Mid-level Features & Spatio-Temporal Context for Activity Recognition

Fei Yuan[a,b,*], Gui-Song Xia[b], Hichem Sahbi[b], Veronique Prinet[a]

[a] *NLPR, Institute of Automation, Chinese Academy of Sciences, Beijing, P.R.C*
[b] *LTCI CNRS, Telecom ParisTech, Paris, France*

Abstract

Local spatio-temporal features have been shown to be effective and robust in order to represent simple actions. However, for high level human activities with long-range motion or multiple interactive body parts and persons, the limitation of low-level features blows up because of their localness. This paper addresses the problem by suggesting a framework that computes mid-level features and takes into account their contextual informations.

First, we represent human activities by a set of mid-level components, referred to as activity components, which have consistent structure and motion in spatial and temporal domain respectively. These activity components are extracted hierarchically from videos, i.e., extracting key-points, grouping them into trajectories and finally clustering trajectories into components. Second, to further exploit the interdependencies of the activity components, we introduce a spatio-temporal context kernel (STCK), which not only captures local properties of features but also considers their spatial and temporal context information. Experiments conducted on two challenging activity recognition datasets show that the proposed approach outperforms standard spatio-temporal features and our STCK context kernel improves further the performance.

Keywords: Activity recognition, mid-level features, activity components, spatio-temporal context kernels

*Corresponding author. Mailing address: 1112, Zidonghua Building, 95 Zhongguancun East Road, Beijing 100190, P.R. China. Tel: (+86)10-62542971, Fax: (+86)10-62647458, Email address: fyuan@nlpr.ia.ac.cn.

1. Introduction

Human activity recognition is one of the most challenging problems in computer vision. By "activity", we refer to a high-level combination of micro-actions with particular spatial and temporal relationships, *e.g.*, hand-shaking, hugging, eating food with a silverware, *etc.* Besides the extrinsic properties of actions, related to camera motion, illumination, occlusion, *etc*, the challenge of activity recognition stems also from their intrinsic aspects related to complex spatio-temporal interactions between micro-actions, for instance moving body parts for multiple persons.

Local interest point features, including 2d corner-based [1, 2] and 3d space-time features [3, 4, 5], have been widely employed for both action and activity[1] recognition. They form sparse and effective action representations usually coupled with machine learning techniques. Their success is also due to their avoidance of pre-processing (such as background subtraction, body modeling and motion estimation) and their robustness to camera motion and illumination changes. Impressive results have indeed been reported in both synthetic and realistic scenarios, see for instance [1, 2, 3, 4, 5, 6, 7, 8]. However, the limitation of these low-level features blows up when used in order to represent complex activities with long-range motions or multiple interactive body parts, since they describe only the local information in a spatio-temporal volume, and their variants (*e.g.*, bag-of-features) usually discard the geometric and the temporal relationships.

Much effort has therefore been undertaken in order to overcome the limitation of local features [1, 8, 9, 10]. Matteo *et al.* [8] extract holistic spatio-temporal features from clouds of interest points accumulated over multiple temporal scales. Andrew *et al.* [1] propose a hierarchical grouping of dense 2d corners in both space and time neighborhood in order to produce an over-complete feature set compound. In this approach, frequently reoccurring feature patterns are hierarchically grouped; as the level of their hierarchy increases, the mined feature compounds become more and more complex,

[1]The terms of action and activity may be used inconsistently when citing related works in the follows, since the differentiation of action and activity is not clear in literatures. However, in this paper, we refer to an activity as a higher-level combination of micro-actions.

sparse and discriminative. In [10], the authors introduce a spatio-temporal match kernel designed to measure the structural similarity between two sets of features extracted from different videos. This kernel explicitly compares temporal relationships (*e.g.*, before and during) as well as spatial relationships (*e.g.*, near and far) between the extracted features in the XY-t space.

Global information is also considered in the literature in order to achieve action recognition [11, 12, 13, 14]. These methods are based on extracting and matching global templates from different videos, but their success is highly dependent on the flexibility of these templates, especially for action categories exhibiting large intra-class variations. In order to improve the flexibility of templates, [13] introduces a deformable action template model based on a learned weighted set of primitives. In [14], the authors split the entire template into parts, which are matched individually. Variants of these approaches rely on mid-level features. For instance, the representative human body model requires a preliminary step of *pose-estimation* and it is based on a pictorial structure model [15]. The latter is a deformable constellation of parts, where each part corresponds to an appearance model and parts are linked using spring-like connections. In [16], the authors introduce a discriminative part-based approach, in which a human action is also modeled as a flexible constellation of parts. The method in [17] is based on a hierarchical action model, in which the bottom of that hierarchy corresponds to local features, while top levels describe constellations of parts; each part is associated to a bag of features, and the relative positions of parts are also modeled.

Other categories of action description and recognition techniques are introduced in the literature. Recently, trajectory-based methods aim to extract long-term motion information [2, 18, 19, 20, 21]. For instance, in [2], each video clip is processed in order to extract trajectories by tracking 2d interest points and these trajectories are described using the bag-of-word model. Another important category of methods seeks to represent the activity in a hierarchical way [22, 23, 24, 25, 26, 27]. In these methods, a complex activity is decomposed into several levels with different semantics, such as "atomic action", "composite action", "single-thread composite events", "multi-thread composite events", *etc.* In [24], complex spatio-temporal events are abstracted into three hierarchies: "primitive events", "single-thread composite events" and "multi-thread composite events", here i) a "primitive event" is defined as a single, coherent unit of movement achieved by one agent, ii)

a "single-thread composite event" corresponds to a consecutive occurrence of multiple primitive events, and iii) a "multi-thread composite event" is a composition of multiple single-thread events with some logical, temporal or spatial relationships. In [26], human activities are classified into three classes: "atomic action", "composite action", and "interaction", based on gestures estimated from different frames; spatial and temporal relationships between atomic actions are also explicitly modeled in order to exploit the structural property of activity. Pinhanez and Bobick [22] propose to represent the temporal structure inherent in human actions using a simplified "past, now, fut" (PNF) network based on Allen's interval algebra, where 13 possible primitive relationships between two time intervals are employed, including equal, before, meet, overlap, during, start, finish, and their inverses. Shi *et al.* [25] introduce Propagation Networks (P-Nets) for representing and recognizing sequential activities, where each activity is represented by partially ordered intervals. Each interval is characterized by information about its duration and its temporal relationship with other intervals.

Context has also been considered as an important cue for action recognition [9, 28, 29]. Authors in [9] propose to learn neighborhood shapes of the space-time features which are discriminative for a given action category, and recursively map the descriptors of the variable-sized neighborhoods into higher-level vocabularies resulting into a hierarchy of space-time configurations. More recently, Wang *et al.* [28] introduce a contextual model in order to capture contextual interactions between interest points. Multiple channels of contextual features for each interest point are computed in multi-scale contextual domains with different shapes, where an individual context is represented by the posterior density of this particular feature class at this pixel location. Then multiple kernel learning is used to select the best combination of channels in a multi-channel SVM classification. In [29], objects and human body parts are considered as mutual context and their interactions are modeled using random fields. Authors in that work, cast the learning task as a structure learning problem, by which the structural connectivity between objects, overall human poses, and different body parts are estimated.

In this paper, we present an approach for action recognition based on mid-level features and spatio-temporal context modeling. First, we represent complex human activity by a set of mid-level features, called *activity components*. We define an activity component as a connected spatio-temporal

part having consistent spatial structure and consistent motion in temporal domain. As we shall see, most of these activity components have physical meanings, for instance, the extracted activity components in "hand-shaking" include "extending arms" and "shaking hands". These activity components are extracted by grouping similar trajectories; key-points are first extracted and tracked frame by frame, in order to form a set of trajectories. Then, trajectories are assigned to different clusters, depending on their appearance and motion, resulting into activity components. Finally, a hierarchical descriptor is used for each activity component in order to encode its appearance, motion and shape informations.
It is worth noticing the difference between our activity components and the part-based models, such as hidden part models [16] and constellation models [17]. In these cited works, parts are hidden or abstract without any physical meaning, in contrast to the proposed components which, in practice, correspond to consistent moving physical body parts. Furthermore, in our work, the extraction of these parts is model-free, *i.e.*, it does not rely on any part-based model, either learned or fixed.

In the second main contribution of this paper, we introduce a spatio-temporal context kernel (STCK) that exploits the structural and the dynamic properties of activity components. We argue that an activity may be decomposed into several components, which interact with each other both in the spatial and the temporal domain. In our preliminary work [30], we already investigated the issue of modeling pairwise relationships between activity components. In contrast to that previous work [30], the similarity of two activities depends on both the local properties of the underlying components and their pairwise and *higher order* spatio-temporal interactions. Moreover, STCK defines a positive definite kernel suitable for SVM learning and classification. Indeed, when plugged into SVMs, it achieves good generalization performance on challenging activity recognition databases.

The rest of this paper is organized as follows. Section 2 gives a detailed description of our approach for extracting and representing mid-level components. In Section 3, we present the spatial-temporal context kernel. We illustrate and interpret the experimental results in Section 4, and finally conclude the paper in Section 5.

2. Activity-Components: a mid-level activity representation

2.1. From trajectories to activity-components

Consider a video $\mathcal{V}$ as a succession of T frames $\{\mathbf{I}_t\}_{t=1}^{T}$ and $\mathcal{P}^{(t)} = \{p_k^{(t)}\}_{k=1}^{K_t}$ the key-points associated with a frame $\mathbf{I}_t$. Let $\tau = (p_i^{(t)}, p_j^{(t+1)}, \ldots, p_k^{(t+\Delta-1)})$ be a trajectory including matched key-points belonging to Δ successive frames and let $\mathcal{T} = \{\tau_n\}_{n=1}^{M}$ be the collection of all trajectories in $\mathcal{V}$. Considering these definitions, a given video $\mathcal{V}$ may be hierarchically defined as

$$\{\mathcal{P}^{(t)}\}_{t=1}^{T} \longrightarrow \mathcal{T} \longrightarrow \mathcal{V}. \tag{1}$$

Existing related works start from this video representation [18, 19, 20, 21], where trajectories are assumed independent and used as elementary activity units. Each trajectory is described by its motion and appearance information, and videos are finally described using the bag-of-word model.

This section presents our first contribution: a novel mid-level representation of activities by relying on video trajectories $\mathcal{T}$. We argue that these trajectories can be partitioned into different clusters where each one corresponds to a meaningful activity unit (such as a moving arm). In what follows, we call these meaningful clusters as *activity-components*. An activity-component (denoted c_ℓ) can be represented as a subset of trajectories

$$c_\ell = \{\tau_{\ell_1}, \tau_{\ell_2}, \ldots, \tau_{\ell_m}\},$$

here $\ell_1, \ell_2, \ldots, \ell_m$ are the lengths of the m trajectories of c_ℓ. Note that any two different activity-components are disjoint, *i.e.*, $c_\ell \cap c_q = \varnothing$ for $\ell \neq q$. Considering $\mathcal{C} = \{c_\ell\}_{\ell=1}^{L}$ as the set of activity-components, the new hierarchical representation of a video $\mathcal{V}$ becomes

$$\{\mathcal{P}^{(t)}\}_{t=1}^{T} \longrightarrow \mathcal{T} \longrightarrow \mathcal{C} \longrightarrow \mathcal{V}. \tag{2}$$

2.2. Extraction of activity-components

For a given video, we consider a bottom-up strategy in order to extract the underlying activity-components. More precisely, trajectories are i) extracted by linking densely sampled key-points, as shown in Figure 1, and ii) grouped into different activity-components according to their appearance and motion properties.

2.2.1. Trajectory extraction

We propose to use densely sampled points from each frame rather than sparse salient points. The latter, used in previous trajectory-based methods [2, 18, 19], do not necessarily capture sufficient details in order to group similar trajectories into meaningful parts. Specifically, we start from dense sampling on a regular grid, where each key-point in that grid is represented by a SIFT descriptor [31]. The correspondence between key-points in successive frame grids is based on the nearest neighbor distance ratio matching [32] and follows two steps:
i) A given key-point in the frame $\mathbf{I}_t$, is constrained to match with key-points located within a spatial window of size $N \times N$ in the subsequent frame $\mathbf{I}_{t+1}$; here N depends on the maximum velocity of motion which is set in practice to 15.
ii) Quasi-static and fast disappearing trajectories are discarded. Precisely, a given trajectory $\tau = (p_i^{(t)}, p_j^{(t+1)}, \ldots, p_k^{(t+\Delta-1)})$ is discarded if its length $\Delta < 5$ or if $\sigma_\tau = \langle \|\partial p\|_2^2 \rangle_{p\in\tau} - \langle \|\partial p\|_2 \rangle_{p\in\tau}^2$ is less than a fixed threshold; here σ_τ is the variance of successive displacements $\|\partial p_i^{(t)}\|_2, \ldots, \|\partial p_k^{(t+\Delta-1)}\|_2$ and $\langle\cdot\rangle$ is an average operator.

2.2.2. Grouping trajectories

Grouping trajectories $\mathcal{T} = \{\tau_n\}_{n=1}^M$ into consistent activity-components, is similar to trajectory-based motion segmentation. As current motion segmentation techniques basically rely on the 2D or the 3D locations of points in trajectories (see for instance [33]), they may fail to obtain satisfactory results on complex scenes including low motion contrast or cluttered background. In contrast to these methods, we use more discriminative features to leverage trajectory description. Specifically, for a given trajectory τ, we consider three informations:

- **Location:** $\{p_\tau^{(t)} = (x_\tau^{(t)}, y_\tau^{(t)})\}_{t=t_\tau^s}^{t_\tau^e}$, where t_τ^s and t_τ^e are the starting and ending times of a trajectory τ;
- **Displacement:** $\{\partial p_\tau^{(t)} = p_\tau^{(t)} - p_\tau^{(t+1)}\}_{t=t_\tau^s}^{t_\tau^e-1}$ are translation vectors between connected points on a trajectory τ;
- **Brightness profile:** this is an N-dim histogram (denoted $h_\tau^{(t)}[n]_{n=1}^N$) defined inside a small patch around $p_\tau^{(t)}$ in the frame $\mathbf{I}_t$.

Note that brightness profile in a given component is usually consistent, and, when combined with location and displacement, it provides a complete information in order to make clustering of trajectories into meaningful components more effective.
Following the previous description, we define our dissimilarity between two trajectories τ, τ' as

$$\frac{\text{dist}(\tau, \tau')}{|t_1 - t_2|}, \tag{3}$$

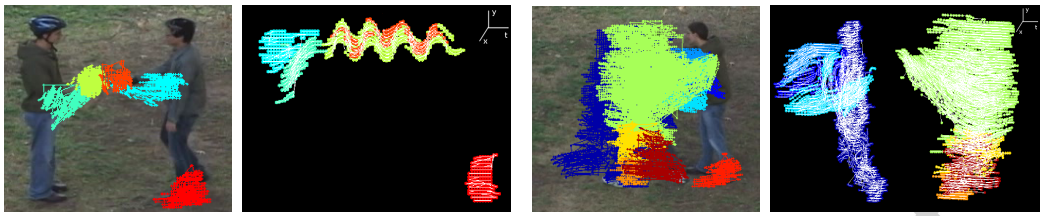
with

$$\begin{aligned}\text{dist}(\tau, \tau') = &\sum_{t=t_1}^{t_2} \left\| p_\tau^{(t)} - p_{\tau'}^{(t)} \right\|_2 \\ &+ \alpha_1 \sum_{t=t_1}^{t_2} \left\| \partial p_\tau^{(t)} - \partial p_{\tau'}^{(t)} \right\|_2 \\ &+ \alpha_2 \sum_{t=t_1}^{t_2} \exp\Big\{ - \sum_{n=1}^{N} \min\left(h_\tau^{(t)}[n], h_{\tau'}^{(t)}[n] \right) \Big\},\end{aligned} \tag{4}$$

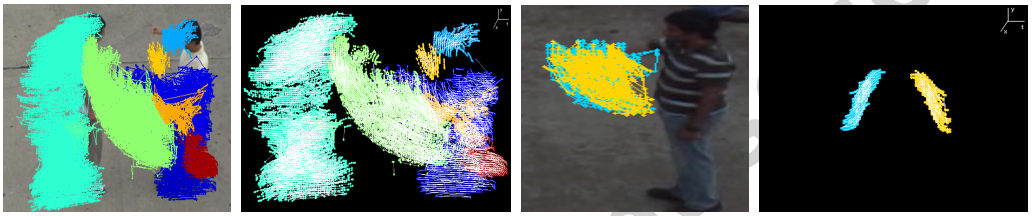
here $t_1 = \max(t_\tau^s, t_\upsilon^s)$, $t_2 = \min(t_\tau^e, t_\upsilon^e)$ are frame-start and frame-end of two overlapping trajectories τ, τ', and $\|\cdot\|_2$ is the L_2-norm. In equation (4), the first term favors trajectories that are spatially close; the second one measures the motion dissimilarity of trajectories; while the third term is the intensity dissimilarity between two trajectories. In practice, the three terms are equally weighted.

Considering the above dissimilarity between all the trajectory pairs in $\mathcal{T}$, we use the graph-based clustering method in [34], in order to partition $\mathcal{T}$ into components. This algorithm has comparable performances wrt others (including normalized-cuts [35]), with an extra advantage of being computationally more efficient. Prior to achieve that clustering, we build a graph where each node corresponds to a trajectory in $\mathcal{T}$ and connections between trajectories are weighted by the dissimilarity defined in Equation (3). This algorithm generates clusters where edges in the same component have relatively low weights, while those in different components have high weights[2].

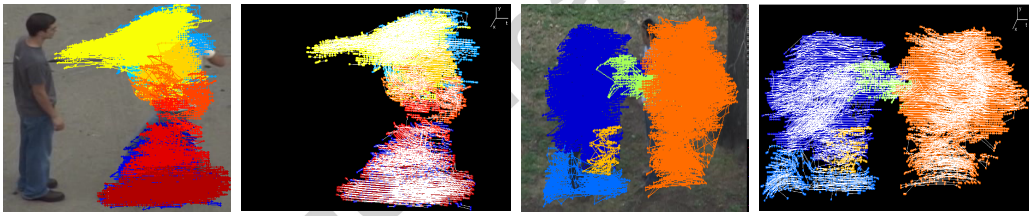
[2]The approach in [34] relies on a threshold T_g which makes it possible to cut edges. In practice, $T_g = 0.6$ generates activity-components with the best recognition rates.



(a) Activity-components of *ShakeHanding* (b) Activity-components of *Hugging*



(c) Activity-components of *Kicking* (d) Activity-components of *Pointing*



(e) Activity-components of *Pushing* (f) Activity-components of *Punching*

Figure 1: Examples of extracted activity-components from the UT-Interaction dataset [36]. Only the largest components are shown. The left one of each subfigure shows the extracted components in X-Y plane, and the right one displays the extracted components in 3-dimensional (X-Y-t) space, where the time information is included (noted that the spatial-temporal locations of points on right pictures may be rotated around X-Y plane for better illustration.). Observe that each activity-component contains a collection of trajectories that are of consistent motion and appearance. Different colors correspond to different activity-components. The points in the same activity-component are displayed in the same color. It is worth noticing the physical meanings of the extracted activity-components. For instance, in Figures 1(a) the most meaningful extracted activity-components are two shaking hands (in red and green) and moving arms (in blue and slight blue), which are also consistent with our human observations.

Figure 1 shows several components obtained on different sequences of 6 activities in the UT-Interaction dataset [36]. These components are shown with different colors and the underlying key-points are displayed with the same color. Note that trajectories, in each extracted activity-component, have consistent motion and appearance. Furthermore, they are easy to interpret; for instance, Figure 1(a) clearly shows hand shaking components (in red and green) and moving arm components (in blue and slight blue).

2.3. Description of activity-components

As shown in Figure 1, extracted activity components may be interpreted as meaningful moving parts, such as moving arms and shaking hands, *etc.* In order to characterize different features of these components, our description is based on three informations: appearance, shape and motion. The first two cues are encoded using the bag-of-word model while the last one is based on a descriptor referred to as *translation matrix*; as will be described later, this translation matrix characterizes the local displacements of points in trajectories and components.

Appearance. A codebook is first generated off-line by clustering a collection of 128 dim SIFTs associated to key-points extracted on a large video set. In practice, we use k-means for clustering and we set the size of the codebook to 300. Then, the appearance information of a given activity-component c is encoded using the bag-of-word model. The latter assigns each key-point in c to an unique cluster and returns a normalized histogram (denoted $h_{appe}(c)$); the latter models the membership distribution of these key-points, in c, to 300 clusters.

Shape. Given an activity-component c, we describe its shape information in successive frames using the bag-of-word model based on shape-context [37]. First, the spatial neighborhood of a given key-point $p^{(t)}$ in c is subdivided into 5×12 cells using the log-polar coordinates (see Figure 2). Then, a *key-point shape descriptor* associated to $p^{(t)}$ is computed; it models the membership of the neighbors of $p^{(t)}$ to these cells.
Similar to appearance, a codebook, of size 300, is built on a collection of *key-point shape descriptors* extracted on a large video set. Again, this codebook is generated using k-means and finally, shape information of a given component c is described by a normalized histogram (denoted $h_{shape}(c)$) that models the membership distribution of key-point shape descriptors, in c, to 300 clusters.

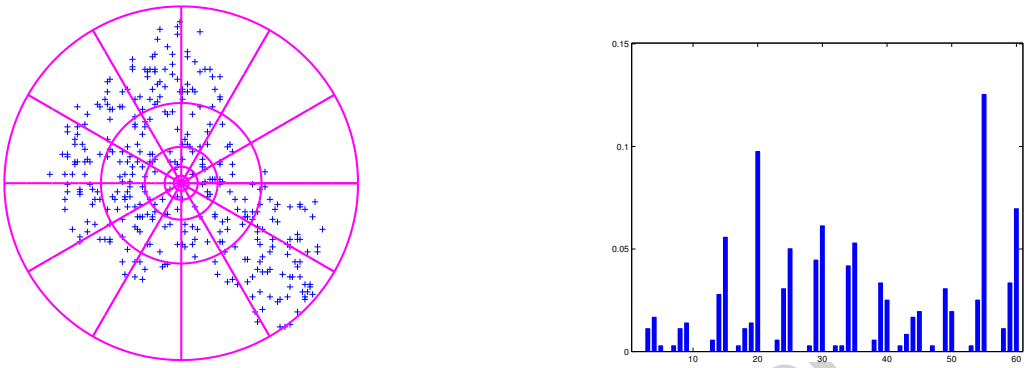


Figure 2: Illustration of a key-point shape descriptor in an activity-component. The spatial neighborhood of $p^{(t)}$ in frame I_t is subdivided into 5×12 cells using the log-polar coordinates. A histogram is used to describe the quantized spatial distribution of its neighbors according to the log-polar partitions.

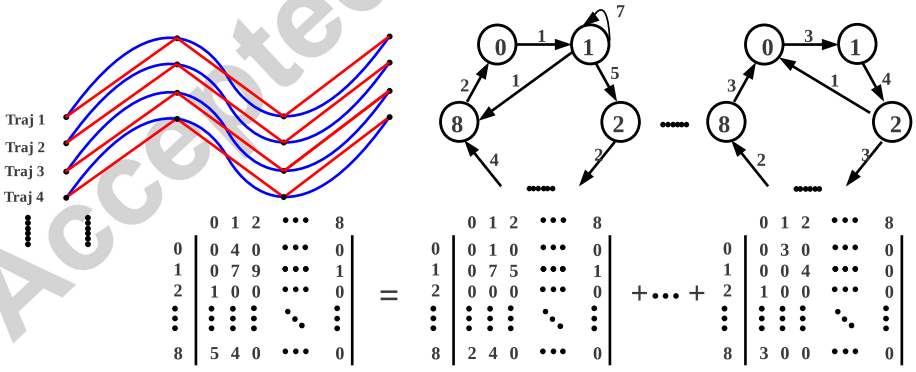


Figure 3: Illustration of the motion descriptor of an activity-component. For each trajectory in an activity-component, we compute its transition matrix based on its line segments. The final motion descriptor is the average of those of trajectories in an activity-component.

Motion. Given an activity component c, we introduce a descriptor, referred to as *translation matrix*, in order to characterize the motion information of c. This descriptor was inspired by [2, 18] and its principle consists in smoothing and extracting curvatures on each trajectory τ of c using anisotropic diffusion [38]. The local extrema of these curvatures correspond to both changes of speed and direction in c.
Following the approach in [39], each trajectory τ, in c, is split into several segments joining the local curvature extrema. The orientation of these segments is quantized into S states, including a "no-motion" one. Thereby, for each trajectory τ, a sparse transition matrix with $S \times S$ states is computed (in practice $S = 9$). The transition matrices of all the trajectories in c are paved together in a normalized motion descriptor (denoted as $\psi_{motion}(c)$). Figure 3 shows an example of this descriptor.

2.4. Activity description

For each activity-component c, the three descriptors mentioned earlier are normalized and weighted in order to form one global component descriptor $f(c) = \langle \alpha_{appe} h_{appe}(c), \alpha_{shape} h_{shape}(c), \alpha_{motion} \psi_{motion}(c) \rangle$, here $\alpha_{appe} + \alpha_{shape} + \alpha_{motion} = 1$. The choice of these weights is described in Sec. 4.2. Finally, an activity is encoded by i) quantizing its component descriptors using a learned codebook and ii) computing a histogram which counts the occurrence of the quantized component descriptors according to that codebook.

3. Spatio-temporal context for activity recognition

The activity description introduced so far relies on a strong assumption that activity-components are independent, so this assumption ignores the structural properties of human activity. These properties, including spatial, temporal and causal relationships, are known to be important in activity recognition [22, 23, 24, 25, 26, 27]. In this and the following sections, we model these spatio-temporal structural properties and we study their influence in activity recognition.

3.1. Spatio-temporal context

The description of each activity component c is extended with a tuple

$$(f(c),\ \overline{x}(c),\ \overline{y}(c),\ \overline{t}(c),\ s_{xy}(c),\ s_t(c)),$$

here $f(c)$ stands for the feature of c (introduced in Section 2.3), $\overline{x}(c), \overline{y}(c), \overline{t}(c)$ correspond to the centroid of key-points in c, and $s_{xy}(c), s_t(c)$ denote respectively the spatial and temporal scale of c. In practice, $s_{xy}(c)$ is set to the average spatial distance of key-points in c with respect to $(\overline{x}(c), \overline{y}(c))$ (i.e., standard deviation), and $s_t(c)$ is similarly defined as the average temporal distance of endpoints in all trajectories of c with respect to $\overline{t}(c)$.

Let $\mathcal{C}_v$ be the union of all components in a given video. We define the asymmetric spatio-temporal neighborhood of a given component $c \in \mathcal{C}_v$ as

$$\mathcal{N}(c) = \Big\{ c' : c' \in \mathcal{C}_v,\ d_{xy}(c,c') < \alpha_{xy} \cdot s_{xy}(c),\ d_t(c,c') < \alpha_t \cdot s_t(c) \Big\}, \quad (5)$$

with

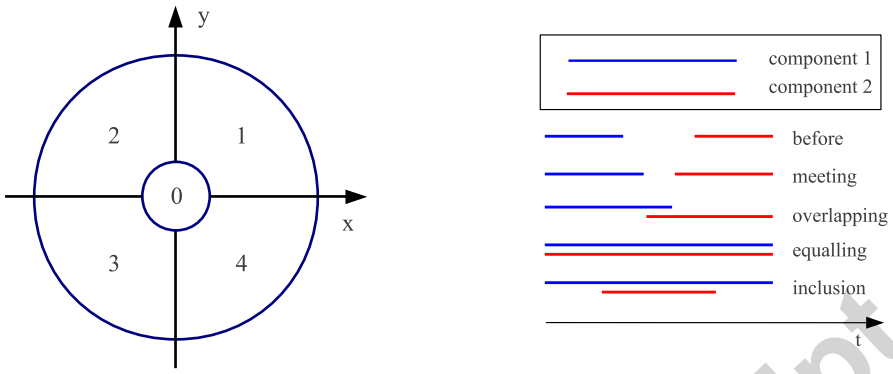
$$d_{xy}(c,c') = \left(\left\| \overline{x}(c) - \overline{x}(c') \right\|_2^2 + \left\| \overline{y}(c) - \overline{y}(c') \right\|_2^2 \right)^{\frac{1}{2}}$$
$$d_t(c,c') = \left\| \overline{t}(c) - \overline{t}(c') \right\|_2,$$

and α_{xy}, α_t being the parameters of a cylinder (of radius $\alpha_{xy} \cdot s_{xy}(c)$ and length $2\alpha_t \cdot s_t(c)$) enclosing components in $\mathcal{N}(c)$. These parameters are set sufficiently large ($\alpha_{xy} = 4$ and $\alpha_t = 6$ in practice) in order to avoid sparse neighborhoods.

Each component $c' \in \mathcal{N}(c)$ is assigned i) to an unique cell in the spatial neighborhood (see Figure 4(a), left), depending on its relative spatial location w.r.t c and ii) to an unique state in the temporal neighborhood (see Figure 4(a), right), depending on the relative temporal position of the extremities of c and c'. In practice, five temporal states are combined with 5 spatial cells resulting into a final spatio-temporal neighborhood of 25 cells (denoted $\{\mathcal{N}_r(c)\}_{r=0}^{24}$). The latter reflect interactions between different components for a better modeling of context in activity recognition as shown in the subsequent sections.

3.2. Kernel design for context-dependent activity recognition

In this section, we adapt the context-dependent kernel, initially introduced in Sahbi *et al.* [40], in order to capture spatial *as well as* temporal context between components. The resulting kernel is referred to as spatio temporal context kernel (STCK).



(a) Spatial relationships (b) Temporal relationships

Figure 4: Illustration of spatio-temporal relationships. In (a), the spatial relationships are quantized into 5 states. In (b), the locations of lines stand for the temporal locations of components, and the lengths of lines stand for the temporal ranges components cover.

Let $\mathcal{C} = \cup_v \mathcal{C}_v$ be the union of all the components in the available (training and testing) videos. Define our STCK kernel $\mathbf{K}$ as a Gram matrix where each entry $\mathbf{K}_{c,c'}$ provides a similarity between $c,\ c' \in \mathcal{C}$.
Using $\mathbf{K}$, one may define the convolution kernel $\mathbf{K}_{conv}$, between any two activities $\mathcal{A}_p = \{c_i^p\}_{i=1}^m$, $\mathcal{A}_q = \{c_j^q\}_{j=1}^n$ as

$$\mathbf{K}_{conv}(\mathcal{A}_p,\ \mathcal{A}_q) = \frac{1}{|\mathcal{A}_p|.|\mathcal{A}_q|} \sum_{c \in \mathcal{A}_p} \sum_{c' \in \mathcal{A}_q} \mathbf{K}_{c,c'},$$

note that when $\mathbf{K}$ is positive (semi) definite, the convolution kernel $\mathbf{K}_{conv}$ will also be positive (semi) definite. Our goal is to design $\mathbf{K}$ by taking into account the local properties of components (i.e., their features) as well as their spatio-temporal context.

Let's define, for each cell r in the neighborhood system $\{\mathcal{N}_r(c)\}_{r=0}^{24}$, an adjacency matrix $\mathbf{P}^r$; here $\mathbf{P}^r_{c,c'} = 1_{\{c' \in \mathcal{N}_r(c)\}}$. Now, we propose to design the kernel $\mathbf{K}$ by minimizing

$$\min_{\mathbf{K} \geq 0, \|\mathbf{K}\|_1 = 1} \mathbf{tr}\big(\mathbf{K}\mathbf{D}'\big) + \beta \mathbf{tr}\big(\mathbf{K} \log \mathbf{K}'\big) - \alpha \sum_{r=0}^{24} \mathbf{tr}\big(\mathbf{K}\mathbf{P}^r(\mathbf{P}^r\mathbf{K})'\big), \qquad (6)$$

here $\alpha, \beta \geq 0$, $\mathbf{D}$ is a dissimilarity matrix between (local) component features (i.e., $\mathbf{D}_{c,c'} = ||f(c) - f(c')||_2$), with $\mathbf{D}'$ being the transpose of $\mathbf{D}$ and

$\mathbf{tr}(\cdot)$ the trace operator. The operations "log"(natural) and "$\geq$" are applied individually to each entry of $\mathbf{K}$.

The first term in the above constrained minimization problem measures the fidelity of component alignments, i.e., the quality of matching components using only their local features. The second term is a regularization criterion that constrains the probability distribution $\{\mathbf{K}_{c,c'}\}_{c,c' \in \mathcal{C}}$ to be flat when no any priori knowledge about the aligned components is available. The third term captures the spatio-temporal context, where a high value of $\mathbf{K}_{c,c'}$ should imply high kernel values between components in $\mathcal{N}_r(c) \times \mathcal{N}_r(c')$.

Solution. The optimization problem in Equation (6) admits a positive definite kernel solution $\tilde{\mathbf{K}}$, which is the limit of

$$\mathbf{K}^{(\eta)} = \frac{G(\mathbf{K}^{(\eta-1)})}{\left\| G(\mathbf{K}^{(\eta-1)}) \right\|_1},$$

with

$$G(\mathbf{K}) = \exp\left\{ -\frac{\mathbf{D}}{\beta} + \frac{\alpha}{\beta} \sum_{r=0}^{24} \left(\mathbf{P}^r \mathbf{K} \mathbf{P}'^r + \mathbf{P}'^r \mathbf{K} \mathbf{P}^r \right) \right\}$$

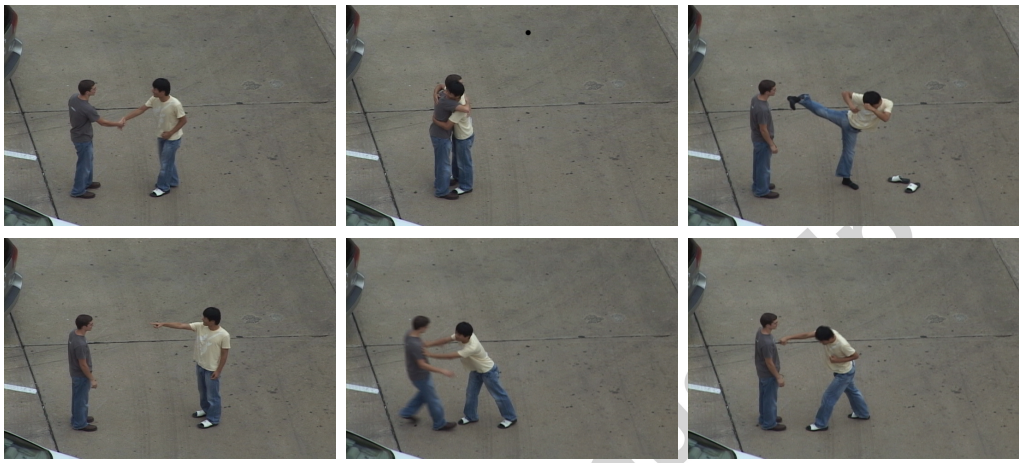
$$\mathbf{K}^{(0)} = \frac{\exp(-\mathbf{D}/\beta)}{\left\| \exp(-\mathbf{D}/\beta) \right\|_1}.$$

Detailed proof of this solution and its convergence to a positive definite fixed-point (with respect to the setting of α, β) is out of the scope of this paper and may be found in [40], for the same family of kernel solutions. Note that, in practice, this kernel solution usually converges after 3 iterations only.

Finally, by replacing $\tilde{\mathbf{K}}$ into the convolution kernel $\mathbf{K}_{conv}$, we obtain a Gram matrix that characterizes the similarity between any two activities $\mathcal{A}_p$, $\mathcal{A}_q$. When plugging this kernel in SVMs, we obtain an effective activity recognizer as shown in the following experiments.

4. Experiments

In this section, we evaluate the performance of our proposed method in two aspects: we first compare activity components with local space-time features [3, 4] and trajectory-based features [19], then we show the advantage of spatio-temporal context in activity recognition when using the STCK kernel.



(a) Snapshots of video sequences in the UT-Interaction Dataset [36], containing 6 classes of activities and 20 samples in each class.

(b) Snapshots of video sequences in the Rochester Activities Dataset [19], containing 10 classes of activities and 15 videos in each class.

Figure 5: Snapshot examples of video sequences in two activity datasets: the UT-Interaction Dataset [36] and the Rochester Activities Dataset [19].

4.1. Experimental Setups

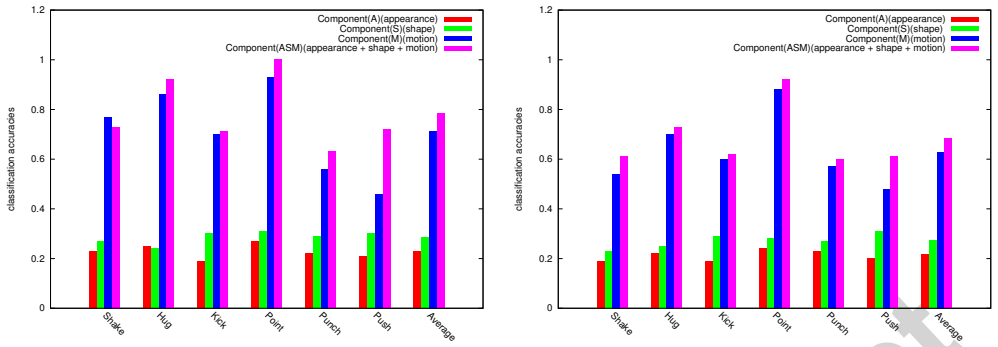
All the comparisons were implemented on two recent activity datasets: the UT-Interaction collection [36] including complex interactions and the Rochester Activities dataset [19] containing more complex daily activities.

UT-Interaction dataset. It contains 6 classes of human-human interactions: *shake-hands, point, hug, push, kick* and *punch*. Each class has 20 video sequences which are divided into two different sets: (1) the first one, named set 1, includes 10 videos taken on a parking lot with slightly different zoom rates and mostly static background with little camera jitter; (2) the other 10 video sequences, named set 2, are taken on a lawn in a windy day with a slightly moving background, *e.g.* tree moves, and the videos contain more camera jitters. Note that background, scale, and illumination of the videos in each set are different. Figure 5(a) shows one snapshot for each class.

Rochester Activities dataset. This dataset contains 10 classes of daily living activities: *answering a phone, chopping a banana, dialing a phone, drinking water, eating a banana, eating snack chips, looking up a phone number in a telephone book, peeling a banana, eating food with silverware* and *writing on a white board*. Each of these activities contains 15 different videos that include five different persons of different shapes, sizes, genders, and ethnicities[3]. Figure 5(b) shows one snapshot for each class on the dataset.

In order to evaluate and fairly compare the performance, we use the same experimental setting as in [19] and [36]. Specifically, on the Rochester Activities Dataset, 12 video sequences taken by four subjects (out of the five) are used for training, and the remaining 3 videos for testing. The experiments are repeated five times. As for the UT-Interaction dataset, a 10-fold leave-one-out cross validation is used for each set (i.e., each time, 9 samples are used for training and one for testing). The performance of different methods is shown using the average recognition rates.

[3]It has been reported [19] that using only the motion information is not sufficient for distinguishing these activities and some other information, such as appearance descriptions, should be taken into account.



(a) UT-Interaction dataset set 1 (b) UT-Interaction dataset set 2

Figure 6: Contribution of appearance, shape and motion features on UT-Interaction dataset.

4.2. Comparison

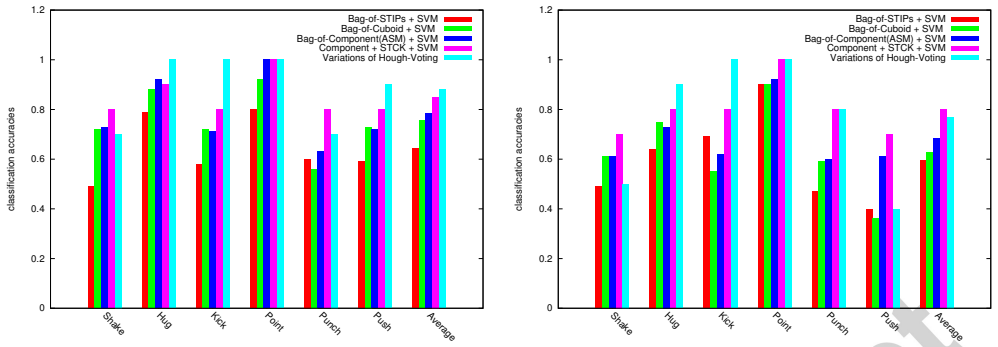
In order to evaluate the effectiveness of the proposed mid-level activity components, we consider three settings:

- **Setting 1 (Bag-of-Components + SVMs):** first, we generate a codebook of activity-components using k-means, then we represent each activity with the bag-of-component model (as described in Section 2), finally we train a SVM classifier for each activity class using the RBF kernel. As in [36], codebooks with 10 different sizes are generated, and the underlying performance is averaged.

- **Setting 2 (Components + STCK kernel + SVMs):** for each activity class, we train a SVM classifier using the STCK kernel . According to [40], higher values of α, in STCK, result into better performance but following [40] this parameter should be upper-bounded in order to guarantee convergence to a fixed-point. Note also that the setting of β is the one used for the RBF kernel, as the latter corresponds to the left-hand side term in STCK. In practice, we found that the best setting of α, β is respectively 20 and 0.5.

- **Setting 3 (Components + Context-free kernels + SVMs):** the setting is exactly the same as the previous one, but $\alpha = 0$.

Comparison using Setting 1: first, we evaluate the influence of component features on performance. Figure 6 shows activity recognition accuracy



(a) UT-Interaction dataset set 1 (b) UT-Interaction dataset set 2

Figure 7: Comparison of classification accuracies of different features and models on UT-Interaction dataset. Our activity-components outperform the local STIPs feature [3] and Cuboid feature [4]. Moreover, our *Component + STCK + SVM* model is comparable to state-of-the-art results.

of *Bag-of-Component + SVMs* using different features (appearance, shape and motion) and their combination on the UT-Interaction dataset (sets 1 and 2). The average classification accuracies of appearance, shape and motion features on the two sets are 22%, 27.8% and 67.1% respectively; and the performance of their combination reaches 73.3%, and outperforms our original motion feature, in [30], by 6.2 points. Compared to appearance and shape features, it is clear from experiments, that the contribution of motion is substantial and this corroborates the fact that motion is the most critical feature for activity recognition.

Second, we compare the proposed mid-level activity components against local spatio-temporal features (*STIPs*) [3] and *Cuboid* [4]. The *baseline* results of *Bag-of-STIPs + SVM* and *Bag-of-Cuboid + SVM* are taken directly from the ICPR 2010 Contest about Semantic Description of Human Activities [41], and of course implemented by the participants. Figure 7 shows these results and comparison on the UT-Interaction dataset. It is clear that our method (referred to as *Component(ASM)*) outperforms *STIPs* and *Cuboid*; indeed the average accuracy of our method on set 1 (resp. set 2) reaches 78.3% (resp. 68.2%) while for *STIPs* and *Cuboid* the accuracy does not exceed 64.2% (resp. 59.7%) and 75.5% (resp. 62.7%). According to these results, activity-components are more discriminative as they model appearance, shape and motion in larger spatio-temporal areas. Notice also that,

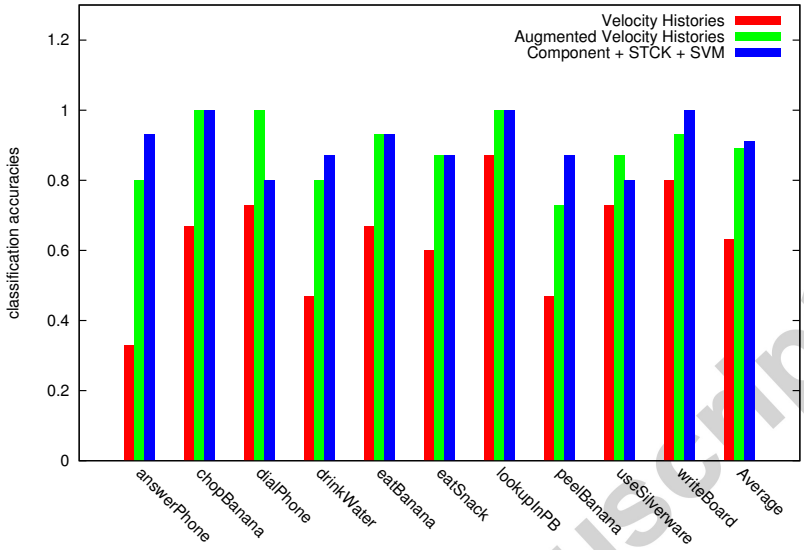


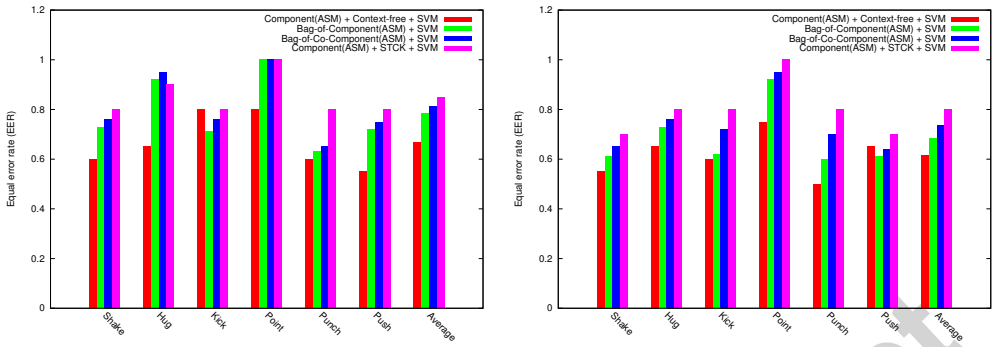
Figure 8: Comparison of classification accuracies of different features on Rochester Activities Dataset.

compared to dense cuboid and sparse STIPs, the proposed mid-level activity components are much sparser. For instance, in different sequences of handshaking, the number of activity components ranges from 30 to 60.

Comparison using Setting 2: when combining *Component + STCK + SVM*, the average accuracy of our method (on sets 1 and 2) reaches 82.5% which is comparable to Hough-Voting in [42]. Nevertheless, our method has an extra advantage of being similarly performant on different activity classes compared to Hough-Voting whose performance changes significantly from one class to another (see for instance class *Push* in set 2).

We also compare our method against trajectory-based features [19]; figure 8 shows different accuracies on the Rochester Activities Dataset. Overall, our mid-level activity-components with STCK outperform the velocity history feature by 28 points, and even the augmented velocity history feature[4] by 2 points. Furthermore, our results are at least comparable to (or bet-

[4]Note that the augmented velocity history feature improves the velocity history of tracked key-points using a set of rich information, such as absolute and relative positions of human face, appearances and color.



(a) UT-Interaction dataset set 1 (b) UT-Interaction dataset set 2

Figure 9: Comparison of classification accuracies of different context models on UT-Interaction datasets.

ter than) the augmented velocity history feature on almost all the activity classes, excepting *Dial Phone* and *Use Silverware*.

Comparison using Setting 3: finally, we show the performance of our method with a particular setting of α (i.e., $\alpha = 0$) and this kernel will be referred to as context-free kernel (CFK). We plug CFK into SVM and we compare it against *the bag-of-components model*, *the STCK kernel*, and the bag-of-co-components method [30]. The latter proposed in our previous work [30] is based on pairwise interactions between components in mutual contexts. Figures 9, 10 show the classification accuracies in the UT-Interaction and the Rochester Activities datasets, respectively. It is clear that performance of STCK is consistently better than the others on the two sets.

Compared to the context-free kernel, STCK shows much better performance. Indeed, its average accuracy improves by 18.3 points compared to the context-free kernel on UT-Interaction dataset (set 1). STCK outperforms the bag-of-components, by 8.75 and 8 points respectively on the UT-Interaction and the Rochester Activities datasets. Note also that STCK outperforms the bag-of-co-components, in [30]. Even though the bag-of-co-components approach models the mutual interaction (context) between components, its influence is restricted to pairwise interactions, while STCK is able to recursively diffuse the similarity from/to larger and more influencing contexts, so it exploits the pairwise (local) as well as higher order interactions (resulting from recursion).

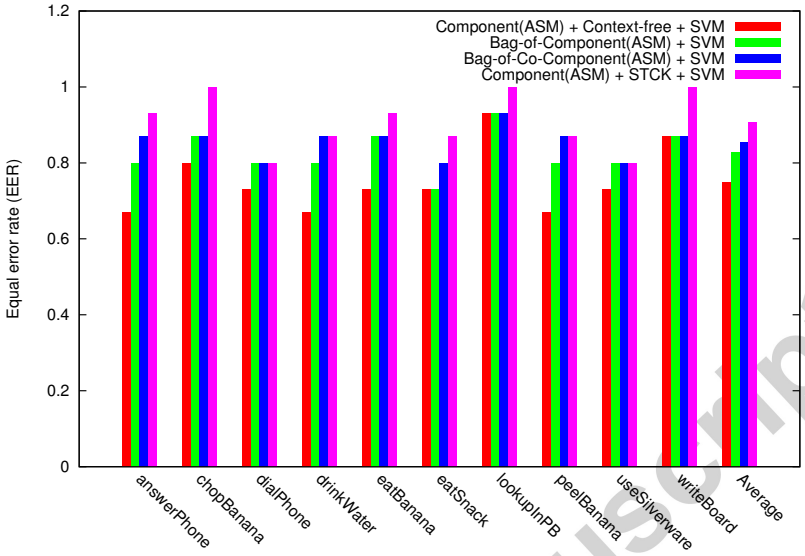


Figure 10: Comparison of classification accuracies of different context models on Rochester Activities Dataset.

5. Conclusion

In this paper, we introduce a novel activity recognition method based on two contributions. In the first contribution, we present a new mid-level activity representation based on clustering trajectories into consistent primitives referred to as activity components. In the second contribution, we model interactions between these components through a spatio-temporal context kernel (STCK) very suitable for action recognition when plugged into support vector machines. Evaluations and comparisons show that these contributions (when taken individually and when combined) bring substantial gain with respect to baseline as well as related state of the art activity recognition techniques.

Indeed, compared to local space-time features, the proposed mid-level activity-components are more discriminant, as they include appearance, shape and motion information in larger spatio-temporal areas. In contrast to trajectory-based methods, the mid-level activity components not only capture motion information but also appearance and shape. Moreover, when combined with our STCK, the latter is able to diffuse the similarity not only to immediate component neighbors but also to larger and more influencing contexts result-

ing from the recursive definition of STCK.

As a future work, we are currently studying other algorithms in order to extract activity components corresponding to better *physically interpretable moving parts* while being robust to different sources of variability (background, illumination, etc.). We believe that this issue may bring further substantial improvements.

[1] A. Gilbert, J. Illingworth, R. Bowden, Fast realistic multi-action recognition using mined dense spatio-temporal features, in: Proc. Int. Conference on Computer Vision (ICCV), 2009.

[2] P. Siva, T. Xiang, Action detection in crowd, in: Proc. British Machine Vision Conference (BMVC), 2010.

[3] I. Laptev, , T. Lindeberg, On space-time interest points, in: Proc. Int. Conference on Computer Vision (ICCV), 2003.

[4] P. Dollar, V. Rabaud, G. Cottrell, S. Belongie, Behavior recognition via sparse spatio-temporal features, in: Proc. Int. Conference on VS-PETS, 2005.

[5] G. Willems, T. Tuytelaars, L. V. Gool, An efficient dense and scale-invariant spatio-temporal interest point detector, in: Proc. European Conference on Computer Vision (ECCV), 2008.

[6] C. Schuldt, I. Laptev, B. Caputo, Recognizing human actions: A local svm approach, in: Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR), 2004.

[7] I. Laptev, M. Marszałek, C. Schmid, B. Rozenfeld, Learning realistic human actions from movies, in: Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR), 2008.

[8] M. Bregonzio, S. Gong, T. Xiang, Recognising action as clouds of space-time interest points, in: Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR), 2009.

[9] A. Kovashka, K. Grauman, Learning a hierarchy of discriminative space-time neighborhood features for human action recognition, in: Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR), 2010.

[10] M. S. Ryoo, J. K. Aggarwal, Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities, in: Proc. Int. Conference on Computer Vision (ICCV), 2009.

[11] A. F. Bobick, J. W. Davis, I. C. Society, I. C. Society, The recognition of human movement using temporal templates, IEEE Trans. Pattern Anal. Mach. Intell. 23 (3) (2001) 257–267.

[12] L. Gorelick, M. Blank, E. Shechtman, M. Irani, R. Basri, Actions as space-time shapes, IEEE Trans. Pattern Anal. Mach. Intell. 29 (12) (2007) 2247–2253.

[13] B. Yao, S. C. Zhu, Learning deformable action templates from cluttered videos, in: Proc. Int. Conference on Computer Vision (ICCV), 2009.

[14] Y. Ke, R. Sukthankar, M. Hebert, Event detection in crowded videos, in: Proc. Int. Conference on Computer Vision (ICCV), 2007.

[15] P. F. Felzenszwalb, D. P. Huttenlocher, Efficient matching of pictorial structures, in: Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR), 2000.

[16] Y. Wang, G. Mori, Learning a discriminative hidden part model for human action recognition, in: Advances in Neural Information Processing Systems (NIPS), 2008.

[17] J. Niebles, L. Fei-Fei, A hierarchical model of shape and appearance for human action classification, in: Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR), 2007.

[18] J. Sun, X. Wu, S. Yan, L. F. Cheong, T.-S. Chua, J. Li, Hierarchical spatio-temporal context modeling for action recognition., in: Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR), 2009.

[19] R. Messing, C. Pal, H. Kautz, Activity recognition using the velocity histories of tracked keypoints, in: Proc. Int. Conference on Computer Vision (ICCV), 2009.

[20] P. Turaga, R. Chellappa, Locally time-invariant models of human activities using trajectories on the grassmannian, in: Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR), 2009.

[21] H. Wang, A. Kläser, C. Schmid, L. Cheng Lin, Action Recognition by Dense Trajectories, in: Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR), 2011.

[22] C. Pinhanez, A. Bobick, Human action detection using pnf propagation of temporal constraints, in: Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR), 1998.

[23] J. M. Siskind, Grounding the lexical semantics of verbs in visual perception using force dynamics and event logic, Journal of Artificial Intelligence Research 15 (2001) 31–90.

[24] R. Nevatia, T. Zhao, S. Hongeng, Hierarchical language-based representation of events in video streams, in: Computer Vision and Pattern Recognition Workshop, 2003. CVPRW '03. Conference on, 2003.

[25] Y. Shi, Y. Huang, D. Minnen, A. Bobick, I. Essa, Propagation networks for recognition of partially ordered sequential action, in: Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR), 2004.

[26] M. Ryoo, J. Aggarwal, Recognition of composite human activities through context-free grammar based representation, in: Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR), 2006.

[27] A. Gupta, P. Srinivasan, J. Shi, L. Davis, Understanding videos, constructing plots learning a visually grounded storyline model from annotated videos, in: Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR), 2009.

[28] J. Wang, Z. Chen, Y. Wu, Action recognition with multiscale spatio-temporal contexts, in: Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR), 2011.

[29] B. Yao, L. Fei-Fei, Modeling mutual context of object and human pose in human-object interaction activities, in: Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR), 2010.

[30] F. Yuan, J. Yuan, V. Prinet, Middle-level representation for human activities recognition: the role of spatio-temporal relationships, in: ECCV'10 Workshop on Human Motion: Understanding, Modeling, Capture and Animation, 2010.

[31] D. G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.

[32] K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors, IEEE Trans. Pattern Anal. Mach. Intell. 27 (10) (2005) 1615–1630.

[33] S. Rao, R. Tron, R. Vidal, Y. Ma, Motion segmentation in the presence of outlying, incomplete, or corrupted trajectories, IEEE Trans. Pattern Anal. Mach. Intell. 99 (10) (2009) 1832 –1845.

[34] P. F. Felzenszwalb, D. P. Huttenlocher, Efficient graph-based image segmentation, Int. J. Comput. Vis. 59 (2) (2004) 167–181.

[35] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 888 –905.

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

[37] S. Belongie, G. Mori, J. Malik, Matching with shape contexts, in: Proc. IEEE Workshop on Content-based access of Image and Video-Libraries, 2000.

[38] P. Perona, J. Malik, Scale-space and edge detection using anisotropic diffusion, IEEE Trans. Pattern Anal. Mach. Intell. 12 (7) (1990) 629–639.

[39] C. Rao, A. Yilmaz, M. Shah, View-invariant representation and recognition of actions, Int. J. Comput. Vis. 50 (2) (2002) 203–226.

[40] H. Sahbi, J.-Y. Audibert, J. Rabarisoa, R. Keriven, Context-dependent kernel design for object matching and recognition, in: Proc. Int. Conference on Computer Vision and Pattern Recognition (CVPR), 2008.

[41] M. S. Ryoo, C.-C. Chen, J. K. Aggarwal, A. Roy-Chowdhury, An overview of contest on semantic description of human activities (sdha) 2010, http://cvrc.ece.utexas.edu/SDHA2010/Human_Interaction.html (2010).

[42] D. Waltisberg, A. Yao, J. Gall, L. Van Gool, Variations of a hough-voting action recognition system, in: Proc. Int. Conference on Pattern Recognition (ICPR) contest on Semantic Description of Human Activities (SDHA), 2010.