First-Person Activity Recognition: What Are They Doing to Me?

M. S. Ryoo and Larry Matthies
Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA
{mryoo,lhm}@jpl.nasa.gov

Abstract

This paper discusses the problem of recognizing interaction-level human activities from a first-person viewpoint. The goal is to enable an observer (e.g., a robot or a wearable camera) to understand 'what activity others are performing to it' from continuous video inputs. These include friendly interactions such as 'a person hugging the observer' as well as hostile interactions like 'punching the observer' or 'throwing objects to the observer', whose videos involve a large amount of camera ego-motion caused by physical interactions. The paper investigates multi-channel kernels to integrate global and local motion information, and presents a new activity learning/recognition methodology that explicitly considers temporal structures displayed in first-person activity videos. In our experiments, we not only show classification results with segmented videos, but also confirm that our new approach is able to detect activities from continuous videos reliably.

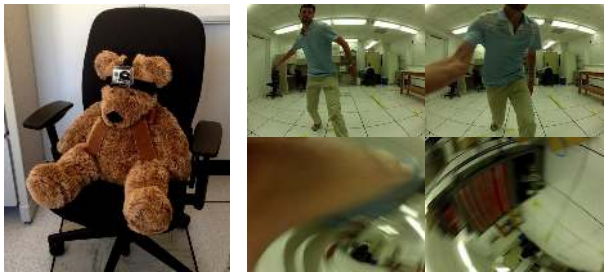
1. Introduction

In the past decade, there has been a large amount of progress in human activity recognition research. Researchers not only focused on developing reliable video features robust to noise and illumination changes [14, 3, 7], but also proposed various types of hierarchical approaches to recognize high-level activities with multiple actors [12, 9, 17] and even group activities [13]. State-of-the-art approaches are obtaining successful results, showing their potential for many real-world applications including visual surveillance.

However, most of these previous works focused on activity recognition from a 3rd-person perspective (i.e., viewpoint). The camera, which is usually far away from actors, analyzed what humans are doing to each other without getting involved in the activities (e.g., 'two persons punching each other'). This 3rd-person activity recognition paradigm is insufficient when the observer itself is involved in interactions, such as 'a person attacking the camera'. In these videos, the camera undergoes a huge amount of ego-motion such as spinning and falling down (Figure 1 (b)), making its videos very different from previous 3rd-person videos. What we require is the ability to recognize physical and social human activities targeted to the observer (e.g., a wearable camera or a robot) from its viewpoint: *first-person human activity recognition*.

This paper discusses the new problem of recognizing interaction-level human activities from first-person videos. Even though there has been previous attempts to recognize activities from first-person videos [6, 4, 10], they focused on recognition of ego-actions of the person wearing the camera (e.g., 'riding a bike' or 'cooking'). There also are works on recognition of gesture-level motion to the sensor [16] and analysis of face/eye directions [5], but recognition of high-level activities involving physical interactions (e.g., 'a person punching the camera') from a first-person viewpoint has not been explored in depth. Recognition of 'what others are doing to the observer' from its own perspective is not only crucial for any surveillance or military systems to protect themselves from harmful activities by hostile humans, but also is very important for friendly human-robot interaction scenarios (e.g., 'shaking hands with the robot') by making the robot socially aware of what people want to do to it.

In this paper, we introduce our new dataset composed of first-person videos collected during humans' interaction with the observer, and investigate features and approaches necessary for the system to understand activities from such videos. We particularly focus on two aspects of first-person activity recognition, aiming to provide answers to the following two questions: (1) What features (and their combination) do we need to recognize interaction-level activities from first-person videos? (2) How important is it to consider temporal structure of the activities in first-person recognition? We first discuss extraction of global motion descriptors capturing ego-motion of the observer (often caused by interactions such as 'picking up the observer') and local motion descriptors describing body movements of an interacting person (generated by activities such as 'throwing an object'), and describe multi-channel kernels to combine them for the recognition. Next, we present a new kernel-based activity recognition approach that explicitly



(a) Our observer setup (b) Video snapshots from the observer

Figure 1. Picture of our setting, and its example observations obtained during a person punching it. The humanoid was placed on a rolling chair to enable its operator emulate translation movements.

learns structures of human activities from training videos. Our approach learns sub-events composing an activity and how they are temporally organized, obtaining superior performance in first-person activity recognition.

1.1. Related works

Computer vision researchers explored various human activity recognition approaches since early 1990's [1]. In the past 5 years, approaches utilizing sparse spatio-temporal features capturing local motion [14, 3, 7] have been particularly successful thanks to their reliability under noise. In addition, there has been hierarchical approaches dividing activities into sub-events for their better recognition [12, 9, 17]. However, these previous human activity recognition works detected human behaviors from videos with third-person viewpoints (e.g., videos captured using surveillance cameras or movie scenes), and did not focus on the first-person recognition of activities. Even though there are recent works on first-person action recognition from wearable cameras [6, 4, 10, 5], research on recognition of physical human interactions targeted to the camera and their influences on the camera movements has been very limited.

Up to our knowledge, this paper is the first paper to recognize human interactions from first-person videos.

2. First-person video dataset

We constructed a new first-person video dataset containing interactions between humans and the observer. We attached a GoPro camera to the head of a humanoid model (Figure 1), and asked human participants to interact with the humanoid by performing activities. This humanoid can be viewed as a model robot. In order to emulate the mobility of a real robot, we also placed wheels below the humanoid and made an operator to move the humanoid by pushing it from behind. The dataset serves as a recognition benchmark.

For the video collection, our robot was placed in 5 different environments with distinct background and lighting conditions. A total of 8 participants wearing a total of 10 different clothings participated in our experiments. The participants were asked to perform 7 different types of activities, including 4 positive (i.e., friendly) interactions with the observer, 1 neutral interaction, and 2 negative (i.e., hostile) interactions. 'Shaking hands with the observer', 'hugging the observer', 'petting the observer', and 'waving a hand to the observer' are the four friendly interactions. The neutral interaction is the situation where two persons have a conversation about the observer while occasionally pointing it. 'Punching the observer' and 'throwing objects to the observer' are the two negative interactions. Videos were recorded continuously during human activities where each video sequence contains 0 to 3 activities.

Figure 2. Seven classes of human activities in our dataset.

Figure 2 shows example snapshots of human activities in our dataset. The videos are in 320*240 image resolution, 30 frames per second. Notice that the robot (and its camera) is not stationary and it displays a large amount of ego-motion in its videos particularly during the human activity. For instance, in the case of 'punching' interactions, the robot collapses as a result of the person hitting it, displaying the ego-motion of falling (e.g., frames in Figure 1 (b)). Similarly, the robot's body shakes as a human is shaking hands with it. Translation movement of the robot is also present even when there are no interactions.

As a result, the video dataset composed of 12 sets are constructed (containing 57 continuous video sequences). Videos in two different sets were taken at a different environment and/or with different human actors. Each set contains multiple continuous videos, which include at least one execution per human activity. In addition, in order to support the training of the robot, we also prepared the segmented version of the dataset: videos in each dataset are segmented so that each video segment contains one activity execution, providing us at least 7 video segments per set.

We emphasize that our first-person videos are different from public activity recognition datasets (e.g., [14, 7, 12]) which are in the third-person viewpoints. It also is different from previous gesture recognition datasets using Kinect sensors [16], since videos in our dataset involves heavy ego-motion (i.e., camera motion) caused by human-observer interactions. It is different from [6, 4, 10] as well, in the aspect that our videos contain movements of interacting persons as well as ego-motion of the observer.

3. Features for first-person videos

In this section, we discuss motion features for first-person videos. We construct and evaluate two categories of video features, global motion descriptors and local motion descriptors, and confirm that each of them contributes to the recognition of different activities from first-person videos. In addition, we present kernel functions to combine global features and local features for the activity recognition. Our kernels reliably integrates both global and local motion information, and we illustrate that these multi-channel kernels benefit first-person activity recognition.

We first introduce video features designed to capture global motion (Subsection 3.1) and local motion (Subsection 3.2) observed during humans' various interactions with the observer. Next, in Subsection 3.3, we cluster features to form visual words and obtain histogram representations. In Subsection 3.4, multi-channel kernels are described. Experimental results evaluating features (and their combinations) are presented in Subsection 3.5.

3.1. Global motion descriptors

For describing global motion in first-person videos, we take advantage of dense optical flows. Optical flows are measured between every two consecutive frames of a video, where each flow is a vector describing the direction and magnitude of the movement of each pixel. We apply a conventional dense optical flow computation algorithm to summarize global motion of the observer.

We designed our global motion descriptor as a histogram of extracted optical flows: We categorize observed optical flows into multiple types based on their locations and directions, and count the number of optical flows belonging to each category. The system places each of the computed optical flows into one of the predefined s-by-s-by-8 histogram bins, where they spatially divide a scene into s by s grids and 8 representative motion directions. Each descriptor is constructed by collecting optical flows in a fixed time duration (e.g., 0.5 seconds). Figure 3 shows an example sequence of global descriptors obtained from one video.

3.2. Local motion descriptors

We use sparse 3-D XYT space-time features as our local motion descriptors. For this purpose, we interpret a video as a 3-D XYT volume, which is formed by concatenating 2-D XY image frames of the video along time axis T. We then pass it to the spatio-temporal feature detector, which searches for a set of small XYT video patches that it believes to contain salient motion (i.e., appearance changes) inside. The intention is to abstract local motion information inside each of the detected video patches, and use it as a descriptor. More specifically, we obtain a local descriptor by summarizing gradient values of the detected video patch.

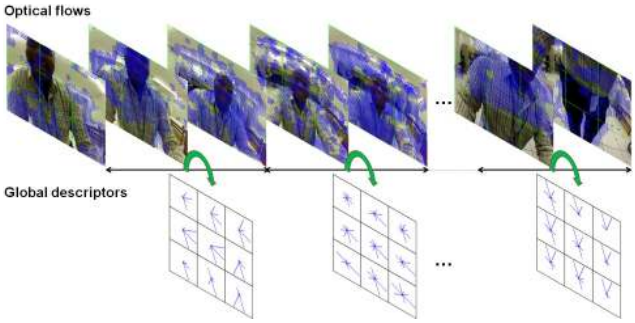


Figure 3. Example global motion descriptors obtained from a video of a human hugging the observer, which concatenates observed optical flows. These three descriptors (obtained during different types of ego-motion of the camera) are distinct, suggesting that our descriptors correctly captures observer ego-motion.

We have chosen a cuboid feature detector [3] as our spatio-temporal feature extractor, applying a dimensionality reduction method (principal component analysis) to compute our local motion descriptors having 100 dimensions. Figure 4 illustrates example cuboids detected.

3.3. Visual words

We take advantage of the concept of *visual words*, in order to represent motion information in videos more efficiently. Motion descriptors are clustered into multiple types (i.e., w words) based on their descriptor values using traditional k-means. As a result, each extracted motion descriptor is interpreted as an occurrence of one of the w visual words (e.g., 800 words).

Once visual words are obtained by clustering motion descriptors, their histogram is computed per video v_i to represent its motion. The histogram H_i essentially is a w-dimensional vector $H_i = [h_{i1}\ h_{i2}\ ...\ h_{iw}]$, where h_{in} is the number of nth visual words observed in the video v_i. Let a_n denote nth visual word, and let d be a motion descriptor. Then,

$$h_{vn} = |\{d \mid d \in a_n\}|\,. \tag{1}$$

Our clustering and histogram construction processes are applied for the global motion descriptors and local motion descriptors separately. Two histograms, one for global motion and the other for local motion, are obtained as a result. The feature histogram H_i for video v_i directly serves as our feature vector representing the video: $x_i = [H_i^1; H_i^2]$, where H_i^1 is the histogram of global descriptors and H_i^2 is the histogram of local descriptors.

3.4. Multi-channel kernels

We present multi-channel kernels that consider both global features and local features for computing video similarities. A kernel $k(x_i, x_j)$ is a function defined to model distance between two vectors x_i and x_j. Learning a clas-

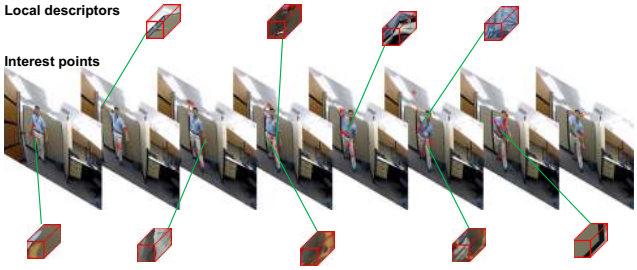


Figure 4. Example local motion descriptors obtained from our video of a person throwing an object to the observer. Locations with salient motion are detected, and their 3-D XYT volume patches are collected as our local descriptors.

sifier (e.g., SVMs) with an appropriate kernel function enables the classifier to estimate better decision boundaries tailored for the target domain. In order to integrate both global and local motion cues for reliable recognition from first-person videos, we defined multi-channel kernels that lead to the computation of a non-linear decision boundary.

We construct two types of kernels: a multi-channel version of histogram intersection kernel, and multi-channel χ^2 kernel which was also used in [19] for object classification. These multi-channel kernels robustly combines information from channels (global motion and local motion in our case).

Our histogram intersection kernel is defined as follows:

$$k(x_i, x_j) = \exp\left(-\sum_c D_c^h(H_i, H_j)\right) \qquad (2)$$

where H_i and H_j are the histograms for channel c of x_i and x_j, and $D_c^h(H_i, H_j)$ is the histogram distance defined as

$$D_c^h(H_i, H_j) = \sum_{n=1}^{w}\left(1 - \frac{\min(h_{in}, h_{jn})}{\max(h_{in}, h_{jn})}\right). \qquad (3)$$

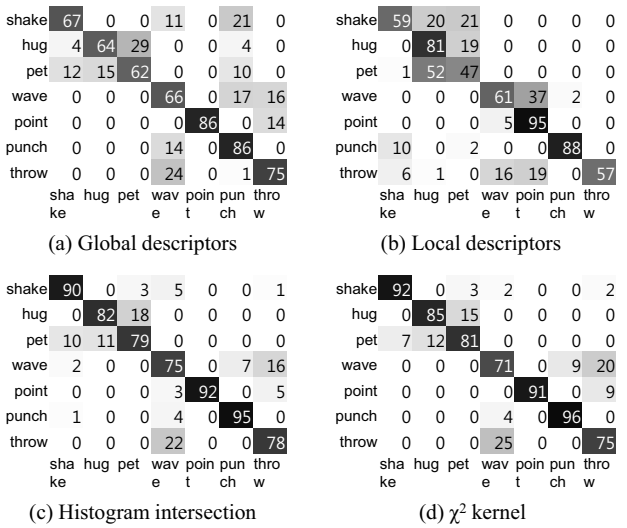
The χ^2 kernel is similar, except that the distance function is newly defined as:

$$D_c^{\chi^2}(H_i, H_j) = \frac{1}{2 \cdot M_c}\sum_{n=1}^{w}\frac{(h_{in} - h_{jn})^2}{h_{in} + h_{jn}} \qquad (4)$$

where M_c is the mean distance between training samples.

3.5. Evaluation

We use a repeated random sub-sampling validation to measure the classification accuracy of our recognition approach. The segmented version of our first-person video dataset was used, where each of its videos contains a single occurrence of one of the seven activities. That is, at each round, we selected a half of our dataset (i.e., 6 sets with 42 videos) as training videos and use the other 6 sets for the testing. The mean classification accuracy was obtained by repeating this random training-testing splits for 100 rounds.



(a) Global descriptors (b) Local descriptors

(c) Histogram intersection (d) χ^2 kernel

Figure 5. Confusion matrices of the baseline activity classification approaches only using one type of motion features and multi-channel classifiers using both features: (a) global motion representation,(b) local motion representation, (c) multi-channel histogram intersection kernel, and (d) multi-channel χ^2 kernel.

In addition, since the clustering algorithm we use in Subsection 3.3 contains randomness, we repeated this step for 10 times and averaged the performances.

Local vs. global motion: First, we evaluate the activity classification ability of our approach while forcing the system to only use one of the two motion features (global vs. local). The objective is to identify which motion representation contributes to recognition of which activity, and confirm that using two types of motion features jointly (using our multi-channel kernel) will benefit the overall recognition.

We implemented two baseline activity classifiers: Both these baseline classifiers are support vector machine (SVM) classifiers, which use a standard Gaussian kernel relying on only one feature channel (either global or local) for the classification. The confusion matrix for these two classifiers are illustrated in Figure 5 (a)(b). Their average classification accuracies were 0.722 (global) and 0.698 (local). The figure illustrates that two feature types capture very different aspects of motion, even though their overall classification accuracies are similar. For example, in the case of 'pointing conversation', the approach with local descriptors showed higher true positive rate while the global descriptors showed better false positive rate. The situation was the opposite for the 'throwing'. This suggest that a kernel to robustly combine both global and local features is needed.

Classification with multi-channel: We evaluated SVM classifiers using our multi-channel kernels. Figure 5 (c)(d) shows the results of our approach with the two types of

multi-channel kernels described in the previous subsection. We are able to observe that our approaches obtain much higher classification accuracy compared to the baseline approaches utilizing only one motion feature (i.e., compared to the confusion matrix Figure 5 (a)(b)): 0.844 and 0.843. This confirms that utilizing both global and local motion benefits overall recognition of human activities from first-person videos, and that our kernel functions are able to combine such information reliably.

4. Recognition with activity structure

In the case of high-level activities, considering activities' structures is crucial for their reliable recognition. More specifically, the system must consider how many components (i.e., sub-events) the activity should be divided into and how they must be organized temporally. This is particularly important for interaction-level activities where cause-and-effect relations are explicitly displayed, such as the observer 'collapsing' as a result of a person 'hitting' it in the punching interaction. The system must learn the structure representation of each activity and take advantage of it for more reliable recognition.

In this section, we present a new recognition methodology that explicitly considers the activity structure, and investigate how important it is to learn/use structures for first-person activity videos. We first describe our structure representation, and define a new kernel function computing video distances given a particular structure. Next, we present an algorithm to search for the best activity structure given training videos. The idea is to enable evaluation of each structure by measuring how similar its kernel function is to the optimal function, and use such evaluation to find the optimal structure.

4.1. Hierarchical structure match kernel

We represent an activity as a continuous concatenation of its sub-events. That is, we define the structure of an activity as a particular division that temporally splits an entire video containing the activity into multiple video segments.

Formally, we represent the activity structure in terms of hierarchical binary divisions with the following production rules:

$$\begin{aligned} S[t_1,t_2] &\rightarrow (S[t_1,t_3], S[t_3,t_2]) \\ S[t_1,t_2] &\rightarrow (t_1,t_2) \end{aligned} \tag{5}$$

where t_3 is a relative time point ($0 \leq t_1 < t_3 < t_2 \leq 1$) describing how the structure is splitting the video duration $[t_1,t_2]$. Any activity structure constructed by applying a number of production rules starting from $S[0,1]$ (until they reach terminals) is considered as a valid structure (e.g., $S = ((0,0.5),(0.5,1))$). Each relative time interval (t_1,t_2) generated as a result of the second rule is a terminal, specifying that the structure representation considers it as an atomic-level sub-event. The above production rules can be viewed as those of an attribute grammar.

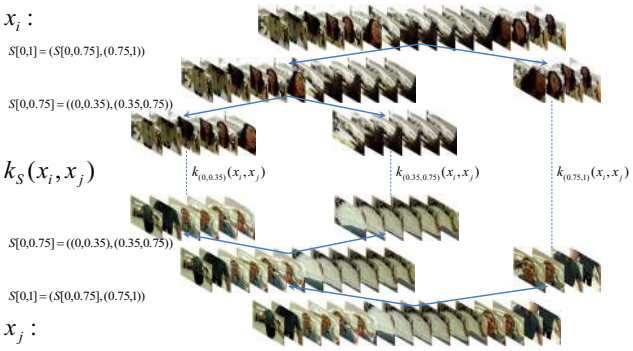


Figure 6. An example matching between two hugging videos, x_i and x_j, using the kernel K_S constructed from the hierarchical structure $S = (((0,0.35),(0.35,0.75)),(0.75,1))$.

The idea behind our structure representation is to take advantage of it to better measure the distance between two videos by performing hierarchical segment-to-segment matching (Figure 6). That is, if two videos contains an identical activity and if they are divided into video segments based on the correct activity structure, the similarity between each pair of video segments must be high.

Given a particular activity structure S, we define the kernel function $k_S(x_i,x_j)$ measuring the distance between two feature vectors x_i and x_j with the following two equations:

$$\begin{aligned} k_{(S[t_1,t_3],S[t_3,t_2])}(x_i,x_j) &= k_{S[t_1,t_3]}(x_i,x_j) + k_{S[t_3,t_2]}(x_i,x_j), \\ k_{(t_1,t_2)}(x_i,x_j) &= \sum_{n=1}^{w} \frac{(h_{in}^{(t_1,t_2)} - h_{jn}^{(t_1,t_2)})^2}{h_{in}^{(t_1,t_2)} + h_{jn}^{(t_1,t_2)}} \end{aligned} \tag{6}$$

where $h_{in}^{(t_1,t_2)}$ is the number of nth visual word detected inside the time interval (t_1,t_2) of the video x_i. Notice that this structure kernel is constructed for each channel c, resulting a multi-channel kernel integrating (i.e., summing) all $k_{S_c}^c$ (i.e., $k_{S_1}^1$ and $k_{S_1}^2$). Instead of ignoring temporal locations of detected descriptors using bag-of-words (e.g., kernels discussed in the previous section), our new kernel considers the structural formulation of descriptors. We call this *hierarchical structure match kernel*.

Our structure match kernel can be efficiently implemented with temporal integral histograms [11], which allows us to obtain a feature histogram of any particular time interval in $O(w)$. Our kernel takes $O(w \cdot r)$ per each (x_i,x_j), where r is the number of segments generated as a result of the structure. In most cases $r < 10$. In principle, our structure kernel is able to cope with any types of classifiers by serving as a distance measure.

4.2. Structure learning

In this subsection, we present our approach to learn the activity structure and its kernel that best matches training

videos. We first introduce *kernel target alignment* [15] that measures the angle between two Gram matrices, and present that it can be used to evaluate structure kernels for our activity recognition. The idea is to represent the 'optimal kernel function' and candidate structure kernels in terms of Gram matrices and measure their similarities. Next, we present our strategy to obtain the optimal structure by hierarchically evaluating multiple candidate structures using the kernel target alignment.

Kernel target alignment: Given a set of training samples $\{x_1, ..., x_m\}$, let K_1 and K_2 be the Gram matrices of kernel functions k_1 and k_2:

$$K = (k(x_i, x_j))_{i,j=1}^{m}. \tag{7}$$

Then, the *alignment* between two kernels can be computed as:

$$A(K_1, K_2) = \frac{\langle K_1, K_2 \rangle_F}{\sqrt{\langle K_1, K_1 \rangle_F \langle K_2, K_2 \rangle_F}} \tag{8}$$

where $\langle K_1, K_2 \rangle_F$ is the Frobenius inner product between the kernel matrix K_1 and K_2. That is, $\langle K_1, K_2 \rangle_F = \sum_{i,j=1}^{m} k_1(x_i, x_j) k_2(x_i, x_j)$. The alignment function A measures the cosine value of the angle between two Gram matrices, evaluating how similar they are.

We take advantage of the kernel target alignment for evaluating candidate activity structures. For this purpose, we define the Gram matrix L corresponding to the optimal distance function:

$$L = (l(i,j))_{i,j=1}^{m}, \quad l(i,j) = \begin{cases} 0 & y_i = y_j \\ 1 & \text{otherwise,} \end{cases} \tag{9}$$

where y_i is the activity class label corresponding to the training sample x_i. The matrix L essentially indicates that the distance between any two training samples must be 0 if they have an identical activity class, and 1 otherwise.

The idea is to compute the alignment $A(K_S, L)$ and evaluate each candidate kernel K_S. That is, our alignment $A(K_S, L)$ measures how similar the kernel function K_S corresponding to a particular structure S is to the optimal distance function L for the training data. This provides the system an ability to score possible activity structure candidates so that it can search for the best structure S^*. We denote $A(K_S, L)$ simply as $A(K_S)$. Computation of $A(K_S)$ takes $O(m^2 \cdot w \cdot r)$.

Hierarchical structure learning: Here, we present our strategy to search for the optimum structure based on training videos. The goal is to find the structure S^* that maximizes the kernel alignment for the training data: $S^* = \arg\max_S A(K_S)$. More specifically, we describe our learning process as:

$$S[t_1, t_2]^* = \arg\max_{S[t_1,t_2]} \left\{ \max_{t'} A(K_{(S[t_1,t']^*, S[t',t_2]^*)}), A(K_{(t_1,t_2)}) \right\} \tag{10}$$

where $t_1 < t' < t_2$. With the above formulation, the structure learning is interpreted as the searching of $S[0,1]^*$, the best structure dividing the entire activity duration $[0,1]$, among an exponential number of possible structures.

For the computational efficiency, we take advantage of the following greedy assumption:

$$\arg\max_{t'} A(K_{(S[t_1,t'], S[t',t_2])}) \approx \arg\max_{t'} A(K_{((t_1,t'),(t',t_2))}). \tag{11}$$

As a result, the following recursive equation T, when computed for $T(0,1)$, provides us the optimal structure S^*:

$$T(t_1, t_2) = \begin{cases} (t_1, t_2) & \text{if } t_3 = 0 \text{ or } 1 \\ (T(t_1, t_3), T(t_3, t_2)) & \text{otherwise,} \end{cases} \tag{12}$$

where $t_3 = \arg\max_{t'} A(K_{((t_1,t'),(t',t_2))})$. This structure can either be learned per activity, or the system may learn the common structure suitable for all activity classes.

As a result, the time complexity for computing the final structure S^* is $O(m^2 \cdot w \cdot p \cdot q)$ where p is the number of layers and q is the number of t' the system is checking at each layer. In most cases, p is smaller than 4, and this computation is only required once at the training stage.

4.3. Evaluation - classification

We evaluated the classification performance of our approach using the same setting described in Section 3.5. For each validation round, our approach learns the optimal structure from training videos for the classification. One common structure that best distinguishes videos with different activities was obtained, and our kernel function corresponding to the learned structure was constructed. SVM classifiers were used as the base classifiers of our approach.

In addition, in order to illustrate the advantage of our structure learning and recognition for first-person videos, we tested two state-of-the-art activity recognition approaches: spatio-temporal pyramid matching [2], and dynamic bag-of-words (BoW) [11]. Spatio-temporal pyramid match kernel is a spatio-temporal version of a spatial pyramid match kernel [8]. Similar to our approach, it divides an entire video into multiple spatio-temporal segments, and hierarchically combines their match. The main difference is that our hierarchical structure match kernel, at each layer, learns the optimal temporal division that best fits the training data. Multiple possible structures are considered to learn the optimal structure in our approach, instead of having one fixed pyramid. A discriminative version of dynamic BoW was also tested. This approach is similar to our kernel and [2] in the aspect that it temporally divides each video into multiple parts to perform matching. However, in dynamic BoW, an activity model was learned only using videos belonging to that class without considering other activity videos, which results inferior performance.

Table 1. Classification performances of recognition approaches measured with our first-person video dataset. Our structure match approach performed superior to the two bag-of-words classifiers from Section 3.4 and the two state-of-the-art methods [2, 11].

Recognition method	Local feature only	Both features
χ^2 kernel	82.4 %	84.3 %
Histogram Intersect.	82.4 %	84.4 %
ST-Pyramid match [2]	82.6 %	86.0 %
Dynamic BoW [11]	82.8 %	87.1 %
Structure match	**83.1 %**	**89.6 %**

Table 1 shows the classification accuracies of the approaches measured with our first-person video dataset. We illustrate performances of the classifiers that only use a single feature type (local features) as well as those of the classifiers with multi-channel kernels. We are able to observe that the approaches with our structure match kernel perform superior to the other state-of-the-art approaches. This confirms that learning the optimal structure suitable for activity videos benefits their recognition particularly in the first-person activity recognition setting.

4.4. Evaluation - detection

In this subsection, we evaluate the activity detection ability of our approach using the first-person dataset. Activity detection is the process of finding correct starting time and ending time of the activity from continuous videos. Given an unknown video sequence (i.e., continuous observations from a camera), for each activity, the system must decide whether the activity is contained in the video and when it is occurring. Activity detection is the ability that we want the system to possess, in order for it to function in real-world environments.

Implementation: We implemented a binary classifier per activity, which is trained to classify all possible time intervals of the input video sequence using the sliding window technique. Multiple activity durations learned from positive training examples of each activity were considered, and we trained the classifier by sampling video segments (with the same length) from continuous training videos. When learning the structure (i.e., Subsection 4.2), we used an identical number of positive examples and negative examples to construct Gram matrices. The structure is learned per activity class.

In addition to the recognition approach with our structure matching kernel, we implemented three baseline approaches for comparison: SVM classifiers only using local features, those only using global features, and the method with our multi-channel kernel discussed in Section 3.4. All three baselines use χ^2-based kernels, which showed superior detection performance compared to histogram intersection-based kernels in the detection task. Results with these baseline approaches represent the performances of conventional bag-of-words approaches using space-time features and/or optical flow features.

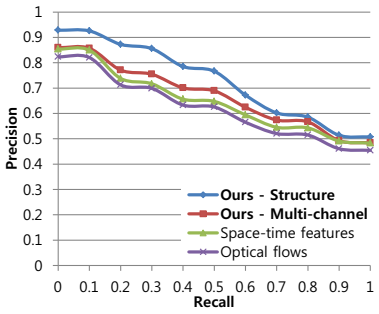


Figure 7. Average precision-recall curves of our first-person activity detectors. A higher graph suggests better performance.

Settings: Similar to the classification experiment, we performed validations by randomly splitting the dataset (i.e., 12 sets) into 6 training sets and 6 testing sets. This training-testing set selection process was repeated 100 rounds, and we averaged their performance.

Results: We measured the detection performance of each approach in terms of *precision* and *recall* values. More specifically, we measured average precision-recall (PR) curves with our dataset. Precision, $tp/(tp+fp)$, and recall, $tp/(tp+fn)$, change as the system changes the detection threshold, and PR curve is obtained by recording (precision, recall) pairs observed. In our SVM classifiers, we used their probability estimate values [18] to make the detection decision and draw PR curves.

Figure 7 shows average PR-curves combining results for all seven activity classes. We are able to confirm that our method using structure match kernel performs superior to the conventional SVMs with the bag-of-words paradigm. The average precision (AP) values for our approach was 0.709, while AP values for baselines were 0.601 (global features), 0.627 (local features), and 0.651 (multi-channel). Figure 8 shows example detection results.

We also present PR curves for each activity category in Figure 9. Our structure match obtained the highest mean APs in all activity categories, and particularly performed superior to baseline approaches for 'punching', 'point-converse', and 'petting'. The structure match kernel not only considers both global motion and local motion of first-person videos (with a optimum weighting computed using kernel target alignment), but also reflects sequential structure of the activity, thereby correctly distinguishing interactions from false positives. The result suggests that fusing global/local motion information and considering their temporal structure are particularly necessary for detecting high-level human interactions with complex motion.

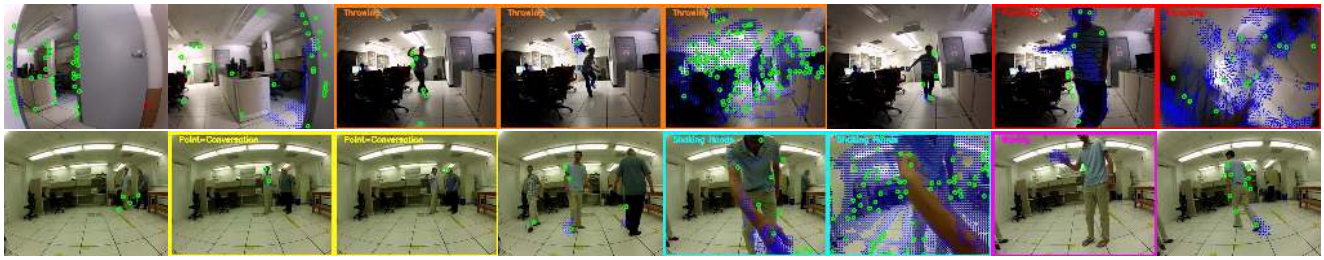
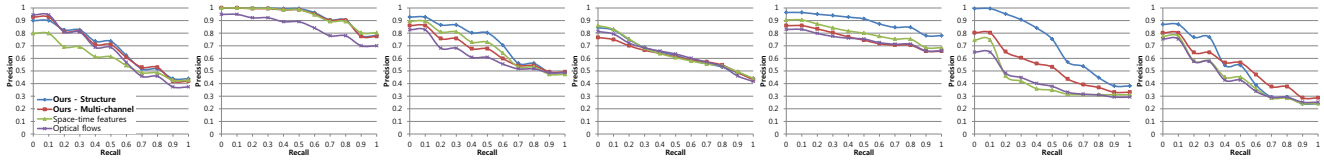


Figure 8. Example activity detection results from continuous videos. Throwing (orange box) and punching (red box) are detected in the upper video, and pointing (yellow box), hand shaking (cyan box), and waving (magenta box) are detected in the lower video. Green circles show local spatio-temporal features and blue arrows show optical flows.



(a) Hand shake (b) Hug (c) Pet (d) Wave (e) Point-Converse (f) Punch (g) Throw

Figure 9. Average precision-recall curves for each activity category are presented. Approaches with our kernels (blue curve and red curve) performed better than the baselines using space-times features (green) and optical flows (purple) overall. Particularly, activity detection using our structure match kernel showed superior performance compared to all the others.

5. Conclusion

In this paper, we introduced the problem of recognizing interaction-level activities from videos in first-person perspective. We extracted global and local features from first-person videos, and confirmed that multi-channel kernels combining their information are needed. Furthermore, we developed a new kernel-based activity learning/recognition methodology to consider the activities' hierarchical structures, and verified that learning activity structures from training videos benefits recognition of human interactions targeted to the observer. As a result, friendly human activities such as 'shaking hands with the observer' as well as hostile interactions like 'throwing objects to the observer' were correctly detected from continuous video streams. Our approach is designed to process various types of human activities, and we illustrated its potential using 7 classes of commonly observed interactions. One future work is to extend our approach for early recognition of humans' intention based on activity detection results and other subtle information from human body movements. Our paper presented the idea that first-person recognition of physical/social interactions becomes possible by analyzing video motion patterns observed during the activities.

Acknowledgment: The research described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. This research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-10-2-0016.

References

[1] J. K. Aggarwal and M. S. Ryoo. Human activity analysis: A review. *ACM Computing Surveys*, 43:16:1–16:43, April 2011.

[2] J. Choi, W. Jeon, and S. Lee. Spatio-temporal pyramid matching for sports videos. In *ACM MIR*, 2008.

[3] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *IEEE Workshop on VS-PETS*, 2005.

[4] A. Fathi, A. Farhadi, and J. M. Rehg. Understanding egocentric activities. In *ICCV*, 2011.

[5] A. Fathi, J. Hodgins, and J. Rehg. Social interactions: A first-person perspective. In *CVPR*, 2012.

[6] K. M. Kitani, T. Okabe, Y. Sato, and A. Sugimoto. Fast unsupervised ego-action learning for first-person sports videos. In *CVPR*, 2011.

[7] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*, 2008.

[8] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006.

[9] J. Niebles, C. Chen, and L. Fei-Fei. Modeling temporal structure of decomposable motion segments for activity classification. In *ECCV*, 2010.

[10] H. Pirsiavash and D. Ramanan. Detecting activities of daily living in first-person camera views. In *CVPR*, 2012.

[11] M. S. Ryoo. Human activity prediction: Early recognition of ongoing activities from streaming videos. In *ICCV*, 2011.

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

[13] M. S. Ryoo and J. K. Aggarwal. Stochastic representation and recognition of high-level group activities. *IJCV*, 93(2):183–200, 2011.

[14] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: a local SVM approach. In *ICPR*, 2004.

[15] N. Shawe-Taylor and A. Kandola. On kernel target alignment. In *NIPS*, 2002.

[16] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from a single depth image. In *CVPR*, 2011.

[17] Z. Si, M. Pei, B. Yao, and S. Zhu. Unsupervised learning of event and-or grammar and semantics from video. In *ICCV*, 2011.

[18] T. Wu, C. Lin, and R. Weng. Probability estimates for multi-class classification by pairwise coupling. *JMLR*, 5:975–1005, 2004.

[19] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *IJCV*, 73:213–238, April 2007.