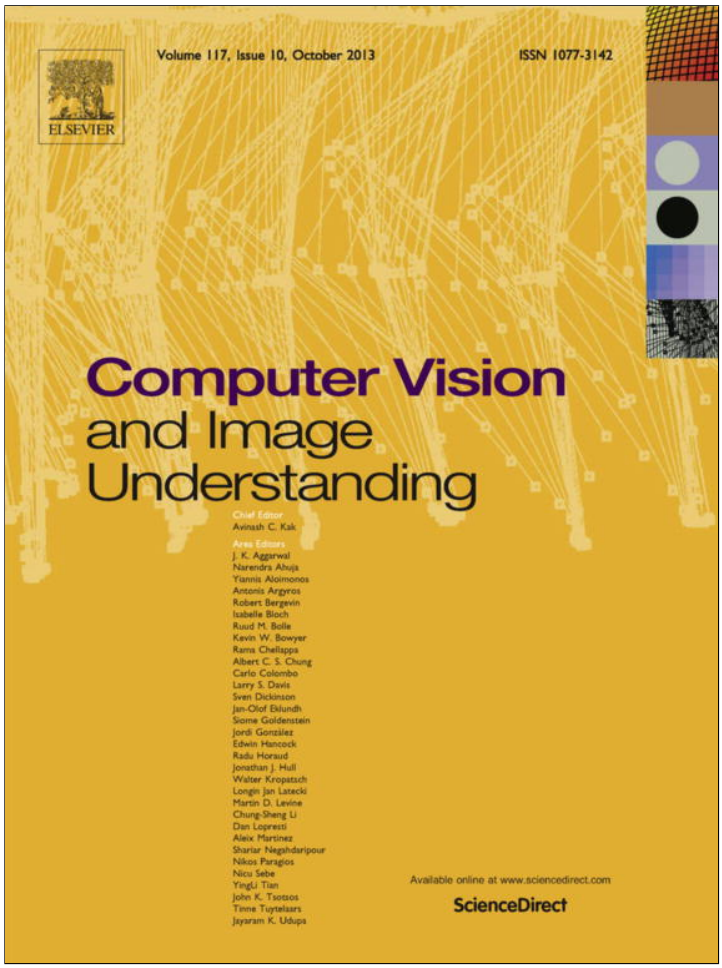

Computer Vision and Image Understanding 117 (2013) 1313–1328

Contents lists available at SciVerse ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Modeling multi-object interactions using "string of feature graphs"

Y. Zhu, N. Nayak, U. Gaur, B. Song, A. Roy-Chowdhury *,[1]

University of California, Riverside, CA 92521, USA

ARTICLE INFO

Article history:
Available online 25 November 2012

Keywords:
Complex activities
String-based activity recognition
Switching systems

ABSTRACT

In this paper, a novel generalized framework of activity representation and recognition based on a 'string of feature graphs (SFG)' model is introduced. The proposed framework represents a visual activity as a string of feature graphs, where the string elements are initially matched using a graph-based spectral technique, followed by a dynamic programming scheme for matching the complete strings. The framework is motivated by success of time sequence analysis approaches in speech recognition, but modified in order to capture the spatio-temporal properties of individual actions, the interactions between objects, and speed of activity execution. This framework can be adapted to various spatio-temporal motion features, and we show details on using STIP features and track features. Furthermore, we show how this SFG model can be embedded within a switched dynamical system (SDS) that is able to automatically choose the most efficient features for a particular video segment. This allows us to analyze a variety of activities in natural videos in a computationally efficient manner. Experimental results on the basic SFG model as well as its integration with the SDS are shown on some of the most challenging multi-object datasets available to the activity analysis community.

1. Introduction

The dynamical interactions between objects in a scene can be described using the following characterization: kinesics of individual objects (e.g., walking, running), temporal aspects (e.g., standing in a line), proximics or spatial relationship between objects (e.g., approaching), and haptics, (e.g., shaking hands, exchanging) [1]. Most work in activity recognition has concentrated on analyzing only one of these aspects (predominantly kinesics) as evidenced by the popular activity datasets like KTH [2] and Weizmann [3]. Most video analysis based applications such as surveillance, sports video analysis, and content-based search require effective approaches for modeling and recognition of far more complex activities than these datasets.

Recognition of complex activities requires understanding of spatio-temporal relationships between different objects, in addition to individual variability, cluttered background, viewpoint changes, and other environment induced conditions. Modeling all these parameters proves to be a challenging task. In this work, we focus primarily on modeling and recognition of complex activities that involve multiple interacting objects – people and vehicles (see Fig. 1 for examples).

The main challenge that needs to be overcome is to develop a generalized representation of the video that respects the spatio-temporal ordering of local features at different resolution levels, ranging from local image interest points to trajectories of individual objects. To achieve this goal, we build abstract graphs upon features. The spatio-temporal representations combined with graph-based spectral matching techniques provides a powerful framework to model complex activities in video, and an efficient computational strategy is applied to estimate the similarities between them. Our framework is motivated by success of time sequence analysis approaches in speech recognition, but modified in order to capture the spatio-temporal properties of individual actions, the interactions between objects, and speed of activity execution.

1.1. Overview of proposed framework

1.1.1. Feature descriptor

A video can be thought of as a spatio-temporal collection of primitive features (e.g., STIP features or track features). In order to handle the execution speed and motion variations of activities, we divide the video into small temporal bins. Each bin consists of a graphical structure representing the spatial arrangement of the local low-level features (see Fig. 2), which is, in this paper,

* Corresponding author.
E-mail addresses: yzhu010@ucr.edu (Y. Zhu), nandita.nayak.m@gmail.com (N. Nayak), utkarsh.gaur@yahoo.com (U. Gaur), bsong@ee.ucr.edu (B. Song), amitrc@ee.ucr.edu (A. Roy-Chowdhury).
[1] Y. Zhu, N. Nayak and A. Roy-Chowdhury are with the Electrical Engineering Dept at UCR. U. Gaur worked on this paper while a graduate student in Computer Science at UCR. He is currently at UCSB. B. Song worked on this paper while a postdoc in Electrical Engineering at UCR. She is currently at Sony Research USA.

1077-3142/$ - see front matter © 2012 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.cviu.2012.08.009

Fig. 1. Representative frames of the datasets used in this work. Note that the videos contain multiple actors performing activities simultaneously, sometimes in the presence of irrelevant subjects.

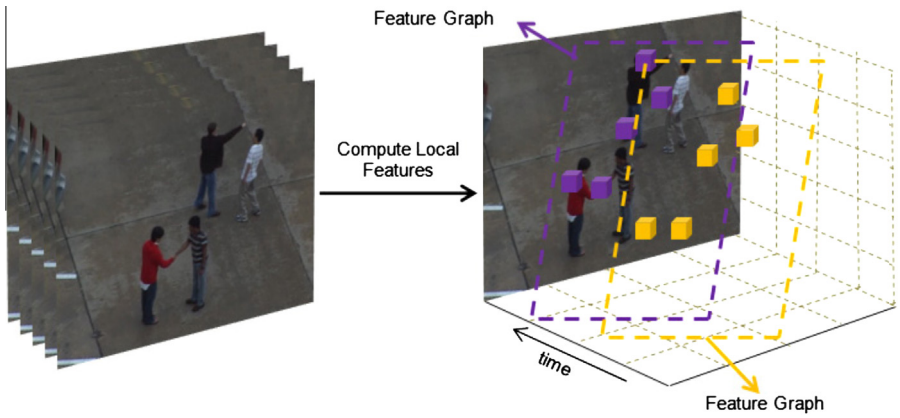

Fig. 2. Activity modeling: local features are computed from the video and grouped together to form feature collections. Temporally ordered strings of these local feature collections is termed as "string of feature-graphs" (SFGs).

called a *feature graph*. Since activities in the video evolve along time, it is natural to represent the video as a "string of feature graphs" (SFGs). Thus the query/training video becomes a string of such graphs, while a test video is also a string of graphs, albeit of a possibly higher complexity.

Then, the problem is, how to match these two strings of graphs. This is cast as a combination of sub-graph matching and time sequence alignment (see Fig. 3). The local feature collections are first matched in a graph-theoretic manner, thereby preserving the spatio-temporal relationships between features.

The final match score between the query and test video is a dynamic programming based temporal alignment score between their corresponding SFGs, thus compensating for differences in speed of execution. By combining local spatial matching with global temporal alignment, we are able to match videos while respecting their spatio-temporal structure of local features. This gives us the ability to recognize activities that involve interactions between multiple objects like people getting into/getting out of a vehicle, following, dispersing, and so on. Our graph matching scheme supports partial matching, i.e., given query examples, similar actions in a testing video can be retrieved even if the testing video contains multiple actions happening simultaneously.

The proposed framework for SFG-based modeling of activities can be implemented using any features which obey spatio-temporal ordering, such as STIP features, cuboid features, or track features. In this paper, we use the STIP features and track features to demonstrate the effectiveness of the framework. The STIP-based SFG method can be thought of as a generalization of the scheme in [4] where the spatio-temporal relationships were modeled using a collection of rules. Our proposed framework allows a more general structure on the video and does not need to recognize body parts, unlike [5,6], or primitive activities [7,8]. Additionally, our feature model tackles action recognition in the two modalities (classification- and query-based). The model is not intrinsically tied to any classification mechanism hence enabling its use in scenarios such as query-based retrieval, i.e., recognition with only a single (or very few) example video(s) of the activity in question. This is a highly desired feature since obtaining multiple training examples for increasingly complex activities is often difficult. We show experimental results on three relatively complex datasets namely the UT-Interaction dataset [9], VIRAT dataset [10] and UCR VideoWeb activity dataset [11]. All these datasets comprise of multiple interactive activities in realistic settings with clutter and changing backgrounds.

1.1.2. Adaptive feature selection

One of the desired properties of the proposed SFG modeling is adaptive feature selection. Natural videos contain activities of different kinds, some of which are very localized (e.g., people shaking hands) while others evolve over wider space–time spans (e.g., two people approaching). The analysis of such "local" and "global" activities requires different kinds of features. For example, the global activity of people approaching can be understood based on the tracks of the individuals (low-resolution features), whereas their handshaking requires more detailed information (higher resolution features). Most existing methods in activity recognition focus only on one level of video resolution and describe features that are relevant only for that scenario [12,13]. A few papers do combine

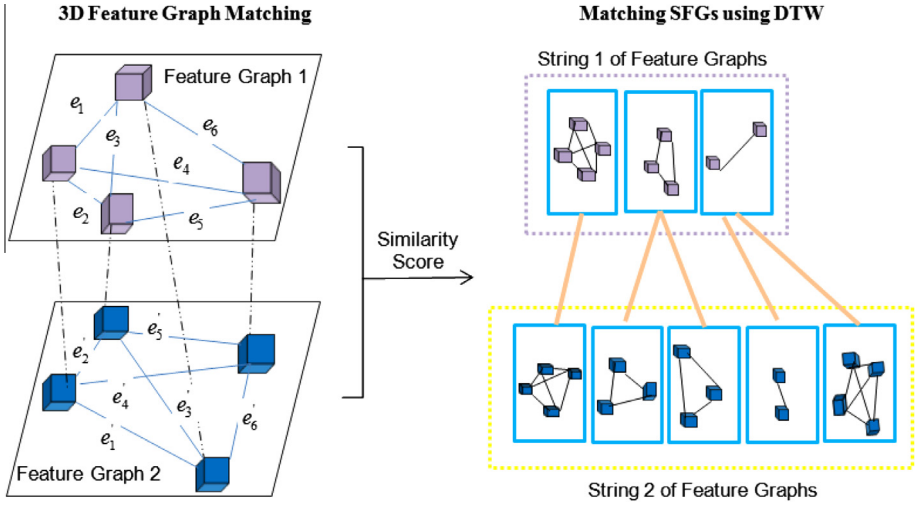

Fig. 3. (Left) Local feature-graphs are matched using the graph-based spectral technique in Section 3.1. (Right) The feature-graph matching scores thus generated are used in DTW matching of the two videos. which are represented by strings of feature graphs, to account for difference in speed and execution.

multiple resolutions in describing features for activity recognition [14,15]. However, they compute features at multiple resolution over each activity segment and integrate them. Our perspective in this work is to develop a switching system that adaptively selects between different feature types, using only one kind of feature in each time segment. This not only allows us to analyze a variety of activities in natural videos, but also does so in a computationally efficient manner.

Consider the example in Fig. 4. The first frame shows a person approaching a vehicle. This can be modeled and recognized using just a single track for the person (assuming that people and vehicles can be detected). However, this is not enough to understand what the person is doing near the vehicle. Higher resolution features are necessary at this stage. This can be done if we can design a system that will automatically switch to a different class of features. It requires developing schemes that will determine the optimum feature describing the right level of motion details and automatically switch between these multi-resolution features. Switching systems, which have been studied widely [16], provide an excellent mechanism to achieve this. Integrated with the SFG modeling of activity, the switching scheme also provides significant computational benefits. Higher resolution features, like those required to recognize a person a loading an object to a vehicle, are computationally more expensive to extract and analyze than low-resolution trajectories of individuals. The switching scheme allows for varying computational loads depending upon the analysis requirement.

1.2. Contributions

This paper makes the following contributions. It proposes a string-based feature representation of activities, the SFG, that respects the spatio-temporal ordering in the scene. It shows how image-based and track-based features can be used in the SFG. It also proposes a switching scheme to automatically choose between the different features, thus reducing computational complexity. Experimental results are shown on state-of-the-art datasets. The main differences of this submission with [17] are as follows: we have shown how the proposed SFG model can be applied to track-based features, in addition to STIP features; we have incorporated a model for switching between different feature types using a switched dynamical system, in order to reduce computation complexity while preserving the recognition accuracy; we have added a significant amount of new experimental results.

2. Related work

Activity recognition has been widely studied, but most of the literature has concentrated on relatively simple activities as evidenced in the KTH or Wiezmann datasets [13]. We focus on the modeling and recognition of more complex activities as explained above.

Complex activities usually involve multiple persons interacting with each other or with other objects like buildings and vehicles. The literature on complex activity modeling and recognition can

Fig. 4. Representative frames of global and local activities recognized in this work. The first frame shows a person approaching a vehicle (global activity). Low resolution motion features are suitable for recognition of such actions. The second frame shows a person loading a trunk. This is a local activity which can be recognized by examining high resolution motion features in the region marked in red (showed in the third frame). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

be classified into three categories: graphical, syntactic, and logical approaches [12,13]. Dynamic Bayesian networks (DBNs), which encode complex conditional dependencies between a set of random variables, is a representative graphical model used for complex activities [18]. Motivated by grammars in language modeling, syntactic approaches specify how activities can be constructed from action primitives, and use these rules as grammars for visual activity recognition [7,6].

Logic-based methods form logical rules to express commonsense knowledge to describe activities; for example, [8] represented each logical rule as first-order logic formula. Usually, such approaches rely on either tracking body parts [18,6], or object detection [18,8], or atomic action/primitive event recognition [7,8].

Spatio-temporal feature based approaches, like [19], hold more promise since no tracking is assumed. The statistics of these features are then used in recognition schemes [20]. However, as these approaches are built upon the statistics of extracted local features, spatial and long-term temporal correlations are often ignored.

The work in [21] models the video as a time-string of frame-wide feature histograms. It does bring the temporal aspect into picture; however the spatial structure information gets lost in the histogram representation. In [22], spatio-temporal relationships are considered by modeling activities as "strings of motion words". However, this method is limited to the availability of the tracks of objects involved. A matching kernel using "correlograms" was presented in [23], which looked at the spatio-temporal proximity among features. A recent work [4] proposed a match function to compare spatio-temporal relationships in the features by using temporal and spatial predicates. By considering the statistics of these relationships, the benefits of spatio-temporal modeling were demonstrated. The number of training videos needed to be large enough to represent the dataset.

Graphical models are commonly used to encode relationships in video analysis. In [24], variable length Markov models were used to learn qualitative spatio-temporal relations relevant to object interactions in the scene. A Dynamic Bayesian network was used to model the temporal evolution in two person activities in [25]. A grid based belief propagation method was used for pose estimation in [26]. Stochastic and context free grammars have been used to model complex activities in [6]. Co-occurring activities and their dependencies have been studied using Dependent Dirichlet Process-Hidden Markov Models (DDP-HMMs) in [27]. Graphical models usually require a good amount of training data.

Often, there are not enough training videos available for learning complex human activities; thus, recognizing activities based on just a single video example is of high interest. An approach for creating a large number of semi-artificial training videos from an original activity video was presented in [28]. A self-similarity descriptor that correlates local patches was proposed in [29]. A generalization of [29] was presented in [30], where space–time local steering kernels were used. These methods require a sliding window through time and space.

Few approaches have looked at integrating multiple features for activity recognition. Most of these approaches aim at using different features for a hierarchical analysis of activities. The authors in [14] have shown that both low resolution and high resolution features are needed for the understanding of human actions and interactions. An integrated framework for the recognition of objects and activities using multiple features has been demonstrated in [31]. A combination of space–time cuboids and vocabulary of spin images is used for single person activity recognition in [32]. Few approaches like [14] compute features at multiple resolutions and integrate them. Alternatively, choosing between multiple features is another possible approach.

Switched dynamical systems have been proposed to compute discrete switches between models in environments which exhibit continuous dynamics and discrete model changes. The authors in [33] cast a switched dynamical system as a Dynamical Bayesian network with applications in human motion analysis. A space dependent Markov chain was used to model the switches between models in [34]. A consensus between multiple classifiers for recognition was proposed in [35]. These approaches utilize a common feature set and the different classifiers are based on a common framework. In the part of our work that deals with adaptive feature selection, we propose switching between models that utilize different features.

3. String of feature graphs

In this section, we describe the framework of "string of feature graphs" modeling of activities in video. In order to take into account of the spatio-temporal properties of individual actions and interactions between objects involved in activity recognition, we represent the features within a time window as a feature graph. Dynamic time warping is applied upon the generated strings of feature graphs in the final recognition, which allows for variations in sampling rates and speed of activity execution.

3.1. Modeling complex activities using strings of feature-graphs

Let us consider a video V of duration T containing a complex activity. V can be represented as a collection of feature points $V = \left\{f^t_{x,y} | t \in [1,T]\right\}$ where $f^t_{x,y}$ is a feature point at spatial location x, y and time index t. Matching two videos would involve matching their corresponding feature points in a spatio-temporal order preserving manner. Let us divide the video into N intervals in time $t_0, t_1 \ldots, t_N$ and let the features contained in a single time interval be collectively denoted as F. Therefore, the video can now be represented as $V = \{F_1, F_2, \ldots, F_N\}$ where $F_1 = \left\{f^t_{x,y} | t \in [t_0, t_1)\right\}$, $F_2 = \left\{f^t_{x,y} | t \in [t_1, t_2)\right\}$, etc. Now, the spatio-temporal matching of two videos $V^{(1)}$ divided into N_1 intervals and $V^{(2)}$ divided into N_2 intervals would involve matching their individual feature collections $\left\{F^{(1)}_i | i = 1 \ldots N_1\right\}$ and $\left\{F^{(2)}_i | i = 1 \ldots N_2\right\}$ in a temporal order-preserving fashion, wherein the similarity measure between two feature collections would involve feature content matching as well as geometric structure matching. This representation of a video naturally leads us to a string representation, where local feature collections F form the elements of the string. In order to keep the structure information within each feature collection F, a graphical description is used and F is represented as a feature-graph. Therefore the temporally ordered collection of F forms a string of feature-graphs (SFGs). Fig. 2 visually explains the modeling process.

3.2. Spatio-temporal matching of string of feature-graphs

As explained earlier, the match score between two videos is the string alignment score between their corresponding SFGs. Since string alignment of any form requires a known method of measuring distance between the characters of the strings, we describe in the following subsections how we (a) use a spectral technique to compute similarity between two feature-graphs (feature-graphs being the characters in the SFG strings) and (b) use the so computed feature-graph match scores to find the optimal alignment score between two SFGs.

3.2.1. Matching two feature-graphs

Computing the similarity between two feature graphs involves matching individual feature-descriptors (i.e., nodes) as well as pairwise neighborhood relationships (i.e., edges).

We represent each feature collection, i.e., each character in the string, as a fully-connected three dimensional graph where feature points form the nodes. Then the feature correspondence problem can be formulated as a graph matching problem by considering the matching between both nodes and edges. Given two such graphs, one being a feature collection from the testing video, P, with n_P nodes, and one being a feature collection from query video, Q, with n_Q nodes, we follow the spectral technique described in [36] to find correspondences between their respective feature points (nodes). This approach avoids the combinatorial explosion inherent to the correspondence problem by formulating it in closed form as a spectral analysis problem on a graph adjacency matrix.

An assignment (i,i') is defined as a correspondence between a pair of nodes from two graphs, where $i \in P$ and $i' \in Q$. For each candidate assignment $a = (i,i')$, there is a distance score between feature i and feature i' associated with it. Let L be a list (with length $n_L = n_P \times n_Q$) of all possible candidate assignments between features of P and Q. Given such a list, let a matrix $\mathbf{M}$ (size $n_L \times n_L$) store the affinities of every possible pair of assignments $(a,b) \in L$. Note that $\mathbf{M}(a,a)$ for $a = (i,i')$ measures how well the feature point i matches the feature point i', and $\mathbf{M}(a,b)$, where $a = (i,i')$ and $b = (j,j')$, describes the relative pair-wise relationships of points (i,j) in P with points (i',j') in Q. We define $d_n(i,i')$ as the distance between the nodes i and i'. It measures the Euclidean distance between the features of nodes i and i'. In order to account for scale, we consider the geometric structure of the graphs based on the angles between the edges in the graph. We define $d_e(\overrightarrow{ij}, \overrightarrow{i'j'})$ as the distance between edges (i,j) and (i',j') based on the angle difference between them. For candidate assignments $a = (i,i')$ and $b = (j,j')$, the elements $\mathbf{M}(a,a)$ and $\mathbf{M}(a,b)$ of matrix $\mathbf{M}$ are defined as

$$\mathbf{M}(a,a) = \begin{cases} \omega_n[1 - d_n(i,i')] & d_n(i,i') \leqslant \tau_n \\ 0 & d_n(i,i') > \tau_n \end{cases}, \tag{1}$$

$$\mathbf{M}(a,b) = \begin{cases} \omega_e[1 - d_e(\overrightarrow{ij}, \overrightarrow{i'j'})] & d_e(\overrightarrow{ij}, \overrightarrow{i'j'}) \leqslant \tau_e \\ 0 & d_e(\overrightarrow{ij}, \overrightarrow{i'j'}) > \tau_e \end{cases}, \tag{2}$$

where τ_n is a pre-defined maximal distance between two features whose relationship should not be ignored and τ_e is a pre-defined threshold for edge difference. d_n and d_e are normalized between [0,1] and thus τ_n and τ_e are also chosen in that range. ω_n and ω_e are weights of node matching and edge matching, which adjust the relative importance of node similarity and edge similarity in the graph matching.

Now, suppose the length of the query feature graph is n_Q and the length of the testing feature graphs is n_P. Let x be an indicator vector of length $n_Q \times n_P$ such that $x(a) = 1$ if candidate assignment $a = (i,i')$ represents a corresponding pair of nodes and 0 otherwise. We aim to find an optimal solution x^* which maximizes the score

$$x^* = \arg\max_x x^T \mathbf{M} x. \tag{3}$$

The solution to the above problem, x^*, gives the optimal correspondence between feature points in P and Q. This solution has to be subject to the mapping constraints required by one-to-one mapping as in [36].

Once we estimate the optimal match, x^*, of two feature collections P and Q, their similarity can be measured by

$$sim(Q,P) = (x^*)^T \mathbf{M} x^*, \tag{4}$$

and the distance between them defined as

$$d(Q,P) = 1 - \frac{sim(Q,P)}{sim(Q,Q)}. \tag{5}$$

3.3. Dynamic time warping of SFGs

Recall that an SFG of a video is a time-ordered strings of its feature-graphs. Matching two SFGs should be flexible, in that it should be robust to the different rates at which an activity might occur and also the actual length of the template video and the test video. This can be achieved by time normalizing the two SFGs. The speech recognition community has successfully used a dynamic programming approach termed dynamic time warping (DTW) [37] for non-linear time normalization. We borrow this idea and apply it to flexibly match two SFGs, hence making them robust to speed differences in different instances of the activity.

The aim of DTW is to minimize the local distortion between two sequences by finding an optimal warping function φ. For our case, the local distortion is defined as the sum of local pair-wise distances between their feature collections. Formally, for two SFGs $\mathcal{Q} = \{Q_1 \dots Q_{N_\mathcal{Q}}\}$ and $\mathcal{P} = \{P_1 \dots P_{N_\mathcal{P}}\}$, where $N_\mathcal{Q}$ and $N_\mathcal{P}$ are the number of characters (i.e., feature graphs) in $\mathcal{Q}$ and $\mathcal{P}$ respectively, the sequence distortion is defined as

$$D_\varphi(\mathcal{Q},\mathcal{P}) = \frac{1}{M_\varphi} \sum_{k=1}^{K_\varphi} d(Q_{\varphi(k)}, P_{\varphi(k)}) m_k, \tag{6}$$

and the distance between the two SFGs can be computed as

$$D(\mathcal{Q},\mathcal{P}) = \arg\min_\varphi D_\varphi(\mathcal{Q},\mathcal{P}). \tag{7}$$

Here m_k are the path-weights, and $M_\varphi = \sum_k m_k$ is a normalization factor. The details of the solution to this optimization problem can be found in [37]. The entire matching process is pictorially presented in Fig. 3.

3.3.1. Subsequence DTW for continuous video

In real applications, the test video is often a continuous video containing multiple persons performing multiple activities. Given a query video, which often contains only the desired activity, we would want to find a subsequence within the testing video sequence that optimally fits the query sequence, i.e., identify the fragment within the testing video that is most similar to the query. For this purpose, we utilize a variant of DTW – subsequence DTW [38], by releasing the restriction on the boundary condition, as explained below.

Let $\mathcal{Q} = \{Q_1 \dots Q_{N_\mathcal{Q}}\}$ and $\mathcal{P} = \{P_1 \dots P_{N_\mathcal{P}}\}$ be two SFGs of the query and testing videos respectively, where $N_\mathcal{P} \gg N_\mathcal{Q}$. The goal is to find a subsequence $\mathcal{P}'(a^*,b^*) = \{P_{a^*} \dots P_{b^*}\}$ with $1 \leqslant a^* \leqslant b^* \leqslant N_\mathcal{P}$ such that

$$(a^*,b^*) = \arg\min_{(a,b):1\leqslant a\leqslant b\leqslant N_\mathcal{P}} (D(\mathcal{Q},\mathcal{P}'(a^*,b^*))). \tag{8}$$

The indices a^* and b^* can be computed by a small modification of the classical DTW algorithm in the generation of the accumulated cost matrix $\mathbf{C}$ used to describe the cost of aligning two sequences [38]. The goal of DTW is to find the minimal cost path through an accumulated cost matrix. By applying subsequence DTW, it can be shown that $b^* = \arg\min_{b\in[1,N_\mathcal{P}]} \mathbf{C}(N_\mathcal{Q}, b)$. $a^* \in [1, N_\mathcal{P}]$ is the maximal index such that path $(a^*,1)$ belongs to the warping path.

It is usually the case that the database contains multiple instances of the activity that are similar to the query example. It is desirable to retrieve all the subsequences of $\mathcal{P}$ that are close to $\mathcal{Q}$ with respect to the DTW distance. This can be achieved by recursively repeating the above process. We present our implementation of matching continuous video using subsequence DTW in Algorithm 1.

Algorithm 1. Matching SFG of continuous video through subsequence DTW

Input: $\mathcal{Q} = \{Q_1 \dots Q_{N_\mathcal{Q}}\}$ SFG of the query video
$\mathcal{P} = \{P_1 \dots P_{N_\mathcal{P}}\}$ SFG of the testing video
$\tau \in \mathbb{R}$ cost threshold
Output: Ranked list of all subsequences of $\mathcal{P}$ that have a DTW distance to $\mathcal{Q}$ below the threshold τ.

1. Initialize the ranked list to be an empty list.
2. Construct accumulated cost matrix **C** whose elements are defined as

$$\mathbf{C}(n,1) = \sum_{k=1}^{n} d(Q_k, P_1), n \in [1, N_\mathcal{Q}],$$
$$\mathbf{C}(1,m) = d(Q_1, P_m), m \in [1, N_\mathcal{P}],$$
$$\mathbf{C}(n,m) = \min\{\mathbf{C}(Q_{n-1}, P_{m-1}), \mathbf{C}(Q_{n-1}, P_m), \mathbf{C}(Q_n, P_{m-1})\} + d(Q_n, P_m).$$

3. Define a distance function: $\Delta(b) \triangleq \mathbf{C}(N_\mathcal{Q}, b)$, $b \in [1, N_\mathcal{P}]$.
4. Determine $b^* \in [1, N_\mathcal{P}]$ that gives minimal Δ.
5. If $\Delta(b^*) > \tau$ (which means no additional subsequence of $\mathcal{P}$ close to $\mathcal{Q}$ exists), then terminate the procedure.
6. Compute the corresponding DTW-minimizing index $a^* \in [1, N_\mathcal{P}]$ using standard DTW algorithm, which searches optimal warping path in **C** in reverse order of the indices starting with $(N_\mathcal{Q}, b^*)$.
7. Extend the ranked list by the subsequence $\mathcal{P}'(a^*, b^*)$.
8. Set $\Delta(b) \triangleq \infty$ for all b within a suitable neighborhood of b^*.
9. Continue with Step 4.

4. SFG Construction: special cases

Irrespective of the features used to describe a video, the task of activity recognition requires us to examine the properties of these features as well as their spatial and temporal arrangement. Therefore, although motion features can be very different from each other, we can represent the local spatio-temporal volume (STV) surrounding the targeted activity as an SFG, whose edge features define the geometric structure of its node features. In this section, we describe the construction of SFG for track and STIP based features. The main task is to develop suitable node and edge measurement techniques discussed in 3.2.1 for the particular motion features.

4.1. Track-based SFG

Activities involving objects exhibiting long-distance motion can be recognized from the global motion trends of the objects and their pattern of interactions [39–41]. Some examples of such activities are car u-turn, car turn, people dispersion, and group walking, etc. In this section, we implement the SFG framework in activity recognition based on motion features of tracks. Suppose we have trajectories and identifications of moving objects in the scene. The local STV surrounding each track is an interesting activity region. All the collected features of tracklets within the interesting spatio-temporal volume make up a feature graph.

4.1.1. Track descriptors

In this section, we develop four motion feature descriptors for tracks. The in-plane rotation-invariant descriptors – normalized change of gradient direction (NDG) and normalized change of gradient magnitude (NMG) – capture the global motion pattern of individual tracks. NDG of a track is its absolute change of gradient direction along time normalized by its maximum absolute value. NMG of a tracklet is its change of gradient magnitude along time normalized by the maximum magnitude of gradient. Let $\widetilde{t_i}$ be the track of object i, and $p_i(t) = [x_i(t) y_i(t)]$ for $t = 1, 2, \dots$ be the position of object i at time t. The features of the track i at time t are defined as

$$\mathrm{NDG}_i(t) = \frac{\left|\frac{d}{dt}\arctan\left(\frac{d_{x_i}(t)}{d_{y_i}(t)}\right)\right|}{max\left(\left|\frac{d}{dt}\arctan\left(\frac{d_{x_i}(t)}{d_{y_i}(t)}\right)\right|\right)}, \tag{9}$$

$$\mathrm{NMG}_i(t) = \frac{\sqrt{d_{x_i}(t)^2 + d_{y_i}(t)^2}}{max\left(\sqrt{d_{x_i}(t)^2 + d_{y_i}(t)^2}\right)}, \tag{10}$$

where $d_{x_i}(t)$ and $d_{y_i}(t)$ are the instantaneous gradients of object i along x and y axis respectively. It is easy to prove that both NDG and NMG are in-plane rotation invariant. Fig. 5 shows the sample descriptors.

Slope of smoothed relative distance (SRD) of a pair of tracks is the change of their relative distance smoothed along time, which captures the interaction trends between the two tracks. Relative distance of two tracks is obtained first. Break-points, where the

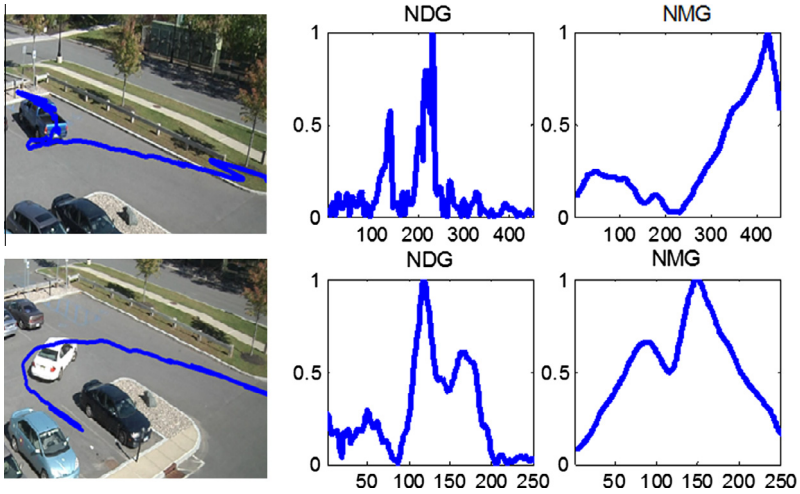

Fig. 5. Examples of NDG and NMG descriptors. The left column shows the sample images for a vehicle-backup (top) and a vehicle-u-turn (bottom) (only regions of interest are shown). The next two columns show the corresponding NDG and NMG descriptors.

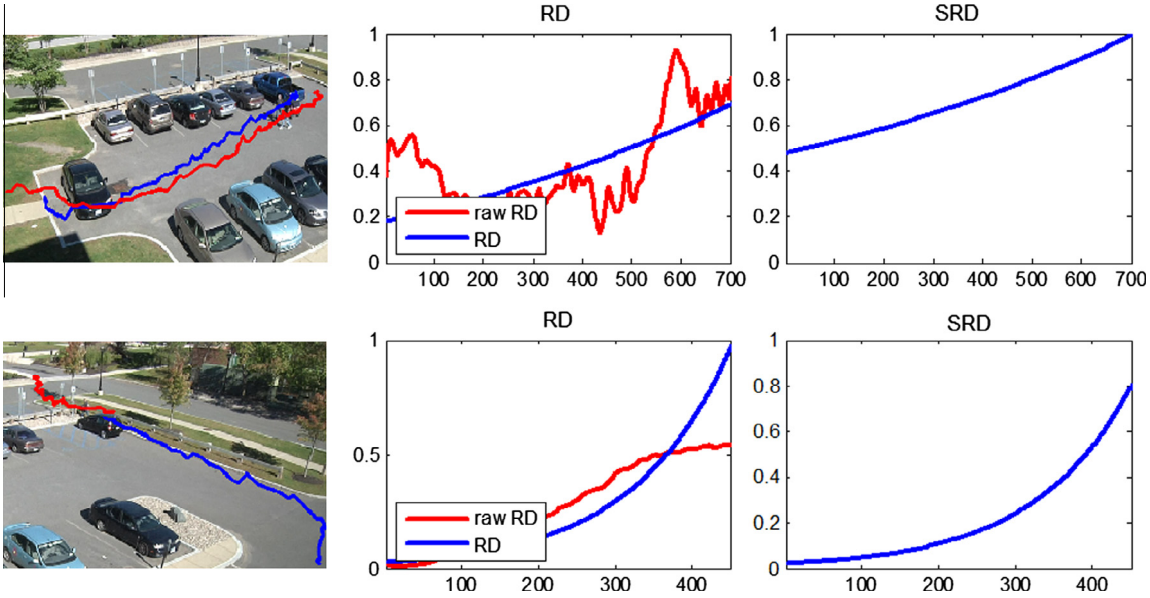

Fig. 6. Example of RD and SRD of two tracks. The images show sample frames of two people walking together (top) and person leaving a vehicle (bottom) (only regions of interest are shown). The graph on the left shows the raw relative distance between the two tracks and the exponential fitting result in each case. The graph on the right shows the derivative of smoothed relative distance (SRD) in each case.

trend of interaction changes (e.g., from approaching to dispersing) are detected and used to segment the RD descriptor. Break-points are defined as those local extrema of the relative distance sequence whose distance with the immediate previous extrema is greater than a pre-determined threshold. Exponential curve fitting is utilized to smooth out the segments in the resulting RD descriptor. Let $\widetilde{t}_i$ and $\widetilde{t}_j$ be the tracks of object i and j respectively, and $p_i(t) = [x_i(t) y_i(t)]$ and $p_j(t) = [x_j(t)\ y_j(t)]$ for $t = 1, 2, \ldots$ be the positions of objects i and j at time t. The relative distance of object i and j at time t is $d(t) = \sqrt{(x_i(t) - x_j(t))^2 + (y_i(t) - y_j(t))^2}$. The detected break points $t_1, t_2, \ldots, t_n$ and the beginning and ending points t_0, t_{n+1} segment the sequence of relative distance of the two objects into $n + 1$ segments $rd(k)$ for $k = 0, 1, \ldots, n$. The *RD* and *SRD* features of tracks of i and j at time t are defined as

$$RD_{(i,j)}(t) = exp\_fit(rd(k)) \quad \text{if} \quad t_k < t \leqslant t_{k+1}, \tag{11}$$

$$SRD_{(i,j)}(t) = \frac{RD_{(i,j)}(t)}{dt}, \tag{12}$$

where *exp_fit* refers to fitting an exponential function to the specific *rd* sequence. Fig. 6 shows the sample descriptors.

4.1.2. Track-based feature graph matching

In the feature graph matching, tracks are segmented into tracklets by concatenated equal-length time windows. Each tracklet forms a node in the graph. The node features in the graph are the smoothed motion features of the tracklets. The edge features quantize the interaction between the two underlying objects. It is natural to use the smoothed Euclidean distance between individual track features of two tracklets as the node distance measurement, and the smoothed distance between the interacting features of two pairwise tracklets as the edge distance measurement.

Assume tracklet i belongs to the query video, and tracklet i' belongs to the testing video. Let f_i^{IND} be the concatenated NDG or NMG features of tracklet i, and $f_{i',m}^{IND}$ be the concatenated NDG or NMG features of tracklet i'. Let $f_{\overrightarrow{ij}}^{SRD}$ be the concatenated SRD between i and query tracklet j. For a feature graph Q in the query video and a feature graph P in the testing video, the node distance, edge distance, and elements of similarity matrix defined in Section 3.2.1 are specified as

$$d_n(i, i') = \frac{\left\| f_i^{IND} - f_{i'}^{IND} \right\|}{s}, \tag{13}$$

$$d_e(\overrightarrow{ij}, \overrightarrow{i'j'}) = \frac{\left\| f_{\overrightarrow{ij}}^{SRD} - f_{\overrightarrow{i'j'}}^{SRD} \right\|}{s}, \tag{14}$$

where s is the length of a tracklet. When we are interested in only the interaction patterns of tracks involved in activities, ω_n defined in 3.2.1 is set to be zero, and only differences in track interactions are considered in the graph matching. When we are only interested in individual motion patterns of objects involved, ω_e is set to be zero, and only node differences are considered in the graph matching.

4.2. STIP-based SFG

Bag-of-Words (BoW) based on STIP features exhibits promising results in object categorization and semantic video retrieval across several datasets [42]. While the statistics of STIP features may indicate which candidate activity the test video contains, BoW needs large amount of training data to achieve good recognition performance. Also, it is easily understandable that the spatio-temporal arrangements of STIP clusters is essential for activity recognition. For instance, the actions – open a trunk and close a trunk – have very similar statistics of STIP descriptors, but the two are actually very different activities due to the different temporal order of STIP clusters. In this section, we systematically incorporate spatial and temporal information of STIPs in the activity recognition model, by implementing the SFG framework on top of STIP features. The proposed method can achieve the same recognition level with much less training data.

4.2.1. STIP feature graph matching

To extract spatial-temporal features, we rely on the spatio-temporal interest point (STIP) detector proposed in [43]. The STIPs are detected by finding the center locations of local spatio-temporal

volumes, which have large variations along both the spatial and the temporal directions, using a spatio-temporal extension of 2D Harris operator [43]. Then, STIP feature-graphs are constructed following the procedure described in Section 3.1.

In the matching of feature graphs with STIPs as the nodes, it is natural to use the Euclidean distance as the similarity measurement between two nodes, and use the difference between angles of two edges as the similarity measurement between the two edges. A STIP feature f typically consists of a location descriptor f^l, which indicates its 3-D location in the spatio-temporal domain, and a local motion descriptor f^m. Let $f_i = (f_i^l, f_i^m)$ be the STIP feature vector of node i, and $f_{i'}$ be the STIP feature vector of node i'. The distance measurements $d_n(i,i')$ and $d_e(\overrightarrow{ij}, \overrightarrow{i'j'})$ in Section 3.2.1 are specified as

$$d_n(i,i') = 1 - \frac{f_i^m \cdot f_{i'}^m}{\|f_i^m\| \|f_{i'}^m\|}, \tag{15}$$

$$d_e(\overrightarrow{ij}, \overrightarrow{i'j'}) = 1 - e^{\left[-p\left(1 - \frac{\left[\left(f_i^l - f_j^l\right)\right] \cdot \left[\left(f_{i'}^l - f_{j'}^l\right)\right]}{\left\|\left(f_i^l - f_j^l\right)\right\| \left\|\left(f_{i'}^l - f_{j'}^l\right)\right\|}\right)\right]}. \tag{16}$$

5. Adaptive feature selection

A video can be thought of as a spatio-temporal collection of primitive low resolution features and high resolution features. Recognition of activities can be achieved by different levels of motion details [44]. Low resolution features are often simpler and more sparse than high resolution features. They work better at recognizing activities characterized by global motion patterns, such as vehicle turn, group walking and people dispersion. On the other hand, algorithms based on high resolution features are suitable for recognizing activities in the local mode because more motion details are captured. Although such algorithms can also recognize global activities, they are often computationally expensive [44]. In order to improve the recognition accuracy while reducing computation complexity, it is important to choose motion features at the right scale of resolution for the recognition task. In this section, we integrate the proposed SFG modeling of activities into a Switched Dynamic System (SDS) to develop a scheme of adaptive feature selection in activity recognition. Our goal is to optimize the recognition accuracy as well as the computational complexity of our system by switching between the two kinds of features.

5.1. Switched dynamic system model

We propose a SDS model for the switching between activities for complex videos containing both global and local activities. In the SDS, two modes are considered: global mode and local mode corresponding to the global activity and local activity. Each spatio-temporal activity volume is assigned with a mode and the feature used in recognizing the activity is determined accordingly (how to locate these sptio-temporal activity volumes is introduced in Section 6.1). Motivated by works in hybrid systems like [45], the SDS model can be specified by the tuple

$$\Psi = \{M, O, \Phi, F^{low}, F^{high}\}, \tag{17}$$

where M denotes the modes in the system, O are the observations from which motion features are extracted. Φ is the attribute pattern derived from observations of low resolution motion details, and are used to decide the modes, F^{low} are the low resolution features and F^{high} are the high resolution features.

5.2. Switching between activity modes

In statistical pattern recognition methods, modes are also known as pattern classes. Each pattern class consists of different patterns, which can be represented by a vector of quantitative attributes $\Phi = [\phi^1, \phi^2, \ldots, \phi^a]$ carrying distinguishing information about the patterns [46], where a is the number of informative attributes. Each mode is assumed to have distinguishing distribution of these attributes. Thus the joint distribution of the informative attributes can be used to determine the mode of the observed pattern.

Let O_t be the observations of motion at time step t; the corresponding pattern is defined as $\Phi_t = \Gamma(O_t) = [\phi_t^1, \phi_t^2, \ldots, \phi_t^a]$, where Γ is the mapping from the observation space to the attribute space. Let $p(M)$ be the prior probability of each mode and $P(\Phi|M)$ be the distribution of attribute vector of a given mode M. Maximum likelihood can be used to determine the modes from the observed attributes. For an observed pattern Φ_t, the mode M_t of the pattern is

$$M_t = max_M[P(\Phi_t|M) \cdot p(M)]. \tag{18}$$

To simplify the estimation of probability distributions, we suppose that different types of attributes are independent of each other. A Naive Bayesian network can be applied to decide the underlying model M_t given a certain pattern Φ_t. Let g denote the global mode and l denote the local mode and $p(g)$ and $p(l)$ be the prior probabilities of global and local modes. Let the distributions of the ith attribute given the mode M be $p(\phi_i|M)$. The distribution of the attribute

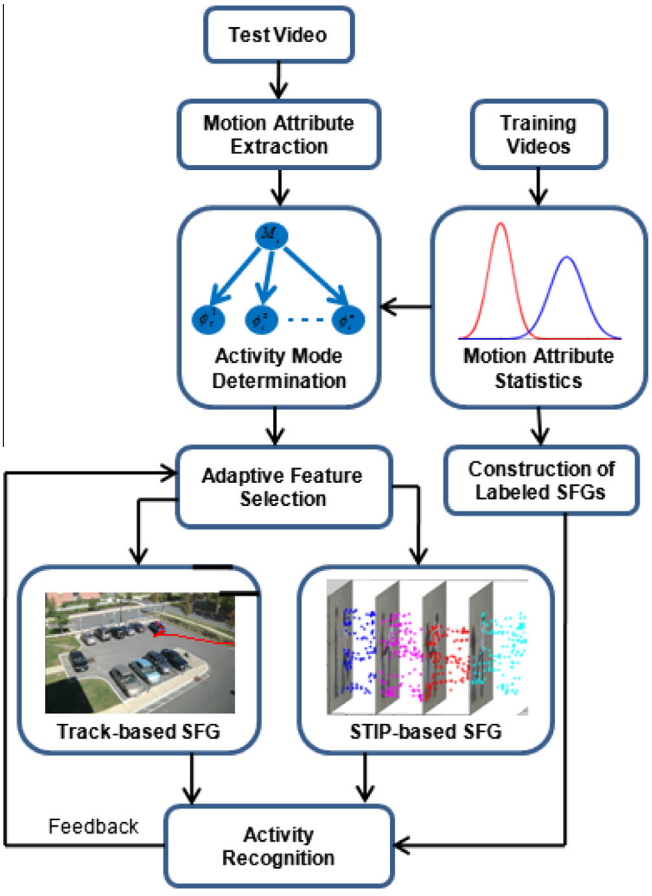

Fig. 7. Overall flow of the activity recognition system using adaptive feature selection in the SFG framework.

vector given the mode is $\prod_i p(\phi_i|M)$. Thus the mode M_t of pattern Φ_t is

$$M_t = \begin{cases} g & \text{if } \prod_i p\left(\phi_t^i|g\right) \cdot p(g) > \prod_i p\left(\phi_t^i|l\right) \cdot p(l) \\ l & \text{if } \prod_i p\left(\phi_t^i|g\right) \cdot p(g) < \prod_i p\left(\phi_t^i|l\right) \cdot p(l) \end{cases}. \quad (19)$$

We integrate the SFG method into the SDS model to realize automatic feature selection in activity recognition. Fig. 7 shows the overall flow of the proposed recognition system.

6. Experiments

In order to evaluate the efficacy of our method to recognize complex activities involving multi-object interactions, we conducted experiments on three state-of-the-art datasets containing long duration videos and a large scale of complex activities including UT-Interaction dataset [9], VIRAT dataset [10] and UCR VideoWeb activity dataset [11],

UT Interaction dataset [9] is composed of both segmented and unsegmented videos, and includes several pairs of interacting people simultaneously executing activities across different background, scale and illumination. The interaction activities which we looked at are shaking hands, hugging, pointing, punching, kicking and pushing. VIRAT dataset is a state-of-the-art activity dataset with many challenging characteristics, such as wide variation in the activities and clutter in the scene. It consists of surveillance videos of six parking lots with different scales of resolution. The activities in the dataset includes single vehicle activities, person and vehicle interactions, and people interactions. In this paper, we examine fourteen kinds of activities – global activities including vehicle u-turn, vehicle turn and vehicle backup, people walking together, people gathering, and people dispersion, and local activities including person loading an object to a vehicle, person unloading an object from a vehicle, person opening a vehicle trunk, person closing a vehicle trunk, person getting into a vehicle and person getting out of a vehicle. The portion of the UCR VideoWeb activity dataset [11] we work on (details can be obtained from the authors) involves up to 10 actors interacting in various ways with each other, vehicles and facilities. The activities were: people meeting, people following, vehicles turning, people dispersing, shaking hands, gesturing, waving, hugging and pointing.

In accordance with the motivation of the paper, we work in both classification-based and query-based activity recognition. In the classification-based scheme, testing instances are classified into predefined kinds of activities given multiple training instances of the same kinds using nearest neighbor classifier. The query-based scheme is based on an example video-based retrieval framework wherein the algorithm is provided with one (or, at most, a few, but not enough to build a classifier) video(s) depicting an action of interest. The aim is to retrieve videos which have similar activity as the query video(s) has.

6.1. Preprocessing

Object detection and tracking are performed first. We utilize the tracking method developed in [47] to obtain the trajectories of moving objects. Identifications of moving objects (person, vehicle, or others) are obtained using [48], and shadows are excluded by

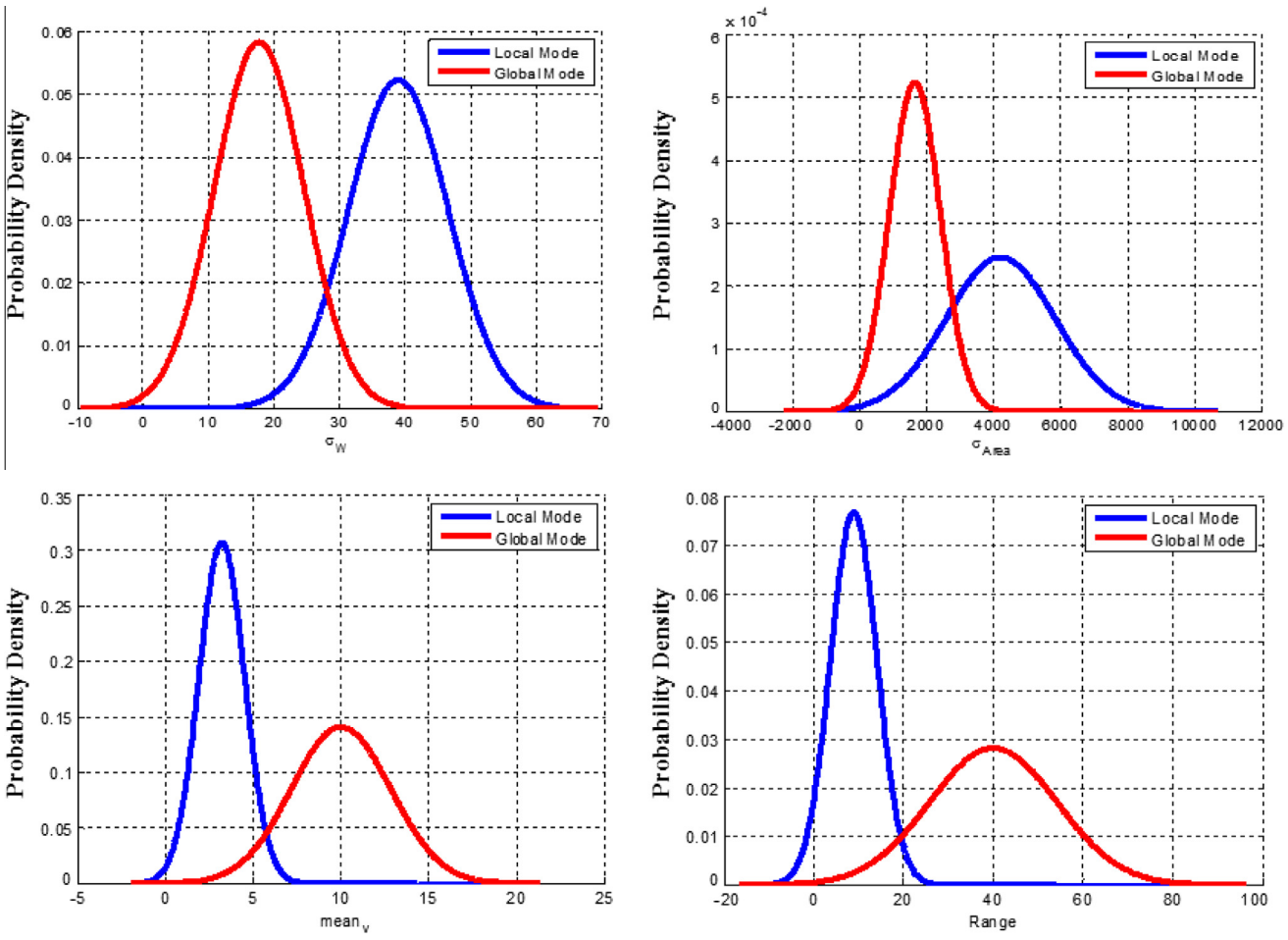

Fig. 8. Estimated conditional probability distributions of the four motion attributes given the activity mode.

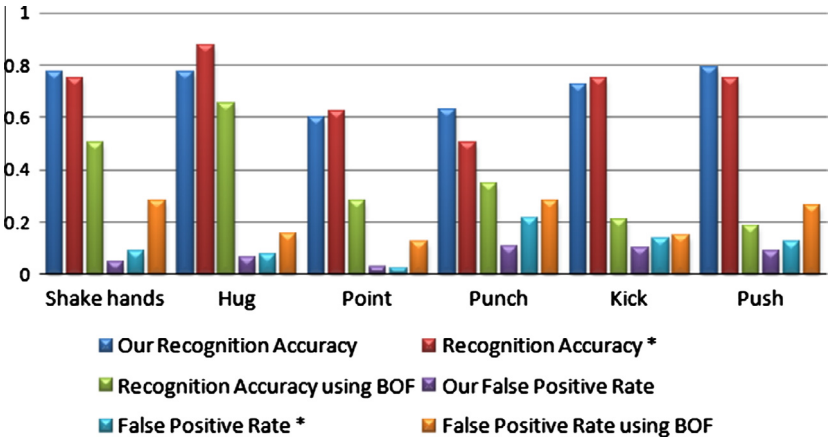

Fig. 9. Recognition accuracy on the UT-Interaction dataset by using voting scheme on top of SFG model.

using color histogram. Note that object identification is applied to each trajectory.

6.1.1. Preprocessing of tracks

A weighted moving average filter is applied to the raw tracks in order to smooth out the effect of local outliers on the global motion pattern. Observing that stopping is often the sign of the end of an global activity, we detect the stopping events on each track and segment long tracks accordingly. Each track segment is considered as a complete activity agent. A stopping is detected when the variance of the positions of the interesting objects within a temporal window is below a predefined threshold.

6.1.2. Adaptive feature selection

Background substraction in [47] is used to obtain the binarized silhouettes of moving objects and their bounding boxes. Informative attributes derived from the silhouettes and bounding boxes are used to specify the Naive Bayesian network discussed in Section 5.2.

Half of the dataset [10] parking lot 04 is used as training data to select the informative attributes, and the probability distribution of these attributes is obtained using Gaussian mixture model and Expectation Maximization. The selected track-based low resolution attributes are:

(1) Variance of width of bounding box σ_W.
(2) Variance of the area inside the silhouette of moving object, where $Area = H \times W$, H and W are the height and width of the bounding box.
(3) Average velocity of the underlying objects $mean_v$.
(4) Range of the track R, which is defined as $R = max[-max(x) - min(x), max(y) - min(y)]$.

The estimated conditional probability distribution of motion attributes given the activity mode is shown in Fig. 8.

Activity instances in the training data are labeled and segmented out, and SFGs are constructed for these instances. For the track-SFG, we use joint NDG–NMG and RD–SRD features. Tracks involved in local activities are often very short and have a small range. We consider it is unlikely that track-based low resolution feature can distinguish local activities. So, we do not train track-based SFGs for local activities.

For distance thresholds τ_n and τ_e in (1), we determined their optimal values experimentally based on labeled training data and used them in testing. The values of τ_n and τ_e used in the following experiments are 0.4 and 0 for STIP-based SFG and are zeros for track-based SFG. The values may be different in other scenarios.

6.2. STIP-based SFG results

In this part of experiments, we implement STIP-based SFG on UCR VideoWeb dataset, UT Interaction dataset, and VIRAT dataset.

6.2.1. Classification-based activity recognition results

The classification-based recognition performance of the proposed algorithm is first evaluated on the UT Interaction dataset [9]. In order to compare with previous systems, we use an experimental setting similar to [4], which proposed a supervised learning method for the same set of activities on this dataset. We randomly choose two among the ten videos of each class to form the training set and leave out the others for testing.

We verify that our system is able to recognize multiple complex activities from continuous videos. We were able to achieve high recognition scores and lower false positive rates. We compare our results with previous methods in Fig. 9. Our overall performance on the UT Interaction dataset is superior to Bag-of-Feature approach. Here the results of Bag-of-Feature approach are reported on segmented video clips, while our results and [4] are reported on continuous video. Our results are similar to those in [4] for some activities and better for others. However, our approach can use only a single query to perform recognition as demonstrated in Fig. 10 and hence has a wider generalizability. In [9], recognition results of several approaches are reported on the same dataset; the recognition accuracy is in the range from 0.49 to 0.88. Our performance is comparable to the best performance in [9]. Note that the experiment settings in [9] are slightly different from ours. Their results are reported by leaving one out among a set of ten for testing and using the other 9 for the training, and the videos are segmented, while we use 2 sets as labeled query videos and test on 8. Thus we are achieving results comparable to [9] with much less training data on continuous videos (a significantly harder problem).

In the next experiment, we verify the effectiveness of our system in correctly classifying activities in VIRAT dataset [10]. We work on the segmented video clips from the portion of VIRAT dataset for which annotation is available, because the evaluation of this experiment needs the ground truth of activities. We segmented out the video clips according to the annotation files. Each video clip contains only one execution of the activities of interest. There are forty-eight video clips in total, eight instances for each type of activity, in the whole dataset used in this experiment. The results are shown as a confusion matrix and ROC curves in Fig. 11.

6.2.2. Query-based retrieval results

We compute the DTW aligning cost between the query SFGs of the testing video and each query video containing a specific action

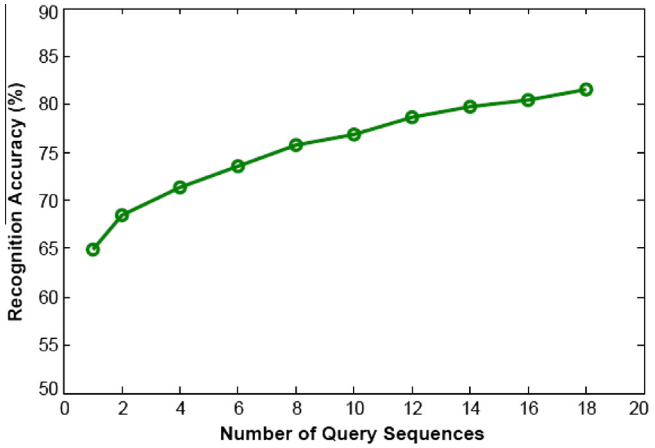

Fig. 10. The recognition accuracy of our method with respect to number of query examples. It can be seen that when number of query example decreases, the performance of our method does not drop significantly.

and count the instances that the DTW distance is less than a threshold. Based on this number (i.e., number of similar training videos), the system makes a decision on the recognized activity.

To evaluate the performance of the STIP-based SFG method in query-based retrieval, we first work with the UCR VideoWeb activity dataset [11]. We work with video clips from this dataset, report the best matches found by our system and accordingly present and analyze the accuracy/false positive rates.

We proceed by taking a small video clip depicting a complex activity and search the dataset for matches. The STIP features for the query and the dataset videos are computed. The query and dataset videos were uniformly segmented into temporal segments, the feature points in each segment forming a feature graph, and the string of time ordered graphs forming the SFG descriptor. The length of each segment is set to be 20 frames. Next we find the pair-wise correspondences between each of the feature collections from the query video with those of dataset videos using the spectral solution in Section 3.2.1. We finally perform the DTW matching across the entire query and dataset SFGs (composed of time ordered feature-graphs) based on the local match scores calculated.

The results from this experiment involving query-based activity video retrieval are shown in Fig. 12. For each activity class, we chose 3 random videos from the samples of that class to be the query. The results reported here are obtained by averaging across the 3 test cases. Recognition on activities like vehicle turning and shaking hands performed especially well since they continue for longer time periods and hence generate better feature points. On the other hand, activities such as "pointing" happen in a short amount of time and are thus more difficult to recognize. We found that the recognition results obtained based on a single sample video generate higher false positive rates. This is justifiable due the fact that in a single query-based retrieval framework, there is no statistically reliable way to set the acceptance threshold.

We also studied variation in recognition performance of our method with change in query videos. The standard deviation in the scores for different query-videos is marked in Fig. 12a. In line with our previous argument, short-duration activities such as "pointing" had higher variability. The activity "hug" was confused with background clutter or actors crossing each other.

We show some results of activity retrieval on UCR VideoWeb using one query video in Fig. 12b. The query videos are shown on the left and the other three columns show the top three best matches. The bounding boxes of the sub-graphs that best match the feature graphs of query video are also shown. This demonstrates the capability of our system in locating the activities of interest in the spatial-temporal video volumes.

Finally, we test our system on activity retrieval using one query video on UT Interaction dataset. Some results are shown in Fig. 13. The query videos are shown on the left and the other three columns show the top three best matches.

6.3. Track-based SFG results

In this set of experiments, we work on VIRAT dataset (parking lot 04) to evaluate the performance of our track-based SFG scheme. Activities whose motion pattern can be determined by the underlying tracks are of interest here. The activities of interest include 25 single vehicle activities (6 vehicle-backup, 13 vehicle-turn, and 2 vehicle u-turn), 9 people interactions (5 people dispersion, 2 people walking together, and 2 people gathering), and 29 people-vehicle interactions (15 people approaching vehicle and 14 people leaving vehicle). For object detection and tracking, we applied the methodology we have developed in [47].

6.3.1. Performance comparison among motion descriptors

In order to assess the performance of different motion descriptors of single tracks in activity recognition – NDG, NMG and joint NDG–NMG which concatenates NDG and NMG, we test our system on VIRAT Testing Dataset to classify single vehicle activities whose

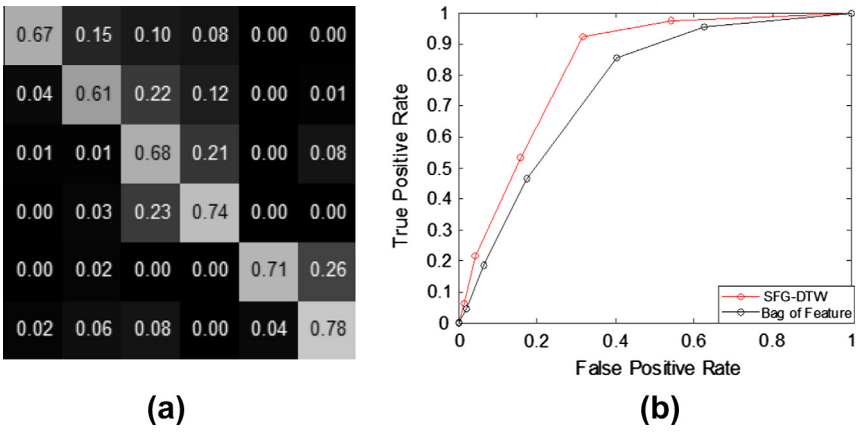

Fig. 11. (a) Confusion matrix for VIRAT dataset. Nearest neighbor classifier and eight-leave-one-out cross-verification are used. (1: person loading an object to a vehicle, 2: person uploading an object from a vehicle, 3: person opening a vehicle trunk, 4: person closing a vehicle trunk, 5: person getting into a vehicle, 6: person getting out of a vehicle). Most misclassifications are inside activity group (1, 2, 3, and 4) and activity group (5 and 6). (b) ROC of 7-NN classifier on VIRAT dataset (Bag of feature: [2]+7NN).

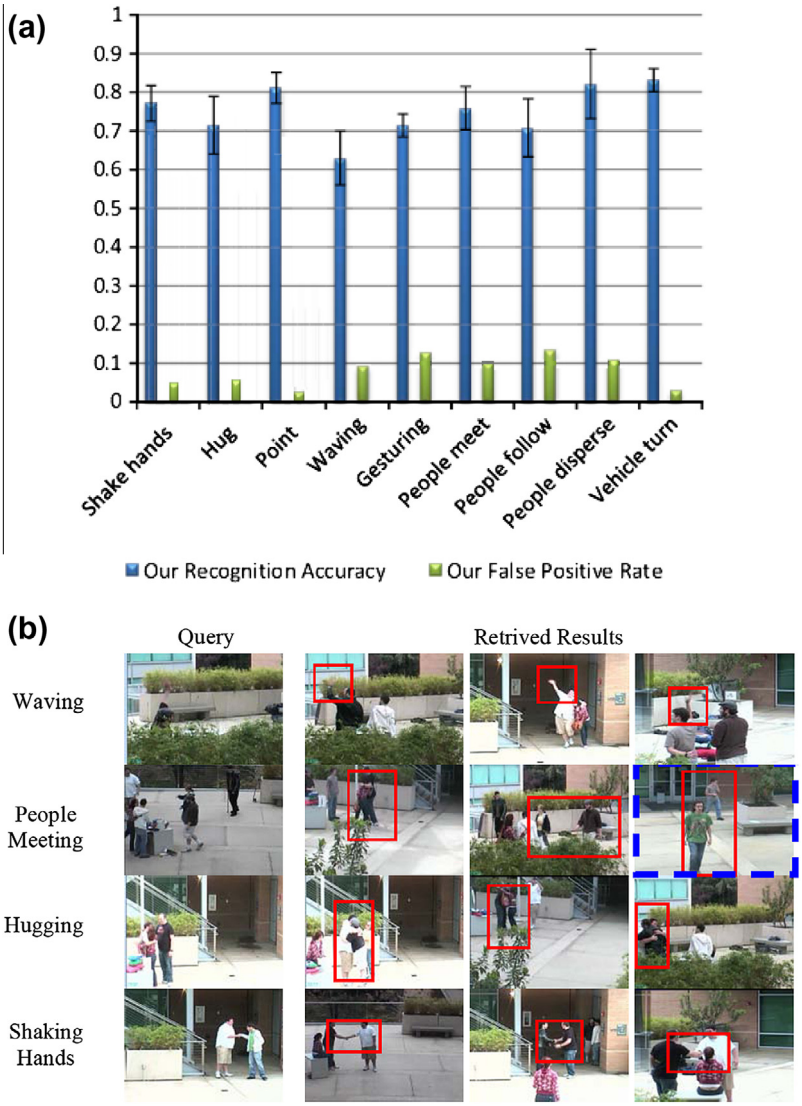

Fig. 12. (a) Recognition accuracy and false positives on 9 activities from the UCR VideoWeb dataset in a query-based retrieval framework. Standard deviation in performance (accuracy) for different queries is marked on the bars. (b) Retrieval results: The left column depicts the query videos and the other three columns are the best matches on UCR VideoWeb dataset. The bounding boxes of the sub-graphs that best match the feature graphs of the query video are shown. A blue dash box represents an incorrect match. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

characteristics only depend on features of individual tracks – vehicle-turn, vehicle-u-turn, and vehicle-backup. There are twenty-five instances of the above activities in total in videos from parking lot 04. Each instance is applied as the query alternatively. We search across the whole dataset for activities of the same type. The results reported are obtained by averaging across all the queries. From Fig. 14a we can conclude that NMG descriptor outperforms other two single track descriptors.

6.3.2. Classification-based results

In this part of experiments, we test the ability of the track-based SFG to classify both single object activities and interactions discussed in Section 6.3 with and without object identification as shown in Fig. 14b. The object identifier can tell whether the underlying object associate with a given track is a vehicle or a person. Joint descriptors NDG–NMG and RD–SRD are applied.

6.3.3. Query-based results

In order to demonstrate the effectiveness of our track-based SFG in retrieving activities, we search across videos of VIRAT dataset to find the tracks which match the query tracks. For each trial, underlying tracks of an interesting activity listed in 6.3 are the input, and the algorithm exhaustively searches across the video dataset to find the sets of tracks of the same kind, which matches to the query tracks. The results are shown in Fig. 15a.

Finally, we show samples of retrieved tracks in Fig. 15b. This demonstrates the capability of our track-based SFG system in locating the activities of interest in the spatial-temporal video volumes.

6.4. Adaptive feature selection

In this subsection, we implement the adaptive feature selection scheme on VIRAT dataset, and compare the result with schemes without feature selection. Encouraging results are shown, demonstrating the efficacy of our adaptive feature selection to recognize complex activities with increased recognition accuracy and reduced computation complexity.

For the entire test video, we first compute the low resolution motion attributes. These attributes are used to detect the location of activities and to decide the mode of each activity. Whenever an activity is detected, the optimum feature type is selected based on

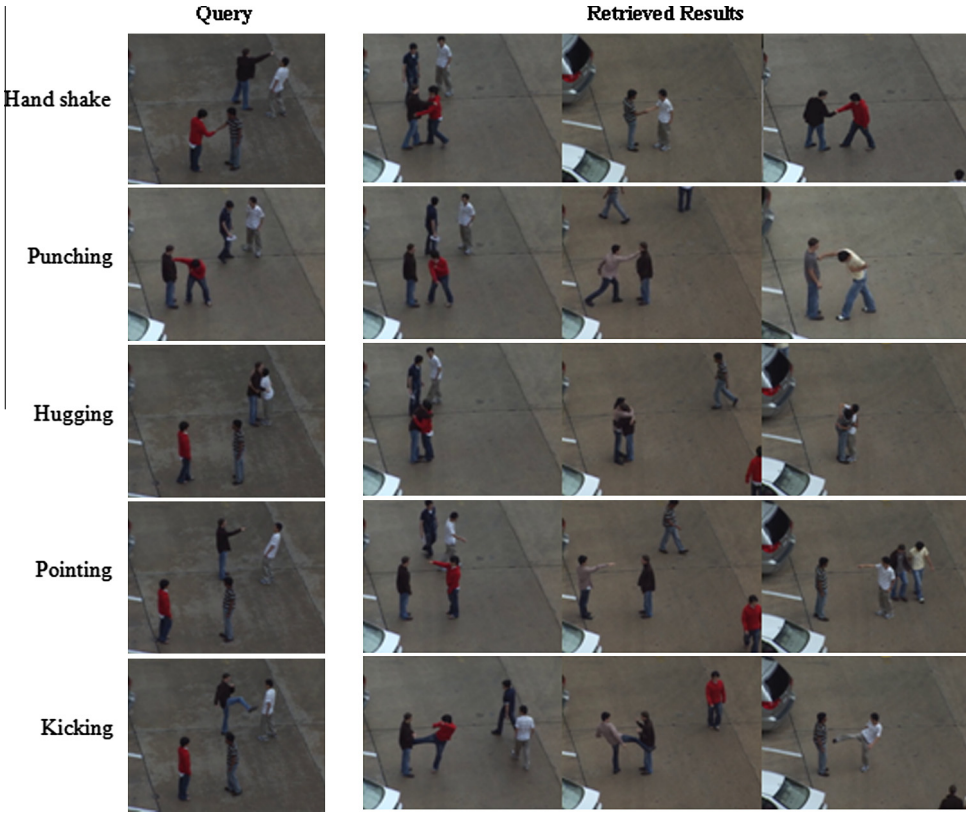

Fig. 13. Retrieval results: The left column depicts the query videos and the other three columns are the best matches on UT-Interaction dataset.

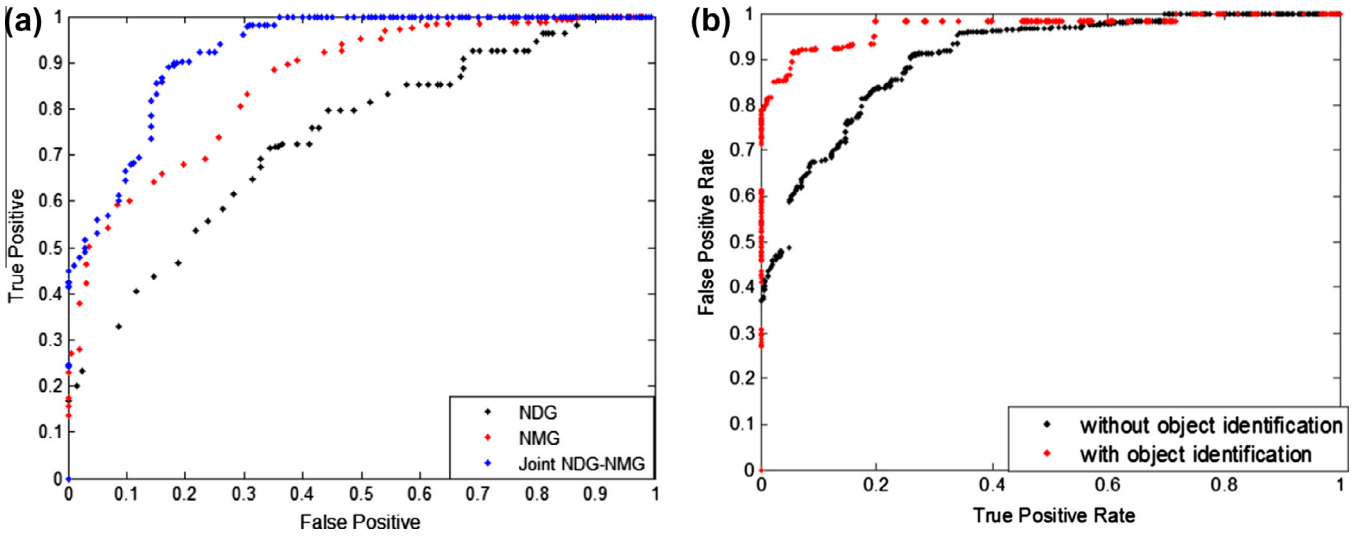

Fig. 14. (a) Average performance of different kinds of motion descriptors on recognizing activities characterized by individual tracks. (Parameter of ROC: the same distance threshold is used for all activities). (b) Average performance of track-based SFG system on VIRAT Testing Dataset. For each run, only one training instance is used for each kind of activity, the rest are treated as testing instances. While the algorithm achieves high recognition performance, the object identifer further enhances the performance. Joint NDG–NMG and joint RD–SRD descriptors are used.

the activity mode and a SFG is constructed on these features. The developed SFG is matched to the training SFGs using a voting scheme. Fig. 16 shows the switching scheme and recognized activities for one video.

Experimental analysis shows that a few simple heuristics can improve the performance of the method further. One is related to identifying regions where the track-based features do not perform very well, in spite of being chosen by the switching scheme. For this reason, our system identifies when the track-based recognition has low confidence and switches to the STIP-based mode. The track-based results are considered as unreliable when the similarity scores between the testing instance and *all* the training instances are low. It is based on the fact that STIP-based features can recognize both global and local activities, but track-based SFGs can recognize only global activities. A second case arises at the beginning and end of track segments. Experience suggests that local activities usually happen in these regions. Therefore, to minimize the chance of missing a local activity, we analyze the

(a)

	Precision	Recall	Total Fetched	True Positive	Ground Truth
Car U-turn	0.67	1	3	2	2
Car Turn	0.90	1	19	17	17
Car Backup	0.75	1	8	6	6
People Approaching Vehicle	0.86	0.93	16	14	15
People Leaving Vehicle	1	1	14	14	14
People Walking Together	1	1	2	2	2
People Meeting	1	1	2	2	2
People Dispersion	1	1	5	5	5

(b)

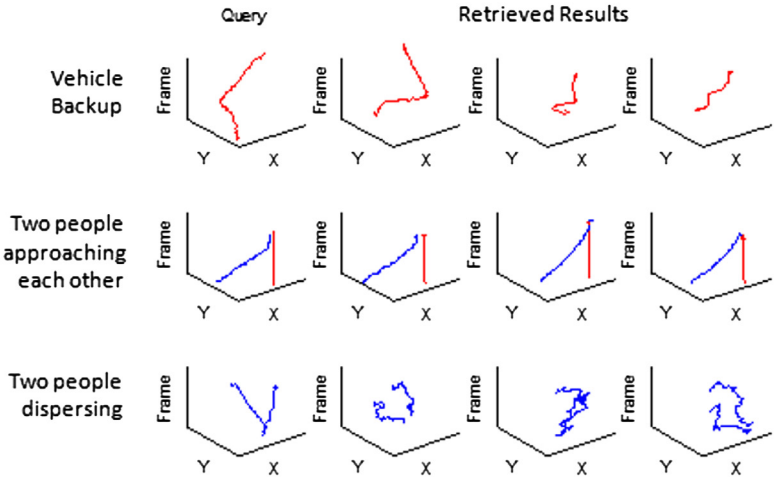

Fig. 15. (a) Average performance of track-based SFG system on VIRAT Testing Dataset with object identifier. Only one query instance is used for each query. (b) Examples of query results on VIRAT testing Dataset. The left column depicts the query tracks involved in the targeted activity and the other three columns are the best matches on part of VIRAT testing dataset.

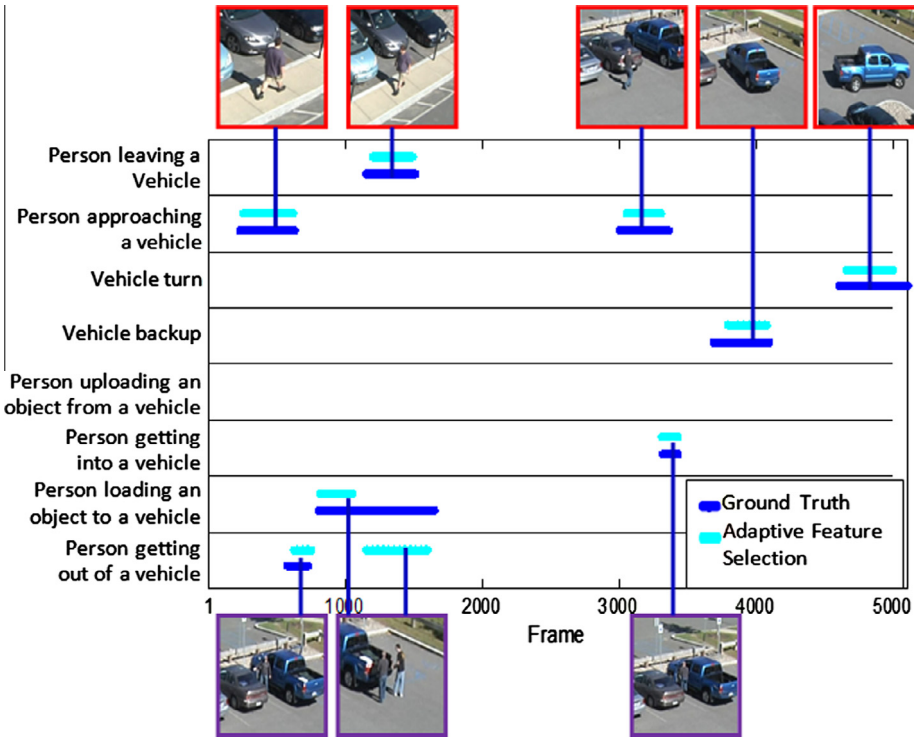

Fig. 16. Switching results on an example video sequence are shown. One sample image for each activity is shown. Each cyan bar in the figure indicates the recognition result from the adaptive feature selection and compares it to the ground truth (blue bar). The length of the bar indicates the duration of the recognized activity. Red bounding box indicates track features are selected, and purple indicates STIP features are selected for the SFG model in adaptive feature selection. The results show that the system is able to automatically switch between different features. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
Results of adaptive feature selection. (Recognition accuracy on the VIRAT dataset by using fixed types of feature and adaptive feature selection. N/A means that the activity cannot be recognized by the corresponding feature.)

Activity	Recognition accuracy		
	Track feature	STIP feature	Adaptive feature selection
Person loading an object to a vehicle	N/A	0.49	0.55
Person unloading an object from a vehicle	N/A	0.42	0.51
Person opening a vehicle trunk	N/A	0.54	0.57
Person closing a vehicle trunk	N/A	0.64	0 68
Person getting into a vehicle	N/A	0.51	0.65
Person getting out of a vehicle	N/A	0.63	0.72
Vehicle u-turn	0.85	0.50	0.90
Vehicle turn	0.9	0.57	1
Vehicle backup	1	0.73	1
People approaching a vehicle	0.93	N/A	0.93
People leaving a vehicle	1	N/A	1
People walking together	0.9	0.65	1
People gathering	1	0.45	1
People dispersion	1	0.53	1

beginning and end of track segments that are not already identified by the switching module to detect if there are any local activities happening there.

Table 1 gives the recognition results for each type of activity using different types of features. From the results, we can see that features of different resolution can only recognize certain types of activities. Track-based low resolution features work better at recognizing global activities while STIP features work better at recognizing local activities. The proposed adaptive feature selection improves the recognition accuracy while reducing the overall computation complexity.

6.4.1. Computation complexity

Table 2 shows the computation time of the entire activity recognition process including the training process. As discussed before, algorithms based on high resolution features are often time consuming. In our algorithm, the most time-consuming part is the graph matching. Assuming the number of nodes of the two graphs to be matched is n_Q and n_P, computational complexity of the graph matching in [36] is $O((n_Q n_P)^{\frac{2}{3}} + (max(n_Q, n_P) - \frac{1}{2}) \cdot min^2(n_Q, n_P))$ [36]. For a feature graph of the same time interval, the number of local features is of the order of tens of times the number of global features. Over a long period of time, this difference in computation can be large.

7. Conclusion

In this work, we argued that spatio-temporal relationships are critical to discriminate real-world activities. We proposed a feature model based on string representation of the video which respects the spatio-temporal dynamics of the complex activities. In order to quantize the similarity of two feature graphs, we leveraged a spectral matching technique to find correspondences between them. Finally, the string formed by the time-ordered set of local feature collections was matched with other strings in a dynamic programming framework to obtain the matching score. This matching score was used to classify a test video as being similar or non-similar to the template video. We show how the SFG can be constructed for high-resolution STIP features and low-resolution track features. To accelerate the matching process while enhancing the recognition accuracy, the proposed SFG algorithm is integrated into a scheme of adaptive feature selection which automatically chooses features for the recognition task based on the states of activities. Our experiments demonstrated the effectiveness of our approaches to successfully recognize and localize complex activities even with multiple interacting actors.

Table 2
Comparison of computation complexity. (Note that the computation time is given as the approximate percentage of total computation time using STIP features only. Training overhead includes the time used to construct the training SFGs from the labeled and segmented video clips for track-based and stip-based algorithm, plus the attribute space construction time for adaptive feature selection algorithm.)

Feature type	Training overhead (%)	Feature extraction during testing (%)	Recognition time (%)
Track feature	3	6	4
STIP feature	20	35	45
Adaptive feature selection	8	17	12

Acknowledgements

This work has been partially supported by NSF grant IIS-0712253, the DARPA VIRAT program and a subcontract from Mayachitra Inc., through DARPA STTR award W31P4Q-11-C0042.

References

[1] P.A. Anderson, Nonverbal Communication: Forms and Functions, second ed., Waveland Press, 2008.
[2] C. Schuldt, I. Laptev, B. Caputo, Recognizing human actions: a local SVM approach, in: Intl. Conf. on Pattern Recognition, 2004.
[3] L. Gorelick, M. Blank, E. Shechtman, M. Irani, R. Basri, Actions as space-time shapes, IEEE Trans. Pattern Anal. Mach. Intell. (2007).
[4] M.S. Ryoo, J.K. Aggarwal, Spatio-temporal relationship match: video structure comparison for recognition of complex human activities, in: IEEE Intl. Conf. on Computer Vision, 2009. doi: 10.1109/ICCV.2009.5459361.
[5] S. Park, J.K. Aggarwal, Recognition of two-person interactions using a hierarchical Bayesian network, ACM J. Multimedia Syst. (2004) (Special Issue on Video Surveillance).
[6] M.S. Ryoo, J.K. Aggarwal, Recognition of composite human activities through context-free grammar based representation, in: IEEE Conf. on Computer Vision and Pattern Recognition, 2006.
[7] Y.A. Ivanov, A.F. Bobick, Recognition of visual activities and interactions by stochastic parsing, IEEE Trans. Pattern Anal. Mach. Intell. (2000).
[8] S. Tran, L.S. Davis, Visual event modeling and recognition using Markov logic networks, in: Euro. Conference on Computer Vision, 2008.
[9] M.S. Ryoo, C.-C. Chen, J.K. Aggarwal, A. Roy-Chowdhury, An overview of contest on semantic description of human activities (SDHA), in: Intl. Conf. on Pattern Recognition, 2010.
[10] S. Oh, A. Hoogs, A. Perera, N. Cuntoor, C.-C. Chen, J.T. Lee, S. Mukherjee, J. Aggarwal, H. Lee, L. Davis, E. Swears, X. Wang, Q. Ji, K. Reddy, M. Shah, C. Vondrick, H. Pirsiavash, D. Ramanan, J. Yuen, A. Torralba, B. Song, A. Fong, A. Roy-Chowdhury, M. Desai, A large-scale benchmark dataset for event recognition in surveillance video, in: IEEE Conf. on Computer Vision and Pattern Recognition, 2011.
[11] G. Denina, B. Bhanu, H. Nguyen, C. Ding, A. Kamal, C. Ravishankar, A. Roy-Chowdhury, A. Ivers, B. Varda, Videoweb dataset for multi-camera activities and non-verbal communication, in: Distributed Video Sensor Networks, Springer, 2011.
[12] N. Nayak, R. Sethi, B. Song, A. Roy-Chowdhury, Motion pattern analysis for modeling and recognition of complex human activities, in: Guide to Video Analysis of Humans: Looking at People, Springer, 2011.
[13] P. Turaga, R. Chellappa, V.S. Subrahmanian, O. Udrea, Machine recognition of human activities: a survey, in: IEEE Trans. on Circuits and Systems for Video Technology, 2008.
[14] C. Fanti, L.Z. Manor, P. Perona, Human motion: modeling and recognition of actions and interactions, in: Proceedings of the 2nd International Symposium on 3D Data Processing, Visualization, and Transmission, 2005.
[15] F. Chen, W. Wang, Activity recognition through multiscale dynamic Bayesian network, in: 16th International Conference on Virtual Systems and Multimedia, 2010.
[16] Z. Sun, S.S. Ge, Stability Theory of Switched Dynamical Systems, first ed., Springer, 2011.
[17] U. Guar, Y. Zhu, B. Song, A.K. Roy-Chowdhury, A "string of feature graphs" model for recognition of complex activities in natural videos, in: IEEE Intl. Conf. on Computer Vision, 2011.
[18] A. Gupta, L.S. Davis, Objects in action: an approach for combining action understanding and object perception, in: IEEE Conf. on Computer Vision and Pattern Recognition, 2007.
[19] P. Dollar, V. Rabaud, G. Cottrell, S. Belongie, Behavior recognition via sparse spatio-temporal features, in: Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005.
[20] J.C. Niebles, H. Wang, L.F. Fei, Unsupervised learning of human action categories using spatial-temporal words, Int. J. Comput. Vision (2008).

[21] R. Chaudhry, A. Ravichandran, G. Hager, R. Vidal, Histograms of oriented optical flow and Binet–Cauchy kernels on nonlinear dynamical systems for the recognition of human actions, in: IEEE Conf. on Computer Vision and Pattern Recognition, 2009.
[22] U. Gaur, B. Song, A.K. Roy-Chowdhury, Query-based retrieval of complex activities using "strings of motion-words", in: IEEE Workshop on Motion and Video Computing, 2009.
[23] S. Savarese, A. DelPozo, J.C. Niebles, L. Fei-Fei, Spatial-temporal correlations for unsupervised action classification, in: IEEE Workshop on Motion and Video Computing, 2008.
[24] A. Galata, A. Cohn, D. Magee, D. Hogg, Modeling interaction using learnt qualitative spatio-temporal relations and variable length markov models, in: Proceedings of the European Conference on Artificial Intelligence, 2002.
[25] S. Park, J. Aggarwal, Recognition of two-person interactions using a hierarchical bayesian network, in: ACM SIGMM International Workshop on Video Surveillance, 2003.
[26] M. Lee, R. Nevatia, Human pose tracking in monocular sequence using multilevel structured models, IEEE Trans. Pattern Anal. Mach. Intell. (2009).
[27] D. Kuettel, M. Breitenstein, L.V. Gool, V. Ferrari, What's going on? Discovering spatio-temporal dependencies in dynamic scenes, in: IEEE Conf. on Computer Vision and Pattern Recognition, 2010.
[28] M.S. Ryoo, W. Yu, One video is sufficient? Human activity recognition using active video composition, in: IEEE Workshop on Motion and Video Computing, 2011.
[29] E. Shechtman, M. Irani, Matching local self-similarities across images and videos, in: IEEE Conf. on Computer Vision and Pattern Recognition, 2007.
[30] H.J. Seo, P. Milanfar, Detection of human actions from a single example, in: IEEE Intl. Conf. on Computer Vision, 2009.
[31] M. Marszalek, I. Laptev, C. Schmid, Actions in context, in: IEEE Conf. on Computer Vision and Pattern Recognition, 2009.
[32] J. Liu, S. Ali, M. Shah, Recognizing human actions using multiple features, in: IEEE Conf. on Computer Vision and Pattern Recognition, 2008.
[33] V. Pavlovic, J. Rehg, J. MacCormick, Learning switching linear models of human motion, Neural Inform. Process. Syst. Found. (2000).
[34] J.C. Nascimento, M.A. Figueiredo, J. Marques, Recognition of human activities using space dependent switched dynamical models, in: Intl. Conf. on Image Processing, 2005.
[35] J. Kittler, M. Hatef, R. Duin, J. Matas, On combining classifiers, IEEE Trans. Pattern Anal. Mach. Intell. (1998).
[36] M. Leordeanu, M. Hebert, A spectral technique for correspondence problems using pairwise constraints, in: IEEE Intl. Conf. on Computer Vision, 2005.
[37] H. Sakoe, S. Chiba, Dynamic programming algorithm optimization for spoken word recognition, IEEE Trans. Acoust., Speech Signal Process. (1978).
[38] M. Müller, Information Retrieval for Music and Motion, Springer Verlag, 2007.
[39] N. Oliver, B. Rosario, A. Pentland, A bayesian computer vision system for modeling human interactions, IEEE Trans. Pattern Anal. Mach. Intell. 22 (2000) 831–843.
[40] U. Gaur, B. Song, A.K. Roy-Chowdhury, Query-based retrieval of complex activities using strings of motion-words, in: IEEE Workshop on Motion and Video Computing, 2009.
[41] R.J. Sethi, A.K. Roy-Chowdhury, Modeling and recognition of complex multi-person interactions in video, in: Multimodal Pervasive Video Analysis, 2010.
[42] Y.-G. Jiang, G.-W. Ngo, J. Yang, Towards optimal bag of word for object categorization and semantic video retrieval, in: ACM Intl. Conf. on Image and Video Retrieval, 2007.
[43] I. Laptev, On space-time interest points, in: International Journal of Computer Vision, 2005.
[44] J.K. Aggarwal, S. Park, Human motion: modeling and recognition of actions and interactions, in: Proceedings of the 2nd International Symposium on 3D Data Processing, Visualization, and Transmission, 2004.
[45] S. Prajna, A. Jadbabaie, Safety verification of hybrid systems using barrier certificates, in: Hybrid Systems: Computation and Control, Lecture Notes in Computer Science, vol. 2993, Springer, Berlin/Heidelberg, 2004, pp. 271–274.
[46] O. Ayad, M. Sayed-Mouchweh, P. Bllaudel, Switched hybrid dynamic systems identification based on pattern recognition approach, in: IEEE Intl. Conf. on Fuzzy Systems, 2010.
[47] B. Song, T. Jeng, E. Staudt, A. Roy-Chowdury, A stochastic graph evolution framework for robust multi-target tracking, in: Euro. Conference on Computer Vision, 2010.
[48] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, Discriminatively trained deformable part models, release 4, 2010. <http://people.cs.uchicago.edu/pff/latent-release4/>.