Discriminative Key-Component Models for Interaction Detection and Recognition

Yasaman S. Sefidgar[a], Arash Vahdat[a], Stephen Se[b], Greg Mori[a]

[a] *Vision and Media Lab, School of Computing Science, Simon Fraser University Burnaby, BC, Canada*
[b] *MDA Corporation, Richmond, BC, Canada*

Abstract

Not all frames are equal – selecting a subset of discriminative frames from a video can improve performance at detecting and recognizing human interactions. In this paper we present models for categorizing a video into one of a number of predefined interactions or for detecting these interactions in a long video sequence. The models represent the interaction by a set of key temporal moments and the spatial structures they entail. For instance: two people approaching each other, then extending their hands before engaging in a "handshaking" interaction. Learning the model parameters requires only weak supervision in the form of an overall label for the interaction. Experimental results on the UT-Interaction and VIRAT datasets verify the efficacy of these structured models for human interactions.

Keywords:
video analysis, human action recognition, activity detection, machine learning

1. Introduction

We propose representations for the detection and recognition of interactions. We focus on surveillance video and analyze humans interacting with each other or with vehicles. Examples of events we examine include people embracing, shaking hands, or pushing each other, as well as people getting into a vehicle or closing a vehicle's trunk.

Detecting and recognizing these complex human activities is non-trivial. Successfully accomplishing these tasks requires robust and discriminative activity representations to handle occlusion, background clutter, and intra-class variation. While these challenges also exist in single person activity analysis, they are intensified for interactions. Furthermore, in surveillance applications, where events tend to be rare occurrences in a long video, we must have representations that can be used efficiently.

To address the above challenges, we represent an interaction by first decomposing it into its constituent objects (human-human or human-object), and then establishing a series of "key" components based on them (Figures 1 and 2). These key-components are important spatio-temporal elements that are useful for discriminating interactions. They can be distinctive times in an interaction, such as the period over which a person opens a vehicle door. We specifically refer to such important temporal components as *key-segments*. We further use *key-pose* to refer to a distinctive pose taken by an individual person involved in an interaction. For instance, a *key-pose* could be the outstretched arms of a person performing a push.

Our models describe interactions in terms of ordered key-components. They capture the temporal and spatial structures present in an interaction, and use them to extract the most relevant moments in a potentially long surveillance video. The spatio-temporal locations of these components are inferred in a latent max-margin structural model framework.

Context has proven effective for activity recognition. As Marszałek et al. [28] observed, identifying the objects involved in the context of an activity improves performance. A number of approaches (e.g. [15, 20, 23, 33]) examine the role of objects and their affordances in providing context for learning to recognize actions. Our approach builds on this line of work. We focus on surveillance video, where events are rare, and beyond the presence of contextual objects, spatio-temporal relations between the humans/objects are of primary importance. We contribute a key-component decomposition method that explicitly accounts for the relations between the humans/objects involved in an interaction. Further, we show that this approach permits efficient detection in a surveillance video, focusing inference on key times and locations where human interactions are highly likely.

Moreover, our discrete key-component series capture informative cues of an interaction, and are consequently compact and robust to noise and intra-class variation. They account for both temporal ordering and dynamic spatial relations. For example, we can account for spatial relationships between objects by simply characterizing their distance statistics. Alternatively, we can directly model the dynamics of relative distance over time in the video sequence.

Structured models of interactions can be computationally intensive. Our key-component model allows efficient candidate generation and scoring by first detecting the relevant objects, and then picking the pairs that are likely to contain an interaction.

We emphasize the importance of leveraging different structural information for effective interaction representation. In

Figure 1: Schematics of the *key-segment* model for interaction detection. Key-segments, enclosed by magenta outline, identify the most representative parts of the interaction. Spatial relations are captured through low-level features derived from distance and relative movement.

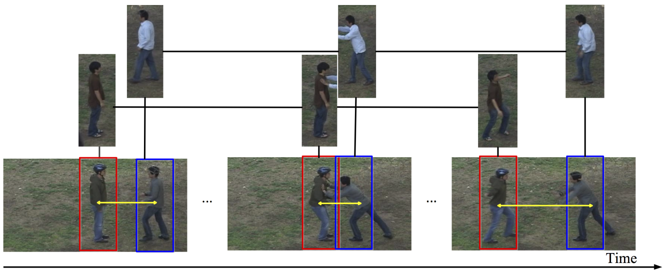


Figure 2: Schematics of the *key-pose* model for interaction recognition. An interaction is represented by a series of key-poses (enclosed by red or blue bounding boxes) associated with the discriminative frames of the interaction. Spatial distance, marked by yellow double-headed arrows, is explicitly modeled over time.

contrast, a common approach is to aggregate appearance and motion cues across the whole interaction track, ignoring potentially informative temporal and spatial relations [40, 30]. While these globally constructed representations can successfully distinguish a person jumping vs. a person walking, they are too simple to differentiate a person merely passing by a vehicle vs. a person getting in/out of it. The two share very similar appearance and motion patterns and a clear distinction becomes possible with the help of structural considerations (e.g. relative object distance and movements).

This paper extends our previous work [43]. We conduct extended experiments on efficient interaction detection and recognition, confirming the advantages of both object decomposition [43] and modeling of the temporal progression of key-components [29, 35] that are spatially related [43]. More specifically our contributions are: 1) efficient localization of objects involved in an interaction while accounting for interaction-specific motion and appearance cues, and 2) modeling of chronologically ordered key-components in a max-margin framework that explicitly or implicitly incorporates objects' relative distance and/or movements.

An overview of this paper is as follows. We review the related literature in Section 2. We then outline our approach to interaction representation in Section 3 and subsequently provide a detailed description of our models for detection (Section 4) and recognition (Section 6). We present empirical evaluation on the efficacy of the proposed representations for each task separately in Sections 5 and 7. We conclude and highlight possible future directions in Section 8.

2. Background

Activity understanding is a well-studied area of computer vision. To situate our research on detecting and recognizing interactions, we first clarify the distinction between these two tasks. We then highlight major trends in handling activity structures. A more comprehensive review of the literature on activity understanding in computer vision can be found in recent survey papers [48, 1, 34].

2.1. Detection vs. Recognition

In a recognition problem, the goal is to determine the type of an activity contained in an input video. That is, we implicitly assume something happens in the video. On the other hand, in detection we are concerned with finding the temporal and spatial location of an activity – crucially, with no prior knowledge on whether or not the input video contains an activity. The detection problem is thus inherently more challenging and computationally demanding as we should both classify the activities vs. non-activities, and specify when and where they occur. A feasible solution requires an efficient initial screening to narrow down the search space. It is common to use techniques such as background subtraction to segment regions of video where objects are moving. An activity model is then applied to these regions in a sliding window fashion [17, 4]. The main limitation of this approach is that the segmentation is not informed by knowledge about the activities we are searching for. Consequently, in the crowded scenes typically encountered in realistic video footage, we end up searching through many irrelevant regions. In our work on interaction detection, we instead identify regions that contain people and objects within a reasonable distance, and only search through these areas where it is highly likely for interactions to occur.

2.2. Structures in Activity Representation

A differentiating aspect in approaches to activity understanding is the incorporation of structural representations. There are two major questions to guide our classification of the literature: *what* sort of structures are deemed relevant, and *how* they are included in the representation. In the following subsections we review the four most significant classes of approach to modeling structures for detecting/recognizing activities.

2.2.1. No Structure

Typically, local low level features of appearance and/or motion over the entire video volume are aggregated in a histogram representation. Therefore, neither temporal nor spatial structure is considered. For example, Schüldt et al. [40] extract motion patterns corresponding to "primitive events" and capture their relevant appearance and motion information as spatio-temporal jets. They cluster these local descriptors to construct a vocabulary of the primitive elements, which is then used to obtain Bag-of-Words (BoW) representations of videos. Similarly, Niebles et al. [30] identify spatially discriminative regions that undergo complex motions and characterize the regions with a gradient descriptor. They represent a video sequence as a collection of words of a vocabulary constructed based on these descriptors. The expressiveness of these BoW representations is limited as they discard potentially discriminative structural information.

2.2.2. Spatial Structure

Similar to part-based object representations in still images, the spatial configuration of "parts" can be modeled on top of low level appearance and/or motion features. Wang and Mori [47] propose a frame level hidden part model based on local motion features. They process a video sequence frame-by-frame using their model and carry out majority voting to identify the video content. Tian et al. [42] developed a deformable part model that organizes discriminative parts over time based on their local appearance and motion captured by HOG3D features [21]. Although capturing spatial structure is sufficient for distinguishing activities consisting of parts with considerably different appearance, it fails to differentiate patterns with similar parts in different temporal order.

2.2.3. Temporal Structure

Sequential. The temporal progression of an activity can be captured by a series of hidden states inferred from appearance and/or motion observations. For example, Yamato et al. [50] develop a Hidden Markov Model (HMM) of an activity that observes a sequence of appearance symbols over the video frames. Once tuned to a particular type of activity, the model assigns higher probabilities to a sequence of symbols that more closely match the learned activity. Lv and Nevatia [27] perform key pose matching with sequence alignment via Viterbi decoding. Tang et al. [41] extend HMMs to also model the duration of each state in the temporal evolution of activities. These models are robust to time shifts as well as time variance in the execution of activities. However, they lack information about the spatial structure. This spatial structure can be crucial for making decisions, for example understanding whether a motion comes from the upper or lower body, or whether two parts meet or miss each other in a relative motion.

Local feature. Efforts have been made to enhance local feature methods by including spatio-temporal structural relations. Ryoo and Aggarwal [38] develop a kernel for comparing spatio-temporal relationships between local features and show effective classification in an SVM framework. Kovashka and Grauman [24] consider higher-order relations between visual words, discriminatively selecting important spatial arrangements. Yao et al. [51] utilize a local feature-based voting procedure to recognize actions. Yu et al. [52] propose an efficient recognition procedure using local features in a spatio-temporal kernelized forest classifier.

Exemplar. The temporal composition of an activity can be characterized by a series of templates on top of low level features. The template series are sometimes very rigid with little provision for variation in the length of an activity. For example, Efros et al. [11] construct a motion descriptor on every frame of a stabilized track and compute its cross-correlation matching score with samples of an activity database. The best matched sample represents the content of the track. Brendel and Todorovic [4] propose a more flexible model that builds exemplars by tracking regions with discriminative appearance and motion patterns. A general limitation of the exemplar models of temporal content is their insufficient generalization to samples that are not close enough to any of the templates.

Key-component. An activity can be represented as a discrete sequence of discriminative components based on appearance and/or motion features. Niebles et al. [29] identify a sequence of key components that are based on pooled HOG [7] and HOF [8] features at interest points. Raptis and Sigal [35] develop an even more compact representation by modeling frame level key poses that are automatically constructed as a collection of poselets. These models are highly robust to noise and intra-class variations. However, they do not exploit important discriminative spatial relations that are particularly relevant to interactions.

2.2.4. Temporal and Spatial

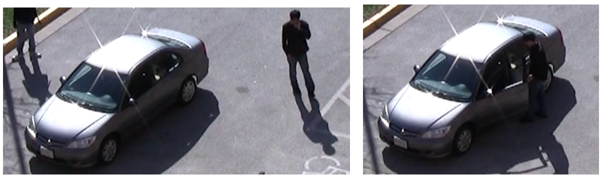
Leveraging both the temporal and spatial composition of activities gives models additional expressive power. Intille and Bobick [16] manually identify "atomic" elements of an activity and specify temporal and spatial relations among them to represent activities, such as a football play, that involve several people interacting with each other. Vahdat et al. [43] present a key-pose sequence model that automatically determines the informative body poses of people participating in an interaction while accounting for the temporal ordering of poses as well as their spatial relations and the roles people assume in the interaction. Methods have been developed that model sophisticated spatio-temporal relations between multiple actors / objects in a scene [2, 6, 25, 18]. In this paper we instead focus on models capturing detailed information about a pair of objects interacting in surveillance environments that lack the strong scene-context relationships that provide much of the benefit for the multi-actor models.

3. Analyzing Human Interactions

Given a surveillance video, our goal is to automatically detect/recognize activities that involve people interacting with objects or with other people. The overall flow of our approach is to first detect and track objects (people and/or vehicles). We then determine which object pairs are likely involved in an interaction. We apply more detailed models to these pairs

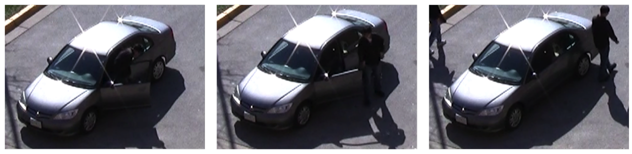
(a) Shaking hands (b) Hugging

Figure 3: People's relative distance changes depending on the type of interaction they participate in. People hugging each other are closer than people shaking hands.

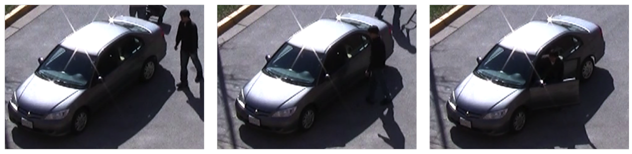


(a) No interaction (b) Getting into a vehicle

Figure 4: People are close enough to reach the objects they are interacting with.



(a) Getting out of a vehicle



(b) Getting into a vehicle

Figure 5: Relative movements of people and objects can distinguish between different interactions.

to find interactions. The initial screening enhances the overall efficiency as it considerably diminishes the search space. We develop methods for analyzing key-segments and key-poses within these pairs of tracks. Depending on the level of visual detail and interaction category granularity, the key-segment or more detailed key-pose model can be deployed.

An important aspect of our model is the selection of discriminative parts of a track. Given tracks of people and objects, we model their interaction as a series of locally discriminative components. We consider these components as latent variables in our model and infer them based on objects' appearance and their interrelations.

More specifically, we note that the objects involved in an interaction have discriminative relative distance and movement patterns. For example, two people's spatial distance when shaking hands is different from their proximity when hugging each other. Similarly, a person interacting with an object, such as a vehicle, is close enough to reach the object – a condition not necessarily true when there is no interaction going on (Figures 3 and 4). Moreover, people's movements with respect to an object are relevant. When a person gets into a car, her/his movements are toward the vehicle, while getting out of a car largely involves movements away from it (Figure 5). In subsequent sections we provide the details of our feature representations.

In the most naive approach, it is possible to feed appearance and relative distance/movement features pooled over an entire interaction track into a classifier (e.g. an SVM). However, this confounds relevant and irrelevant features of the track. Additionally, almost all informative structural information is washed out in this global representation. Instead, we leverage spatial and temporal structures and represent an interaction in terms of its most discriminative parts. By incorporating the most pertinent information, our representation can handle intra-class variation due to differences in the execution of the same interaction. For example, it is sufficient to find two nearby people with arms first alongside their bodies at one point in time and then concurrently extended toward each other at another point to reliably identify that they are shaking hands. Neither occlusion/clutter present at any other point, nor the time duration of reaching the other's hand and shaking it impacts this representation.

We introduce two such representations in Sections 4 and 6. Briefly, we develop a key-segment model for interaction detection and key-pose model for interaction recognition. Following the insight explained above, both models look for "key" temporal and spatial structural components. In dealing with the challenging task of interaction detection in long videos, the key-segment model finds the temporally discriminative sequences of frames, the key-segments, in a video over time. On the other hand, the more complex key-pose representation explicitly specifies how objects are located in time and space in a given track containing a type of interaction. Its enhanced expressive power thus allows it to tell different interactions apart.

4. Interaction Detection: Key-Segment Model

Our approach to interaction detection consists of two major steps (Figure 6). We first coarsely localize objects, in time and space, using off-the-shelf detection and tracking methods. We then use a discriminative max-margin key-segment model to more closely examine if a particular set of objects contains an interaction of interest. The timings of the most informative parts of an interaction track, the *key-segments*, are considered as latent variables in our model. The model therefore encodes the most relevant appearance features and spatial relations in a temporal context. With this two-stage approach we can efficiently process large volumes of video to narrow our search, expending more expensive computations only on a subset that is likely to contain an interaction. This advantage is particularly of interest in surveillance applications where very few interactions happen in a long stream of video. In the following subsections we describe the above steps in more detail.

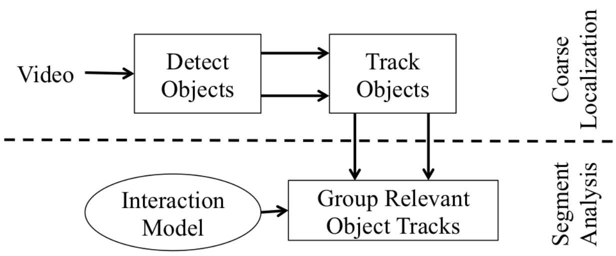


Figure 6: Overview of interaction detection system. There are two major steps: 1) we efficiently but coarsely localize potential interactions in time and space, 2) we more closely examine the content of these space-time volumes to determine if they contain interactions.

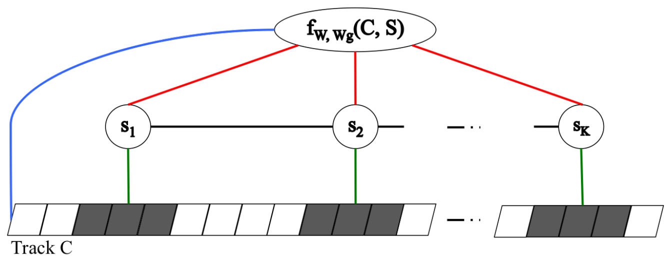


Figure 7: Graphical representation of key-segment model. We score $S = \{s_i < s_{i+1}; i = 1, 2, \ldots, K-1\}$, the arrangement of segments shaded in gray, on a (tentatively) localized track C. The model parameters $W = [w_1, w_2, \ldots, w_K]$ and W_g are adjusted such that the score $f_{W,W_g}(C, S)$ is maximized for the arrangement of key-segments.

4.1. Coarse Localization

We use available object detectors to obtain bounding boxes of objects at the rate of three frames per second. We set the detection threshold low to ensure as few potential candidate interactions as possible are lost; there is no way to find an interaction past this stage if one of the objects involved in it is not retrieved. This comes at the cost of a larger false positive rate which we mitigate by filtering out detections that are unreasonably large and fall in a region where interactions are less likely to occur. We assume access to scene homography and regions of interest that are typically available in surveillance applications. However, automatic discovery of such regions in a given setup is possible as demonstrated in [49].

We use the above object detections to initialize a tracker that follows the object for a fixed duration forward and backward in time. The length of a track, L, is set to be at least twice as long as the average length of an interaction. The tracks centered at the initial detections provide a coarse localization of objects for further analysis where we build potential interaction tracks, the so called *candidates*, by pairing the object tracks.

4.2. Key-Segment Model Formulation

When analyzing a track of a person nearby a vehicle, we can not only use a global description of the entire track, but also focus our attention on specific time instances. For example, important key-segments can include frames portraying the person first bent within the door frame and then moving away from the vehicle. Together with global descriptions of the tracks, these can lead us to infer that the person is getting out of the vehicle. Our key-segment model formalizes this (Figure 7). We treat the temporal location of the important portions of an interaction track, the key-segments, as latent variables and infer their timing by evaluating all the possible ordered arrangements of the segments: we assign each arrangement a score and pick the one with the highest score as representative of the interaction. For a (tentatively) localized track C and an arrangement of its K segments $S = \{s_i < s_{i+1}; i = 1, 2, \ldots, K-1\}$, we define the following scoring function to evaluate the arrangement:

$$f_{W,W_g}(C, S) = \sum_{i=1}^{K} w_i^T \phi(C, s_i) + W_g^T \phi_g(C), \tag{1}$$

where the model parameters $W = [w_1, w_2, \ldots, w_K]$ and W_g are adjusted such that the more representative the segment arrangement within the track, the higher the score it is assigned. Feature functions $\phi(\cdot,\cdot)$ and $\phi_g(\cdot)$ encode the relevant spatio-temporal information across each segment and entire track respectively. In our work, we use appearance features and spatial dynamics: densely sampled HOG3D, center-to-center Euclidean distance of object bounding boxes, and the inner angle of the relative object movement vectors. A detailed description of the features appears below.

Given the above scoring scheme, the arrangement of key-segments within a track is:

$$S^* = \arg\max_{S \in U} f_{W,W_g}(C, S), \tag{2}$$

where U is the set of all possible arrangements of segments in C. In the present work, we only considered segments of fixed length l. Therefore, the i^{th} segment spans a window at frames $[s_i, s_i + l - 1]$ of the track.

4.3. Features

To capture the appearance, motion, and spatial relations of interacting people and vehicles we use HOG3D, distance, and joint direction and distance features. These are computed as follows.

HOG3D. We construct the HOG3D representation of a human-vehicle interaction by concatenating HOG3D features [21] of the human and the vehicle participating in the interaction. We densely sample the regions of video spanned by the human/vehicle bounding boxes in time and space and construct a BoW histogram representation of an entire object track (global representation), or segments of it (Figure 8a). The X (horizontal) and Y (vertical) stride width of dense sampling are equal and scene-dependent. They are set such that at least four horizontal and vertical strides cover a bounding box. Overlapping temporal strides have a width of 10 frames and cover each other by five frames. The histograms of the human and vehicle each have 1000 bins associated with visual words, obtained from K-Means clustering [12] of densely sampled HOG3D features of ground truth object tracks. Both human and vehicle BoW features are normalized so their L_1 norm is 1. A kd-tree structure by [44] speeds up visual word look-up when constructing the histograms.

(a) HOG3D (b) Distance (c) Joint direction and distance

Figure 8: The construction of appearance as well as the relative distance and direction features on the VIRAT dataset [31]. △X, △Y, and △T in (a) are the width of spatial and temporal strides for HOG3D feature extraction.

Distance. For a pair of human and vehicle bounding boxes on a given frame we compute the Euclidean distance between their centers in world coordinates using homography information (Figure 8b). We then pool the distance measurements over the entire interaction track or segments of it to construct a four-bin histogram. The bins are associated with very close, close, far, and very far distance values, quantified by clustering the measurements on ground truth interaction tracks. We use the soft-assignment scheme of [32] to construct the histograms and carry out L1-normalization to get the final distance feature vector.

Joint Direction and Distance. The angle between the person motion vector and the vector connecting the centers of the person and vehicle bounding boxes is indicative of the person's movements with respect to the vehicle (Figure 8c). If a person is about to interact with a vehicle, s/he is likely moving toward the vehicle and not away from it. However, several back and forth movements may occur during the interaction. To capture this, we jointly construct a direction and distance histogram with four bins for each quantity (a total of 4x4 = 16 bins). The direction bins are [-90°, 11.25°, 90°, 168.75°] and encode no motion, moving toward, moving along, and moving away from the vehicle. We use the distance bins quantified above for computations. As before, we perform soft-assignment and L1-normalization to construct the feature vector.

4.4. Learning

We adjust the model parameters in the SVM framework by solving the following constrained optimization problem for N training tracks $\{C_1, C_2, \cdots, C_N\}$ labeled $\{y_1, y_2, \cdots, y_N\}$ respectively where $y_i \in \{1, -1\}$; we do not have annotations for key-segments and infer their value during the training:

$$\min_{W,Wg,\xi_i} \frac{\lambda}{2}(W^TW + W_g^TW_g) + \sum_{i=1}^{N} \xi_i,$$
$$s.t.\ \forall i \;\; y_i \max_{S\in U} f_{W,W_g}(C_i, S) \geq 1 - \xi_i, \;\; \xi_i \geq 0. \tag{3}$$

Combining the two constraints of Equation 3 into one as $\xi_i \geq \max\{0, 1 - y_i \max_{S\in U} f_{W,W_g}(C_i, S)\}$, we can write:

$$\min_{W,Wg,\xi_i} \frac{\lambda}{2}(W^TW + W_g^TW_g) + \sum_{i=1}^{N} \max\{0, 1 - y_i \max_{S\in U} f_{W,W_g}(C_i, S)\}. \tag{4}$$

In general, the objective function in Equation 4 is non-convex. However, it is always convex for the negative samples and convex for the positive ones given a fixed assignment of the latent variables. Therefore, it is possible to iteratively optimize the objective by first inferring the latent variable for a set of parameters, and then optimizing the parameters once the variables are inferred as in [14].

We use the discriminative pre-training trick to simplify the optimization and initialize model parameters to those of an SVM model [9]. We use the NRBM optimization package [10] to solve Equation 4.

4.5. Inference

For track C and interaction model parameters (W, W_g) we would like to find a strictly increasing assignment for latent variables $S^* = \{s_i < s_{i+1}; i = 1, 2, \ldots, K-1\}$ that has the maximum score $f_{W,W_g}(C, S)$ among all the possible assignments S. Given the ordering constraint, we can formulate the inference as a dynamic programming problem.

We define $F(m, t)$ to be the optimal value of $f_{W,W_g}(C, \widehat{S})$ where $\widehat{S} = \{s_i < s_{i+1}; i = 1, 2, \ldots, m-1\}$ and s_m is located on the t^{th} frame ($m \leq K$ and $t \leq L$). We can subsequently define the following recursive relations:

$$F(1, t) = w_1^T \phi(C, t), \tag{5}$$
$$F(m, t) = \max_{m-1 \leq j < t} \{F(m-1, j) + w_m^T \phi(C, t)\}. \tag{6}$$

The best assignment score is given by $\max_{K \leqslant t < L} F(K, t)$ and S^* can be retrieved by backtracking. The time complexity of this process is $O(KL)$, i.e. linear in track length L for a fixed choice of K.

5. Evaluation of Key-Segment Model

We evaluate the key-segment model for interaction detection on the VIRAT Ground Dataset Release 2.0 [31]. VIRAT contains varied interactions in relatively longer videos of wide scenes and is therefore appropriate for detection performance analysis. In the following subsections we describe the data, features, and the experimental setup in detail.

5.1. VIRAT Ground Release 2.0

The dataset contains 8.61 hours of high-definition fixed-camera surveillance videos portraying people naturally performing activities in real environments (e.g. parking lots, construction sites, walkways). There is a total of 11 scenes that significantly vary in terms of lighting condition, camera viewpoint, and human height in pixels. Detailed annotations are available at both event and object levels for 12 different activities, including six human-vehicle interactions: loading/unloading an object to/from a vehicle, opening/closing a vehicle's trunk, getting in/out of a vehicle. Instances of these events occur in a wide spatial range and are temporally scattered. The official release documentation [19] identifies two training-testing schemes: 1) scene-independent: training is carried out on a subset of scenes while testing happens on another mutually exclusive subset. 2) scene-dependent: training and testing samples come from the same set of scenes and thus scene-specific regularities learned during training are helpful at the test time.

We use videos in 10 (out of 11) scenes that are relevant to the task of human-vehicle interaction detection (Table 1) — the only scene we dropped (0100) captures a building facility where no interaction of interest can occur. We follow a scene-independent setting for evaluations [19], and to the best of our knowledge there are not comparable previously published results that use the same setting. Zhu et al. [54] achieve state-of-the-art results on a subset of the dataset in the *scene-dependent* setup, but comparison is difficult without the details of the experimental setup and feature computation. In the experiments reported here, the training scenes are 0101, 0400, 0401, 0502 and comprise 3.43 hours of video. There are a total of 167 correctly annotated interactions in these scenes (Table 1).

5.2. Experiments

Next, we describe the experiments we conducted to verify our choice of features and to evaluate the efficacy of our proposed interaction localization and representation.

5.2.1. Evaluation of Features

We start by using the ground truth tracks from the dataset to evaluate if the proposed features adequately capture the relevant information for detecting interactions. We acknowledge that the features we evaluate in this error-reduced setting may not be ideal in other more realistic settings (e.g. that of 5.2.2), and emphasize that our concern here is how well these features capture the underlying patterns of an interaction.

We construct global BoW representations of HOG3D, HOG3D + Distance, and HOG3D + Distance + joint Direction and Distance features to represent ground truth tracks. We use approximate Histogram Intersection kernel expansion [45] and train a linear SVM model on the expanded features. Any instance of the six interaction classes is considered a positive sample. Pairs of humans and vehicles that do not interact but are spatially close to each other are considered as negative samples. We compiled 145 such pairs for training (See Table 1).

Figure 9 depicts the precision-recall performance of each model, illustrating the importance of features capturing the inter-relations of objects. While all three feature settings perform better than chance, the inclusion of distance features dramatically improves the performance. The overlapping information that joint direction and distance features bring provides additional discriminative power. See Table 2 for a summary of quantitative measurements.

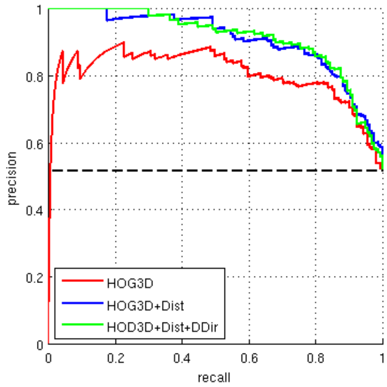


Figure 9: Feature evaluation experiments on VIRAT Ground Release 2.0: Precision-Recall Curves of models trained on appearance (HOG3D), appearance & relative distance (HOG3D+dist), and appearance & relative distance & direction (HOG3D+Dist+DDir) features in red, blue, and green respectively.

5.2.2. Key-Segment Model for Detection

We examine our key-segment interaction model in two different settings. We first show the effectiveness of considering more discriminative segments of an interaction track by comparing the key-segment model against a global BoW + SVM model on ground truth interaction tracks. We then detect interactions based on automatically generated tracks.

Ideal Interaction Tracks. We use the best performing feature representation of 5.2.1 (i.e. HOG3D + Distance + joint Direction and Distance) within the training-test split summarized in Table 1. We train both global BoW + SVM and key-segment models and compare their scores. The key-segment model in the following experiments works with a single latent variable ($K = 1$) and segment length of 20 frames ($l = 20$). As demonstrated in Figure 10, the key-segment model significantly improves detection performance, confirming the insight that examining more discriminative portions of a track is helpful. While the global BoW + SVM model uses the same features, it does not pick the most relevant information; it considers both relevant and irrelevant cues. However, the key-segment model selects the most informative signals to score a track.

Automatically Generated Interaction Tracks. We use human and vehicle detectors Felzenszwalb et al. [14] trained on the PASCAL VOC2009 dataset and tune them to VIRAT by additionally training a kernelized SVM classifier based on HOG3D BoW features densely sampled in detection bounding boxes. We filter out low scoring detections from further analysis. We use [5] to train the SVM classifier.

We use the human detections to initialize the MIL tracker Babenko et al. [3] developed and track them in a time window spanning 200 frames before and after the detection frame (i.e. L = $2 \times 200 = 400$). We do not explicitly track vehicle detections. Since in these human-vehicle interactions the vehicle does not move, we copy the vehicle detection in its place to get its track.

Any pair of coarsely localized human and vehicle tracks that are close enough to each other in time and space is a *candidate* interaction. We use interaction models trained on ground truth

Scene #	0000	0001	0002	0101*	0102	0400*	0401*	0500	0502*	0503	total
Number of Videos	5	2	39	46	76	28	17	14	30	14	329
Length of Videos (h)	0.8	0.46	1.42	0.74	1	1.29	0.54	0.24	0.86	0.4	7.75
(1) Loading objects	2	0	1	0	0	6	5	0	3	0	17
(2) Unloading objects	8	4	3	0	0	19	18	2	4	0	58
(3) Opening trunk	8	2	8	6	0	9	3	0	3	0	39
(4) Closing trunk	9	2	8	6	0	7	2	0	3	0	37
(5) Getting in	16	3	21	9	1	9	3	1	25	6	94
(6) Getting out	14	4	33	0	0	6	6	1	15	2	81
All Interactions	57	15	74	21	1	56	37	4	53	8	326
Background	0	1	22	75	11	31	36	32	3	84	295

Table 1: Statistics of VIRAT Ground Dataset Release 2.0 data. Training scenes are marked by *. Interaction samples have been obtained by cross referencing valid entries of mapping files in objects files and visually inspecting the tracks to verify their content. Background samples are pairs of spatially close people and vehicles not involved in an interaction. We have randomly picked a subset of size 295 out of these pairs for our experiments.

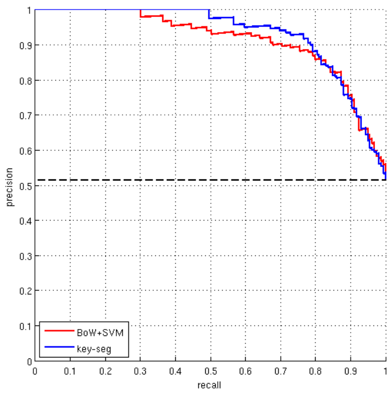


Figure 10: Interaction detection experiment on ideal tracks of VIRAT Ground Release 2.0: Precision-Recall Curves of BoW+SVM (red) and key-segment (blue) models both trained on appearance & relative distance & direction (HOG3D+Dist+DDir) features extracted from ground truth person and vehicle tracks.

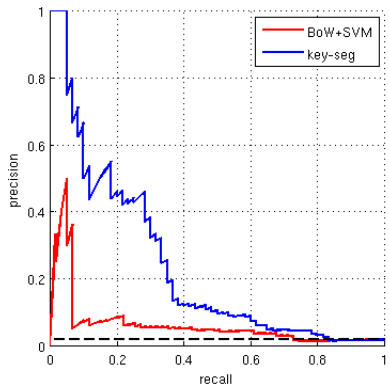


Figure 11: Interaction detection experiment on automatically generated tracks in VIRAT Ground Release 2.0: Precision-Recall Curves of BoW+SVM (red) and key-segment (blue) models applied to automatically generated tracks of people and vehicles based on their appearance & relative distance & direction (HOG3D+Dist+DDir) features.

data (i.e. the two models from 5.2.2) and score how well these candidates represent an interaction. Following [19]'s evaluation methodology, we consider candidates whose temporal and spatial intersection over union overlap with a ground truth sample is larger than 10% as a correct detection.

In Figure 11, we report the performance of the scheme described above for videos in scenes 0000 and 0001, where the height of the humans in the scene is large enough for the detection models to work reasonably well. Figure 12 shows sample key-segment model outputs.

Analysis. The key-segment model significantly outperforms the global BoW model by incorporating structural information. A comparison of key-segment and global BoW performance in the two evaluation settings, one involving ground truth tracks and the other involving automatically generated tracks, reveals the importance of selecting the most informative cues. For ground-truth tracks, the key-segment model achieves ~2% additional improvement over global BoW; for automated tracks it increases average precision by ~17%.

Inspecting the top scored samples, we see that the key-segment model usually favors the moments when the person makes a move with respect to the vehicle; a reasonable cue of an imminent interaction. Additionally examining the top ranked false positives reveals some of the difficulties in working within the limited settings that VIRAT dataset offers. For example, Figure 12b displays a person moving toward the vehicle and bending over the window. Such an event can be considered as an interaction, although it is not specified as one and so there is no label for it. Also, there are lost interactions as in Figure 12d, where the annotations are not available for an occurrence of the already defined interaction.

The performance is heavily dependent on the quality of the interaction tracks built on top of the object tracks. Developing robust detection and tracking for the diverse VIRAT videos is a challenge, and we are not aware of published results with effective methods (e.g. based on moving region detection or person/vehicle detectors) that are effective. However, our results on ground-truth tracks show that the features and model we propose are effective. We provide evidence that with improved detection and tracking modules, the overall system could obtain results closer to average precision of 93.03% obtained by ground-truth tracking. Further, more detailed models with $K > 1$ can be applied in finer-grained settings with more reli-

Model	AUC	AP
Trained and Tested on Ground Truth Tracks		
HOG3D BoW + SVM	80.16%	80.57%
HOG3D+Dist BoW + SVM	90.88%	90.92%
HOG3D+Dist+DDir BoW + SVM	91.37%	91.40%
HOG3D+Dist+DDir + key-seg	**93.01%**	**93.03%**
Automatically Generated Tracks		
HOG3D+Dist+DDir BoW + SVM	5.97%	6.63%
HOG3D+Dist+DDir key-seg	23.36%	23.78%

Table 2: Results of interaction detection on VIRAT Ground Release 2.0. AUC: area under Precision-Recall curve, AP: average precision. HOG3D: appearance feature, Dist: distance feature, DDir: joint direction and distance feature.

able detection and tracking. In the next section we explore more detailed models in the context of human-human interactions.

6. Interaction Recognition: Key-Pose Model

In our approach to recognizing human interactions, we are looking for descriptive and infrequent moments in (tentative) tracks of people. To this end, we use a discriminative max-margin key-pose model to identify the most informative frames of person tracks, the so-called key-poses. We characterize the key poses by their role, timing, location, and appearance. This information is encoded as latent variables in our model. Moreover, we account for the spatial arrangements of the key-poses over time. Our model thus considers the relevant frames of a track only and ignores the misleading and highly variable ones. Its expressive power is also improved by explicitly encoding the spatial structure of people participating in the interaction. In the following section we formally describe the key-pose model for human-human interaction recognition.

6.1. Model Formulation

Observing two people, one approaching the other with his hand extended in an offensive pose and the other defensively stepping back shortly after, leads us to infer that an agressive act, for instance one person punching another, is taking place. We formalize this with our key-pose model. Given a pair of person tracks we represent their interaction by two series of chronologically ordered inter-related key-poses (one for the subject and the other for the object of the interaction) that are discriminative in appearance and spatial structure. We consider as latent variables the role (subject vs. object), timing, location, and specifics of appearance of these key-poses, and infer them by evaluating all the valid combinations of these variables. The evaluation is based on a score we assign to a set of values for latent variables and quantifies how well it encodes the underlying interaction; the highest scored combination represents the interaction. Below, we describe these variables and our scoring function in more detail.

6.1.1. Latent Variables

A key-pose is identified by its role, timing, location, and appearance to capture the following information:

- Role (r): whether the sequence containing the key-pose is the subject or the object of the interaction.
- Timing (t): when in a tentative track of the person the key-pose occurs. Chronological order is enforced among key-poses of a sequence.
- Location (s): where in the space around the tentative track of the person the key-pose is located. That is, s varies in a vicinity of a tracker's output that roughly estimates where people are in a video and allows us to handle modest tracking errors.
- Appearance (e): how the key-pose looks. For example, does it look like a punch in the face or a punch in the armpit? e is selected from a discrete set of exemplars, $\mathcal{E}$, containing possible appearance variants of key-poses. We separately construct $\mathcal{E}$; see 7.2 for details.

Formally, we aggregate this information in a single variable $h = [r, t, s, e]$. We can thus encode a sequence of K key-poses by $H = [h_1, h_2, \ldots, h_K]$ where h_i is the i^{th} key-pose. r_i's take a single value in all the key-poses of one sequence, i.e. $\forall i, r_i = r_1$ and r_1 is either subject or object. In the present work, we assume there is a fixed number of key-poses in any sequence.

6.1.2. Scoring Function

For tentative tracks C^1 and C^2 of two people and an arrangement of their key-poses H^1 and H^2 we define the following scoring function:

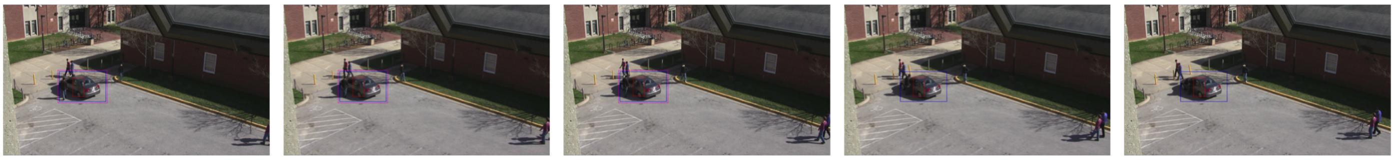
$$\begin{aligned} f_{W_s,W_o,W_d}(C^1, C^2, y, H^1, H^2) = P_{W(r_1^1)}(C^1, y, H^1) &+ \\ P_{W(r_1^2)}(C^2, y, H^2) &+ \\ Q_{W_d}(C^1, C^2, y, H^1, H^2), & \quad (7) \end{aligned}$$

to evaluate how representative the key-pose series are for an activity labeled y. Function P scores the compatibility between the activity label and the appearance of the key-poses as well as their temporal order. $W(\cdot)$ equals W_s if the sequence takes the subject role, and equals W_o if it takes the object role. We thus account for the asymmetry in many interactions by explicitly modeling each role. Function Q examines the relative spatial distance between the key-poses of one track from the other track, and whether the distance pattern is compatible with the underlying interaction. Formally, we define P and Q as follows:

$$\begin{aligned} P_W(C, y, H) = \sum_{i=1}^{K} \alpha^T \Phi_0(C, t_i, s_i, e_i) \quad &+ \\ \sum_{i=1}^{K} \beta_i^T \Phi_1(y, e_i) \quad &+ \\ \sum_{i=1}^{K} \gamma^T \Phi_2(C, y, t_i, s_i). & \quad (8) \end{aligned}$$

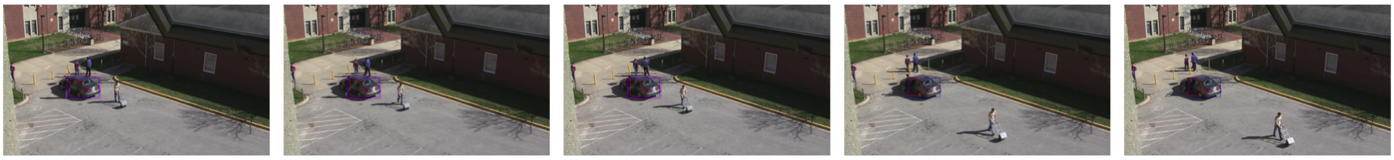
The three terms in the above formulation are graphically illustrated in Figure 13 by links associated with potential functions Φ_0, Φ_1, and Φ_2 respectively. They represent:

(a) rank = 1, label = 1, the top scored true positive. The person moves toward the vehicle and opens the trunk.



(b) rank = 4, label = -1, the top scored false positive. The person moves toward the vehicle and bends over the window

(c) rank = 5, label = 1. The person gets into the vehicle and disappears.



(d) rank = 8, label = -1. The person moves toward the vehicle and gets into it. The annotations were missing for this sample.

Figure 12: Top scored samples of VIRAT Ground Release 2.0. We show a subset of frames that best exemplify the output. Person and vehicle bounding boxes are in red and blue respectively. They are enclosed by a magenta box on frames of the inferred key-segment. The figure is best viewed magnified and in color.

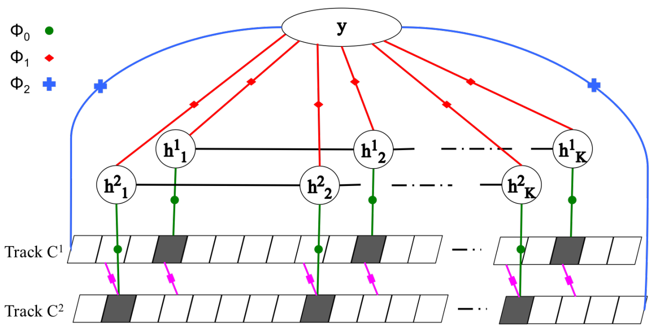


Figure 13: Graphical representation of key-pose model. We score the key-pose series $H^1 = [h^1_1, h^1_2, \ldots, h^1_K]$ and $H^2 = [h^2_1, h^2_2, \ldots, h^2_K]$ for tentative tracks of people C^1 and C^2. A h^j_i is a key-pose identified by its role, timing, location, and appearance. A temporal order constraint is enforced among key-poses in each sequence. The lines with circle (dark green), diamond (red), cross (blue), and square (magenta) shapes on them represent the potential functions: exemplar match, activity-key-pose match, image appearance match, and distance respectively. The model parameters W_s, W_o, W_d are adjusted such that the score $f_{W_s,W_o,W_d}(C^1, C^2, y, H^1, H^2)$ is maximized for the combination of key-poses that best represent the interaction. For example, a person in an offensive pose with one hand extended and another bent in a defensive pose are representative of a punching interaction.

Exemplar Matching Link. $\alpha^T \Phi_0(C, t_i, s_i, e_i)$ measures the compatibility between exemplar e_i and the image evidence at time t_i and location s_i. It is defined as:

$$\alpha^T \Phi_0(C, t_i, s_i, e_i) = \sum_{j=1}^{|\mathcal{E}|} {\alpha_j}^T D(\phi(C, t_i, s_i), \phi(e_i)) \mathbb{1}_{\{e_i = \, j^{th} \ element \ of \ \mathcal{E}\}}. \quad (9)$$

$\phi(C, t_i, s_i)$ encodes appearance features at time t_i and location s_i of track C. $\phi(e_i)$ captures similar information in exemplar e_i. In our work we densely sample HOG [7] and HOF [8] features in an 8×8 grid of non-overlapping cells covering a person's bounding box and concatenate them to represent the appearance and motion of the person. We measure the similarity between two appearance representations by calculating $D(\cdot, \cdot)$, the normalized Euclidean distance between the features of corresponding cells in the grid (Figure 14). $D(\cdot, \cdot)$ is therefore a vector with its i^{th} element being the normalized Euclidean distance of HOG and HOF features at the corresponding locations. $\mathbb{1}$ is an indicator function selecting the parameters associated with exemplar e_i.

Activity-Keypose Link. $\beta_i^T \Phi_1(y, e_i)$ measures the compatibility between exemplar e_i and activity y; the higher it is, the stronger the exemplar e_i is associated with activity y. It is formulated as:

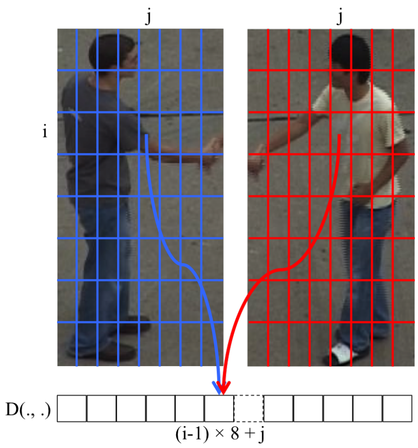


Figure 14: 8×8 grid of HOG and HOF dense sampling and the visualization of $D(\cdot,\cdot)$ computation between two representations.

$$\beta_i{}^T\Phi_1(y, e_i) = \sum_{a\in\mathcal{Y}}\sum_{j=1}^{|\mathcal{E}|}\beta_{iaj}\mathbb{1}_{\{y=a\}}\mathbb{1}_{\{e_i=\ j^{th}\ element\ of\ \mathcal{E}\}}, \tag{10}$$

where $\mathcal{Y}$ is the finite set of activities we want to recognize. The activity key-pose term β_i is indexed to capture variations of compatibility between an exemplar and an activity over time; a particular e_i may be better associated with the beginning of y than the ending of it. It also allows our model to account for the varied orders a key-pose can take in different activities.

Direct Root Model. $\gamma^T\Phi_2(C, y, t_i, s_i)$ directly measures the compatibility between the activity and the image evidence at time t_i and location s_i:

$$\gamma^T\Phi_2(C, y, t_i, s_i) = \sum_{a\in\mathcal{Y}}\gamma_a{}^T\phi(C, t_i, s_i)\mathbb{1}_{\{y=a\}}. \tag{11}$$

In our overall model formulation in Equation 7, $W_s = [\alpha, \beta_s, \gamma]$ and $W_o = [\alpha, \beta_o, \gamma]$ explicitly model for subject and object roles. Note that α and γ are assumed to be identical in both roles.

Function Q evaluates the spatial structure between people participating in the interaction by assessing the compatibility between activity y and the distance of the i^{th} key-pose of one track from the other. It is calculated as:

$$Q_{W_d}(C^1, C^2, y, H^1, H^2) = \sum_{i=1}^{K}\mu_i{}^T\theta(C^2, y, t_i^1, s_i^1) + \sum_{i=1}^{K}\mu_i{}^T\theta(C^1, y, t_i^2, s_i^2), \tag{12}$$

where $W_d = [\mu_1, \mu_2, \ldots, \mu_K]$ and $\mu_i{}^T\theta(C^b, y, t_i^j, s_i^j)$ is

$$\sum_{a\in\mathcal{Y}}\mu_{ia}{}^T bin(\|l(C^b, t_i^j) - s_i^j\|_2)\mathbb{1}_{\{y=a\}}. \tag{13}$$

$b \neq j$ and $l(C^b, t_i^j)$ is the location of the person enclosed in track C^b at time t_i^j. The distance is computed as the center-to-center Euclidean distance, d, of bounding boxes (in pixels) and is discretized as $bin(d) = \lceil\frac{d}{30}\rceil$.

We adjust the model parameters $[W_s, W_o, W_d]$ such that the more representative a combination of values for latent variables is, the higher the score it is assigned. With this scoring scheme, the key-pose representation of an interaction is:

$$(H^{1^*}, H^{2^*}) = \arg\max_{(H^1,H^2)\in\mathcal{H}_1\times\mathcal{H}_2} f_{W_s,W_o,W_d}(C^1, C^2, y, H^1, H^2), \tag{14}$$

where $\mathcal{H}_1\times\mathcal{H}_2$ is the space of all possible combinations of key-poses. In the next sections we describe learning and inference procedures for adjusting model parameters and deploying them to obtain (H^{1^*}, H^{2^*}).

6.2. Learning

We adjust model parameters in a latent structural SVM framework for N pairs of person tracks $\{(C_1^1, C_1^2), (C_2^1, C_2^2), \ldots, (C_N^1, C_N^2)\}$ labeled $\{y_1, y_2, \ldots, y_N\}$ with y_i's in $\mathcal{Y}$, a discrete set of interaction categories. We formulate the learning criteria as:

$$\begin{aligned}\min_{W_s,W_o,W_d,\xi_i} \frac{\lambda}{2}(W_s^T W_s + W_o^T W_o + W_d^T W_d) + \sum_{i=1}^{N}\xi_i,\\ s.t.\ \ \forall i\ \ f_{W_s,W_o,W_d}(C_i^1, C_i^2, y_i, H^1, H^2) -\\ f_{W_s,W_o,W_d}(C_i^1, C_i^2, y, H^1, H^2) > \Delta(y_i, y) - \xi_i,\end{aligned} \tag{15}$$

where $\Delta(y_i, y)$ is 0-1 loss. The constraint in Equation 15 ensures that the correct label for a training sample is scored higher than any incorrectly hypothesized label. The optimization problem above is non-convex and is solved using the non-convex extension of the cutting-plane algorithm provided in NRBM optimization package [10]. We also heuristically initialize model parameters: we divide each track into K non-overlapping temporal segments and match the frames in each segment to its nearest exemplar. β_{iyj} for the i^{th} segment is set to the frequency of the j^{th} exemplar in that segment for class label y.

6.3. Inference

For tracks C^1 and C^2 of two people and model parameters (W_s, W_o, W_d), we are looking for a combination of latent variables (H^{1^*}, H^{2^*}) among all possible (H^1, H^2) that maximizes $f_{W_s,W_o,W_d}(C^1, C^2, y, H^1, H^2)$ for each activity label y. Label with the maximum f_{W_s,W_o,W_d} indicates the category of the interaction contained in C^1 and C^2. Note that maximization can be decomposed into two terms each corresponding to one sequence as the interaction distance function Q in Equation 12 is decomposable into two independent terms each measuring distance of key-poses in one sequence from the other track:

$$\max_{(H^1,H^2)\in\mathcal{H}_1\times\mathcal{H}_2} f_{W_s,W_o,W_d}(C^1,C^2,y,H^1,H^2) = \quad (16)$$
$$\max_{(H^1)\in\mathcal{H}_1}\{P_{W(r_1^1)}(C^1,y,H^1) + \sum_{i=1}^{K}\mu_i{}^T\theta(C^2,y,t_i^1,s_i^1)\} +$$
$$\max_{(H^2)\in\mathcal{H}_2}\{P_{W(r_1^2)}(C^2,y,H^2) + \sum_{i=1}^{K}\mu_i{}^T\theta(C^1,y,t_i^2,s_i^2)\}.$$

We can rewrite the maximization for a track C as:

$$\max_H \sum_{i=1}^{K} A_i^{t_i} \quad s.t.\ \ t_i < t_{i+1}\forall i = 1,2,\ldots,K-1; \quad (17)$$

where for each h_i in an H, $r_i \in \{subject, object\}$, $1 \le t_i \le L$ (L is the track length), s_i varies in a neighborhood around the t_i^{th} frame of the track, and $e_i \in \mathcal{E}$. $A_i^{t_i}$ is defined as:

$$A_i^{t_i} = \max_{r_i,s_i,e_i}\{\alpha^T\Phi_0(C,t_i,s_i,e_i) + \beta_i^T\Phi_1(y,e_i) + \gamma^T\Phi_2(C,y,t_i,s_i) + \mu_i^T\theta(C^b,y,t_i,s_i)\}; \quad (18)$$

C^b is the other track involved in the interaction. β is β_s if r_i's take the subject role and is β_o otherwise.

The chronological ordering constraint on key-pose timings allows us to formulate inference as a dynamic programming problem that can be solved efficiently. We define $F(m,t)$ as the maximum value of max $\sum_{i=1}^{m} A_i^{t_i}$ for $t_i < t_{i+1} \in \{1,2,\ldots,t\}$ $\forall i = 1,2,\ldots,m-1$. The following relations specify how $F(m,t)$ can be computed recursively:

$$F(1,t) = \max\{A_1^1, A_1^2, \ldots, A_1^t\}, \quad (19)$$
$$F(m,m) = F(m-1,m-1) + A_m^m, \quad (20)$$
$$F(m,t) = \max\{F(m-1,t-1) + A_m^t, F(m,t-1)\},\ m < t \quad (21)$$

$F(K,L)$ gives the solution to each term in Equation 17. The optimal key-poses for each track can then be retrieved by back-tracking. The order of growth for this process is $O(KL)$, again linear in track length L for fixed K.

7. Evaluation of Key-Pose Model

We evaluate the key-pose model for interaction classification on the UT-Interaction [39] benchmark. We first describe the data and our training-test setup as well as the preprocessing steps for obtaining tentative tracks of people and the set of their discriminative poses. We subsequently specify the key-pose model parameters and present the quantitative and qualitative results of interaction recognition based on key-pose representations.

7.1. UT-Interaction Dataset

The dataset portrays two people interacting with each other in two scenes: a parking lot (Set 1) and a lawn (Set 2). There are 10 videos (720×480, 30fps) in each scene with average duration of one minute. Each video provides an average of 8 sample interactions that are continuously performed by actors and contains at least a sample of each interaction category: shake-hands, point, hug, push, kick, and punch. While there is some camera jitter and pedestrians walking by in some of the videos, the scenes are otherwise static and clear. People's appearance varies across videos but camera viewpoint and the human height in pixels is stable (~200). Ground truth annotations provide time intervals and bounding boxes for interactions that give the 120 cropped video clips for the classification task. We augment these annotations for the pointing interaction to also account for the person being pointed to. In our training-test setup, we follow the 10-fold leave-one-out cross validation scheme of [39] and report the average performance.

7.2. Preprocessing

We should provide our model with initial tracks of people and a set of exemplar poses, $\mathcal{E}$, they take while interacting with each other. Below, we detail the steps to obtain this information:

Person Tracks. We use Dalal and Triggs [7]'s human detector on the first frame of every video clip and pick the two out of the three top scoring detections that are closest horizontally. We initialize Ross et al. [36]'s tracker to get the person tracks that will be later input to our model. We construct tracks at two different scales to accommodate the camera zoom in videos of Set 1.

Exemplar Set. We train a multi-class linear SVM classifier based on HOG and HOF features to score how discriminative frames of annotated tracks are of the interactions they each belong to. We then cluster the highest scored bounding boxes to get the discriminative exemplars for each interaction category separately. Note that the initial classification step ensures that our K-Means clustering does not simply favor the most common as opposed to the most discriminative poses when constructing clusters. This heuristic procedure is efficient and effective, while it achieves what more sophisticated clustering algorithms (e.g. [26]) do in our experiments. We use [13] to train the pose classifier and [12] to perform K-Means clustering with 20 clusters and $D(\cdot,\cdot)$ (see 6.1.2) as the distance measure. Since the cluster centroids are averaged virtual poses and do not exist in the data, we use the samples from training set that are nearest to the cluster centers as the final set of exemplars.

7.3. Experiments

We compare our key-pose model against a global BoW + SVM model that does not account for any structure. We also construct two other baselines to examine the importance of structural information, namely the relative spatial movements and the differentiation of subject-object role in the interaction: 1) a model that includes neither the distance term, Q, nor the

Model	Set 1	Set 2	Avg
Key-pose model and its structural elements			
Global BoW + SVM	68.6%	70.0%	69.3%
Temporal ordering only	83.3%	86.7%	85.0%
Temporal + role	86.7%	88.3%	87.5%
Spatial + temporal + role	93.3%	90.0%	91.7%
Other models in the literature			
Ryoo [37]	85%	-	-
Yu et al. [52]	-	-	83%
Yao et al. [51]	88%	80%	84%
Zhang et al. [53]	95%	90%	92%
Kong et al. [22]	88.3%	-	-
Raptis and Sigal [35]	93.3%	-	-

Table 3: Classification performance of our model on the UT-Interaction benchmark and comparisons with other models. Set 1 and Set 2 refer to parking lot and lawn scenes respectively. We progressively consider more structural information, moving from the first baseline (global BoW + SVM) to our full model that incorporates spatial and temporal structure as well as the subject-object role of actors. The best reported performance of other papers are included in the table.

latent variable "role" (i.e. $\beta_s = \beta_o$), and 2) a model where only the distance term is ignored.

The key-pose model in the following experiments identifies a fixed number of key-poses ($K = 5$) in tracks obtained from video clips. The (X, Y) location, s, of a key-pose varies in the vicinity of the input track (X_{tr}, Y_{tr}) in a small grid, i.e. $X \in \{X_{tr} - \delta_X, X_{tr}, X_{tr} + \delta_X\}$ and $Y \in \{Y_{tr} - \delta_Y, Y_{tr}, Y_{tr} + \delta_Y\}$. In our experiments we set δ_X and δ_Y to 20 and 15 pixels respectively.

The global BoW + SVM model is a "bag of poses" approach – we use the exemplar set (see 7.2) as pose prototypes. The frequency of the occurrence of these prototypes over a video sequence is computed and stored in a histogram. This bag of words-style approach is akin to that used in Wang and Mori [46], capturing the frequencies of human pose prototypes across a video sequence. The subsequent models build additional spatio-temporal structure that enhance classification accuracy.

Our model achieves 91.7% average accuracy for the classification task, a 22.4%-point improvement over the global model (Table 3). Accounting for the temporal ordering of discriminative poses alone achieves 85.5% accuracy and is improved by ≈3% with the addition of the role variable. By additionally modeling the relative distance in our full model, we obtain the highest accuracy. Confusion matrices in Figure 15 provide more details regarding the performance of our model for different interactions. As shown in the figure, there is some confusion between "push" and "punch." It is not unexpected though; the two activities are similar in both appearance and relative movements of the people involved.

Varying the number of key-poses K (Table 4) suggests that very few key-poses (i.e. $K = 1$ or 2) fail to capture the temporal dynamics of interactions. Moreover, performance is relatively unchanged for very large K's (e.g. $K = 10$).

Overall, our method is competitive with the state of the art methods. Further, it does not require additional labeling effort – it only needs a per-sequence interaction label. The key-poses

#key-poses (K)	Set 1	Set 2	Avg
$K = 1$	89.9%	86.7%	88.3%
$K = 2$	83.5%	86.7%	85.1%
$K = 5$	**93.3%**	**90.0%**	**91.7%**
$K = 10$	88.0%	**90.0%**	89.0%

Table 4: Classification performance of our model on the UT-Interaction benchmark for varied number of key-poses (K). Very few key-poses fail to capture the temporal dynamics of interactions. Larger values, such as $K = 5$, are effective for the UT-interaction dataset. Very large numbers, e.g. $K = 10$, do not lead to any improvements.

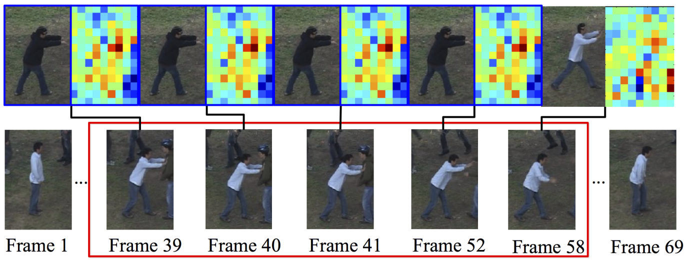


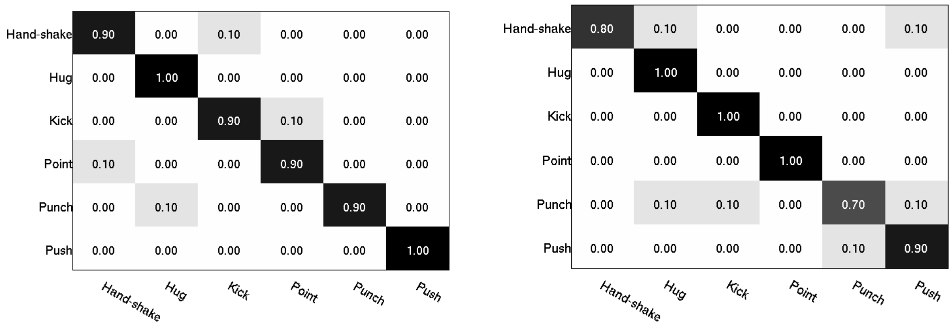
Figure 16: The key-pose series our model produces for a 69-frame video clip. At the top, we have visualized the exemplars matched to each frame at the bottom. The key-poses are enclosed in a red box. The number under each frame is the frame number. The appearance of exemplars matches the image evidence. The heat-map next to each exemplar depicts the learned model weights for matching to each exemplar. As the heat-maps show, higher weights (darker red cells) are learned for the discriminative appearance that covers the person and are largely concentrated on the extended hands for pushing. The key-poses are more densely localized at discriminative moments such as when extending hands and making contact with the other person.

and their spatio-temporal locations are discovered by the model. The approach seems robust to intra-class variations and inter-person occlusions, likely due to the proposed key-pose representation.

Figures 16-18 illustrate how our model works by visualizing exemplar matching, activity-key pose weights, and the distance profile of key-poses over time. We observe that the key-pose model successfully localizes discriminative frames of a track (enclosed by a red box in Figure 16) and associates them with similar exemplars. Another interesting observation is that the key-poses are not uniformly spaced in time. In fact, they are denser at the peak moments, for example the duration when the attacker's hands are extended and the contact happens in a pushing interaction.

Moreover, our model handles pose variations using the exemplar representation. The three top scored exemplars depicted for each key-pose in Figure 17 vary considerably in appearance.

We also examine the contribution of the spatial distance constraint when a key-pose is localized. As Figure 18 reveals, the spatial relation profile differs across interactions. As expected, the model learns shorter distances for hugging and longer ones for pointing. Additionally, the profile for pushing correctly captures the variations in distance throughout the interaction; the model associates shorter distances with the starting key-poses and longer distances with the ones at the end.



(a) Set 1 (b) Set 2

Figure 15: Confusion matrices of classification performance on the UT-Interaction dataset. Rows are associated with ground truth, while columns represent predictions.

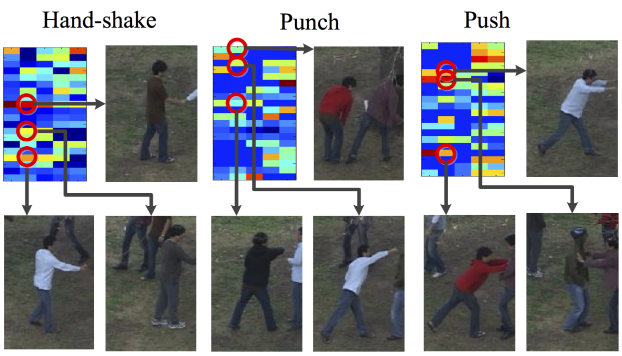


Figure 17: The heat-map and top scored exemplars for a key-pose in hand-shake, punch, and push interactions. Each heat-map represents 20 exemplars associated with the activity vertically, and the 5 key-poses in the key-pose series horizontally. Therefore, each cell on the heat-map scores how well a particular exemplar matches the activity at the time of the key-pose; the higher the score, the redder the cell. The top scored exemplars are varied in appearance.

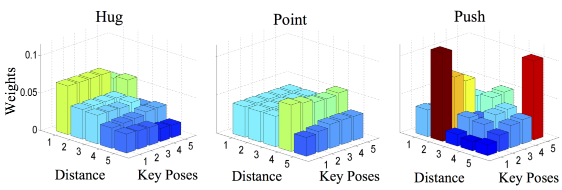


Figure 18: Visualization of discretized spatial distances of key-poses for hug, point, and push interactions with discrete distance, key-poses, and the associated weights on three axes. The higher and darker the bar, the larger its weight. Not surprisingly, smaller distances are preferred for hug while the opposite is true for point. The preferred distance during pushing changes from near (first key-pose) to far (last key-pose).

8. Conclusion

In this paper we developed structured models for human interaction detection and recognition in video sequences. These models select a set of key-components, discriminative moments in a video sequence that are important evidence for the presence of a particular interaction. We demonstrated the effectiveness of this model for detecting human-vehicle interactions in long surveillance videos. On the VIRAT dataset we showed that appearance features combined with relative distance and motion features can be effective for detection, and accuracy is enhanced by the selection of an important key-component. Further experiments on the UT-Interaction dataset of human-human interactions verified that incorporating temporal and spatial structure in the form of a series of key-components results in state-of-the-art classification performance, and improvements over unstructured baselines.

We demonstrated highly accurate interaction detection when good quality human detection and tracking are available, from ground truth data on VIRAT and automatic tracks on UT-Interaction. Automatic tracks on VIRAT still resulted in effective pruning of potential interactions. Directions for future work include further experimentation with other trackers and refinements to the model to choose the appropriate number of key-poses for each sequence automatically.

Acknowledgements

This work was supported by NSERC, CIIRDF, and MDA Corporation.

REFERENCES

[1] Aggarwal, J., Ryoo, M., 2011. Human activity analysis: A review. ACM Computing Surveys 43 (2), 16:1–16:43.

[2] Amer, M. R., Xie, D., Zhao, M., Todorovic, S., Zhu, S.-C., 2012. Cost-sensitive top-down/bottom-up inference for multiscale activity recognition. In: European Conference on Computer Vision.

[3] Babenko, B., Yang, M.-H., Belongie, S., 2011. Robust object tracking with online multiple instance learning. IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (8), 1619–1632.

[4] Brendel, W., Todorovic, S., 2010. Activities as time series of human postures. In: European Conference on Computer Vision.

[5] Chang, C.-C., Lin, C.-J., 2011. LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2 (3), 27:1–27:27.

[6] Choi, W., Savarese, S., 2012. A unified framework for multi-target tracking and collective activity recognition. In: European Conference on Computer Vision.

[7] Dalal, N., Triggs, B., 2005. Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition.

[8] Dalal, N., Triggs, B., Schmid, C., 2006. Human detection using oriented histograms of flow and appearance. In: European Conference on Computer Vision.

[9] Desai, C., Ramanan, D., Fowlkes, C., 2009. Discriminative models for multi-class object layout. In: International Conference on Computer Vision.

[10] Do, T. M. T., Artières, T., 2009. Large margin training for hidden markov models with partially observed states. In: International Conference on Machine Learning.

[11] Efros, A., Berg, A., Mori, G., Malik, J., 2003. Recognizing action at a distance. In: International Conference on Computer Vision.

[12] Everingham, M., 2003. VGG K-means. http://www.robots.ox.ac.uk/ vgg/software.

[13] Fan, R., Chang, K., Hsieh, C., Wang, X., Lin, C., 2008. LIBLINEAR: A library for large linear classification. Journal of Machine Learning Research 9, 1871–1874.

[14] Felzenszwalb, P., Girshick, R., McAllester, D., Ramanan, D., 2010. Object detection with discriminatively trained part based models. IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (9), 1627–1645.

[15] Gupta, A., Kembhavi, A., Davis, L., 2009. Observing human-object interactions: Using spatial and functional compatibility for recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (10), 1775–1789.

[16] Intille, S., Bobick, A., 2001. Recognizing planned, multiperson action. Computer Vision and Image Understanding 81 (3), 414–445.

[17] Ke, Y., Sukthankar, R., Hebert, M., 2007. Event detection in crowded videos. In: International Conference on Computer Vision.

[18] Khamis, S., Morariu, V. I., Davis, L. S., 2012. Combining per-frame and per-track cues for multi-person action recognition. In: European Conference on Computer Vision.

[19] Kitware, 2011. Data release 2.0 description. http://www.viratdata.org.

[20] Kjellstrm, H., Romero, J., Kragi, D., 2011. Visual object-action recognition: Inferring object affordances from human demonstration. Computer Vision and Image Understanding 115 (1), 81–90.

[21] Kläser, A., Marszałek, M., Schmid, C., 2008. A spatio-temporal descriptor based on 3D-gradients. In: British Machine Vision Conference.

[22] Kong, Y., Jia, Y., Fu, Y., 2012. Learning human interaction by interactive phrases. In: European Conference on Computer Vision.

[23] Koppula, H. S., Gupta, R., Saxena, A., 2013. Learning human activities and object affordances from rgb-d videos. International Journal of Robotics Research 32 (8), 951–970.

[24] Kovashka, A., Grauman, K., 2010. Learning a hierarchy of discriminative space-time neighborhood features for human action recognition. In: Computer Vision and Pattern Recognition.

[25] Lan, T., Wang, Y., Yang, W., Robinovitch, S. N., Mori, G., 2012. Discriminative latent models for recognizing contextual group activities. Pattern Analysis and Machine Intelligence, IEEE Transactions on 34 (8), 1549–1562.

[26] Lazebnik, S., Raginsky, M., 2009. Supervised learning of quantizer codebooks by information loss minimization. IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (7), 1294–1309.

[27] Lv, F. J., Nevatia, R., 2007. Single view human action recognition using key pose matching and viterbi path searching. In: Computer Vision and Pattern Recognition.

[28] Marszałek, M., Laptev, I., Schmid, C., 2009. Actions in context. In: Computer Vision and Pattern Recognition.

[29] Niebles, J. C., Chen, C.-W., Fei-Fei, L., 2010. Modeling temporal structure of decomposable motion segments for activity classification. In: European Conference on Computer Vision.

[30] Niebles, J. C., Wang, H., Fei-Fei, L., 2006. Unsupervised learning of human action categories using spatial-temporal words. In: British Machine Vision Conference.

[31] Oh, S., Hoogs, A., Perera, A., Cuntoor, N., Chen, C.-C., Lee, J. T., Mukherjee, S., Aggarwal, J., Lee, H., Davis, L., et al., 2011. A large-scale benchmark dataset for event recognition in surveillance video. In: Computer Vision and Pattern Recognition.

[32] Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A., 2008. Lost in quantization: Improving particular object retrieval in large scale image databases. In: Computer Vision and Pattern Recognition.

[33] Pieropan, A., Ek, C. H., Kjellstrom, H., 2013. Functional object descriptors for human activity modeling. In: International Conference on Robotics and Automation.

[34] Poppe, R., 2010. A survey on vision-based human action recognition. Image and Vision Computing 28 (6), 976–990.

[35] Raptis, M., Sigal, L., 2013. Poselet key-framing: A model for human activity recognition. In: Computer Vision and Pattern Recognition.

[36] Ross, D. A., Lim, J., Lin, R.-S., Yang, M.-H., 2008. Incremental learning for robust visual tracking. International Journal of Computer Vision 77 (1-3), 125–141.

[37] Ryoo, M., 2011. Human activity prediction: Early recognition of ongoing activities from streaming videos. In: International Conference on Computer Vision.

[38] Ryoo, M., Aggarwal, J., 2009. Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities. In: International Conference on Computer Vision.

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

[40] Schüldt, C., Laptev, I., Caputo, B., 2004. Recognizing human actions: A local svm approach. In: International Conference on Pattern Recognition.

[41] Tang, K., Fei-Fei, L., Koller, D., 2012. Learning latent temporal structure for complex event detection. In: Computer Vision and Pattern Recognition.

[42] Tian, Y., Sukthankar, R., Shah, M., 2013. Spatiotemporal deformable part models for action detection. In: Computer Vision and Pattern Recognition.

[43] Vahdat, A., Gao, B., Ranjbar, M., Mori, G., 2011. A discriminative key pose sequence model for recognizing human interactions. In: IEEE International Workshop on Visual Surveillance.

[44] Vedaldi, A., Fulkerson, B., 2008. VLFeat: An open and portable library of computer vision algorithms. `http://www.vlfeat.org/`.

[45] Vedaldi, A., Zisserman, A., 2011. Efficient additive kernels via explicit feature maps. IEEE Transactions on Pattern Analysis and Machine Intellingence 34 (3), 480–492.

[46] Wang, Y., Mori, G., 2009. Human action recognition by semi-latent topic models. IEEE Transactions on Pattern Analysis and Machine Intelligence Special Issue on Probabilistic Graphical Models in Computer Vision 31 (10), 1762–1774.

[47] Wang, Y., Mori, G., 2011. Hidden part models for human action recognition: Probabilistic vs. max-margin. IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (7), 1310–1323.

[48] Weinland, D., Ronfard, R., Boyer, E., 2011. A survey of vision-based methods for action representation, segmentation and recognition. Computer Vision and Image Understanding 115 (2), 224–241.

[49] Xie, D., Todorovi, S., Zhu, S. C., 2013. Inferring “dark matter” and “dark energy” from videos. In: International Conference on Computer Vision.

[50] Yamato, J., Ohya, J., Ishii, K., 1992. Recognizing human action in time-sequential images using hidden markov model. In: Computer Vision and Pattern Recognition.

[51] Yao, A., Gall, J., Gool, L. V., 2010. A hough transform-based voting framework for action recognition. In: Computer Vision and Pattern Recognition.

[52] Yu, T.-H., Kim, T.-K., Cipolla, R., 2010. Real-time action recognition by spatiotemporal semantic and structural forest. In: British Machine Vision Conference.

[53] Zhang, Y., Liu, X., Chang, M.-C., Ge, W., Chen, T., 2012. Spatio-temporal phrases for activity recognition. In: European Conference on Computer Vision.

[54] Zhu, Y., Nayak, N. M., Roy-Chowdhury, A. K., 2013. Context-aware modeling and recognition of activities in video. In: Computer Vision and Pattern Recognition.