

Discovering Discriminative Action Parts from Mid-Level Video Representations

Michalis Raptis
UCLA

mraptis@cs.ucla.edu

Iasonas Kokkinos
Ecole Centrale de Paris and INRIA-Saclay

iasonas.kokkinos@ecp.fr

Stefano Soatto
UCLA

soatto@ucla.edu

Abstract

We describe a mid-level approach for action recognition. From an input video, we extract salient spatio-temporal structures by forming clusters of trajectories that serve as candidates for the parts of an action. The assembly of these clusters into an action class is governed by a graphical model that incorporates appearance and motion constraints for the individual parts and pairwise constraints for the spatio-temporal dependencies among them. During training, we estimate the model parameters discriminatively. During classification, we efficiently match the model to a video using discrete optimization. We validate the model's classification ability in standard benchmark datasets and illustrate its potential to support a fine-grained analysis that not only gives a label to a video, but also identifies and localizes its constituent parts.

1. Introduction

We develop a mid-level representation of video data that can be inferred from video sequences and can serve to classify human actions, events, or activities. Among various classification tasks, we consider recognition (“*What action?*”) and localization (“*Where in the video?*”). The human perceptual system can identify and localize an action even if a significant portion of the actor is occluded. This requires a combination of *local* appearance, relative motion, and contextual information. We argue that capturing the statistics of *spatio-temporally localized* regions can improve the performance of action recognition methods.

The main contribution of our work is the development of a spatio-temporal latent variable model for actions that is able to discriminatively identify *salient structures* and *exploit their pairwise relationships* for the tasks outlined above. In particular, our model combines ideas from part-based models [33, 32, 9] with the extraction of a sparse, low-level video representation.

In particular, a generic low-level representation is used to describe the spatio-temporal content of a collection of moving points. Such points are grouped based on spatial

and dynamic similarity, and constitute putative “parts” in a complex spatio-temporal event, action or activity portrayed in a video snippet. Each part is associated with a descriptor, capturing statistics of intensity, motion and appearance; we attain time efficiency by employing a regular grid during descriptor construction, which allows us to accommodate dense trajectories.

Our model takes as input this cluster-based video representation and determines which clusters should be used to ‘instantiate’ the parts of an action class; the association of “groups” with “parts” is treated by employing latent variables. During the training phase, the classifier parameters are learned simultaneously with the group association using a weakly annotated training set. In particular, the quality of a given instantiation is phrased in terms of the energy of a Markov random field (MRF) [32]. During both training and testing, optimizing the MRF energy amounts to solving an assignment problem; we efficiently solve this using discrete optimization [15]. The model can thus be used both to classify an entire video snippet and to highlight local associations and determine which parts are relevant to a given classification task for the purpose of analysis. The learning of the cost function that drives the matching is performed discriminatively using large-margin learning of a ranking function, while the individual part properties and the relations among them are estimated in a bootstrapped, concave-convex procedure [35]. Some additional technical contributions we make at the modeling level include the introduction of pairwise features for trajectories, the treatment of scale, and the large-margin ranking objective training procedure [12, 2].

We evaluate our approach on two benchmark action datasets. Our results indicate that our model performs competitively in the overall classification task, while it exhibits the additional benefit of enabling localization analysis. Thus, our method can support more fine-grained decision tasks than reflected in the available benchmark datasets, including for instance the discovery of spatio-temporal relations between parts and the “parsing” of a video into action parts.

After briefly reviewing previous work below, in Sect. 2

we describe our low-level representation, and then we proceed to describe our part-based representation in Sect. 3. The performance of our method on action recognition and localization is evaluated in Sect. 4.

1.1. Related Work

Existing approaches to action and activity recognition can be coarsely lumped into three classes. The first uses bag-of-words representations, where the “words” are computed either statically for each frame [25, 14] or from trajectories [27, 19, 22, 26]. The second uses global spatio-temporal templates, such as motion history [3], spatio-temporal shapes [1], and other templates [13], that retain the spatial structure. This class suffers from sensitivity to nuisance factors such as vantage point, scale, or partial occlusions. The third class of approaches attempts to decompose an action or activity into “parts” designed to capture aspects of the local spatial or temporal structure in the data. Sequential data models have been employed to represent the temporal variability [11, 21]. For instance, Brendel and Todorovic [4] use a time series of activity codewords, identifying at each frame only one promising region as a part of an activity and modeling the temporal consistency through a Markov chain. More complex part-based models have been proposed [32] where pairwise relationships among predefined image patches are encoded explicitly. However, the performance of this model heavily relies on the independent detector of salient image patches. Further, Niebles *et al.* [20] extend the notion of a part from a spatial segment [4, 32] to a set of consecutive video frames. This enables temporal composition, but the ensuing model lacks the ability to spatially localize action parts, because each video segment is represented as a collection of spatio-temporal interest points [25]. Our approach falls in the latter class. However, we take the prior methods [20, 9] a step further and encode the spatial *and* temporal structure of the action, enabling part localization both in space and time.

Laptev *et al.* [17] have shown that encoding the spatio-temporal layout of a video using a fixed space-time grid improves the recognition performance compared to bag-of-words approaches [25]. To maximize the recognition accuracy, Sun *et al.* [27] adapted Multi-Kernel methods to learn the optimal weights between the several different feature channels obtained from a fixed space-time grid. In contrast, our algorithm adaptively identifies “relevant” video segments and selects a portion of them as parts of an action according to their local appearance, motion statistics and spatio-temporal structure.

Recently, Brendel and Todorovic [5] and Lan *et al.* [16] proposed activity models that also enable them to learn the relevant action parts of the video. [5] proposed a generative method that encodes an activity as a weighted directed graph defined on a “blocky” hierarchical over-segmentation

of the video. However, this model lacks the ability to discriminatively disambiguate between repeated structures in videos and the actual parts of the activity. Thus it can fail to distinguish actions captured with similar time-varying background. The discriminative training of our model enables us to have robust classification prediction even in those cases. On the other hand, Lan *et al.* [16] introduced a discriminative model that couples activity recognition with person detection and tracking. However, the model assumptions restrict its applicability to scenarios where the actor’s figure is fully visible for the entire duration of the video. Our approach aims to overcome those restricting assumptions by focusing on local spatio-temporal regions.

2. Low-Level Representation

We outline below our video processing front-end, which largely follows prior tracklets works [22, 31]. We detail certain technical aspects which resulted in improved classification and time efficiency, in particular the clustering of trajectories (2.1) and the use of a regular grid for efficient descriptor construction (2.2).

2.1. Trajectory Groupings

Our goal is to determine regions of the video that are relevant to a specific action. To achieve this goal we segment the video volume into regions that are biased to belong to the same moving object or person. This spatio-temporal segmentation is based on grouping of dense trajectories [6]. Trajectories with low spatial variation are pruned, since they are considered uninformative regions of the video sequence. We employ a ‘distance’ to measure the similarity of trajectories that co-exist in a time interval, are spatial neighbors and have similar motion. Given two trajectories $\{\mathbf{x}_a[t]\}_{t=\tau_a}^{T_a}$ and $\{\mathbf{x}_b[t]\}_{t=\tau_b}^{T_b}$ that co-exist in $[\tau_1, \tau_2]$, we have:

$$d(a, b) = \max_{t \in [\tau_1, \tau_2]} d_{\text{spatial}}[t] \cdot \frac{1}{\tau_2 - \tau_1} \sum_{t=\tau_1}^{\tau_2} d_{\text{velocity}}[t] \quad (1)$$

where $d_{\text{spatial}}[t] = \|\mathbf{x}_a[t] - \mathbf{x}_b[t]\|_2$ is the ℓ_2 distance of the trajectory points at corresponding time instances and $d_{\text{velocity}}[t] = \|\dot{\mathbf{x}}_a[t] - \dot{\mathbf{x}}_b[t]\|_2$ is the distance of the velocity estimates, obtained by differentiation: $\dot{\mathbf{x}}[t] = \mathbf{x}[t] - \mathbf{x}[t-1]$. The ‘distance’ penalizes trajectories that are (spatially) far apart even in a small subset of their temporal overlap. This is slightly different from previous work [6] in that our ‘distance’ enforces spatial compactness for trajectories.

To group trajectories we compute an affinity $w(a, b) = \exp(-d(a, b))$ between each trajectory pair (a, b) and form an $n \times n$ affinity matrix for a video containing n trajectories. To ensure the spatial compactness of the estimated groups, we enforce the above affinity to be zero for trajectory pairs that are not spatially close ($\max_{t \in [\tau_1, \tau_2]} d_{\text{spatial}} \geq 30$). We



Figure 1. Examples of trajectory groups; each group has a distinct color.

then cluster trajectories using an efficient greedy agglomerative hierarchical clustering procedure [28] that returns a membership indicator function $m(\cdot)$. To determine the appropriate number of clusters in a video sequence, we used Cattell’s scree test [7]. Specifically, we set the number of clusters to $N = \operatorname{argmin}_i \lambda_{i+1}^2 / (\sum_{j=1}^i \lambda_j^2) + c \cdot i$, where λ_i are the eigenvalues of the affinity matrix and $c = 10^{-4}$. This produces bundles of trajectories, illustrated in Fig. 1.

2.2. Trajectory descriptors

We construct a simple and computationally efficient low-level description designed to be insensitive to partial occlusion and coarse variability in illumination and pose. Histogram of gradient (HoG), histogram of optical flow (HoF) [17] and histogram of the oriented edges of the motion boundaries (HoMB) [31] descriptors are extracted on a *regular grid* at three different scales. By virtue of using a regular grid, the descriptors can be computed in linear time [10] while also covering the largest part of the video signal’s spatio-temporal domain.

We use a dictionary for each low-level descriptor (HoG, HoF, HoMB) independently, using K-means, and quantize all descriptors by assigning them to their closest dictionary element based on ℓ_2 distance. The use of a regular grid allows us also to accelerate the estimation of descriptors around trajectories when compared to other methods [22, 31]. In particular, each trajectory within a group is spatially ‘quantized’ to the grid, and the codebook labels for each of the three descriptors are accumulated into a histogram. This process is repeated for all group members and results in three histograms for the HoG, HoF and HoMB features respectively. Their concatenation yields our group descriptor h_k .

To capture the coarser spatio-temporal shape characteristics of the ensemble of trajectories \mathbf{x}_i in the group k defined by their absolute positions: $D_k = \{\{\mathbf{x}_i[t]\}_{t=\tau_i}^{T_i}, \forall i :$

$m(i) = k\}$, we compute the *mean group trajectory*:

$$g_k[t] = \frac{1}{|\{i\}|} \sum_{\substack{\forall i: t \in [\tau_i, T_i], \\ m(i)=k}} \mathbf{x}_i[t], \quad t \in \bigcup_{\{i:m(i)=k\}} [\tau_i, T_i] \quad (2)$$

The mean group trajectories are used to estimate the pairwise relationships between groups within a video sequence, as described in Sect. 3.1. We describe each group G_k as a pair $G_k = \{h_k, g_k\}$.

Moreover, at the coarsest level we form a simple bag-of-words (BoW) representation, h_o , of all groups in terms of the concatenation of the three histograms of all descriptors within the video. Consequently, a video S can be described as the collection of the groups in combination with the histogram h_o : $S = \{h_o, \{G_k\}_{k=1}^N\}$.

3. Mid-level part model

The low-level descriptor S described above constitutes the front-end of our mid-level action model. Our model is learned discriminatively by treating part-cluster assignments as latent variables and quantifying the quality of a presumed action configuration in terms of an MRF score, as detailed in Sect. 3.1. We detail how inference is used for classification in Sect. 3.2, describe model training in Sect. 3.3, and experimentally validate our model on activity recognition and localization in Sect. 4.

3.1. Modeling group interactions

To capture the spatial context of an action, event, or activity, we leverage the relation among mid-level parts using a graphical model. We use a fully connected graph $\mathcal{G} = (V, E)$, with each node $i \in V$ encoding a *part* and each edge $(i, j) \in E$ encoding pairwise *relations* between parts. An additional isolated node F represents the video as a whole.

Given a video \mathbf{x} that has been decomposed into N clusters, we consider a vector of discrete *latent variables* $P = [p_1, \dots, p_{|V|}]$, with $p_i \in \{1, \dots, N\}$ associating each node i with one of the N trajectory clusters. We employ one more latent variable, σ , shared among all spatio-temporal relations; this allows us to bring the relative locations of the action parts to a canonical scale and thereby cope with scale variability.

The latent variable vector thus identifies the locations and appearances of the action parts in the video. Conditioned on the latent variable vector, we can score a video using our model’s unary and pairwise terms, which capture appearance and spatio-temporal information, respectively.

The i -th **unary term** u_i scores the group descriptors h_{p_i} computed for cluster p_i using a linear kernel as $u_i = \langle w_i, h_{p_i} \rangle$. The parameters w_i are estimated discriminatively and allow each part to be tuned to different mid-level action

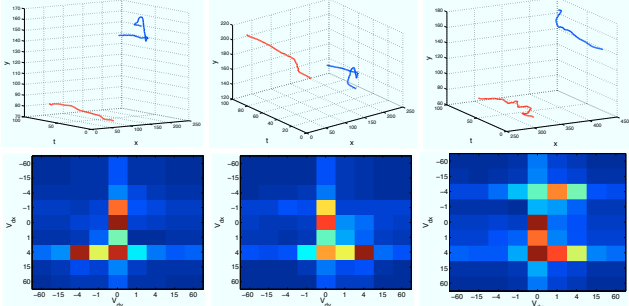


Figure 2. The first row shows pairs of mean group trajectories, while the second shows their corresponding pairwise descriptors.

properties. The isolated node F has a score $u_F = \langle w_0, h_0 \rangle$ where h_0 is the bag-of-words representation.

The **pairwise term** $u_{i,j}$ captures the spatio-temporal relation of the action parts i, j . If nodes i and j have been matched to groups G_{p_i}, G_{p_j} with mean group trajectories $g_{p_i} \in \mathbb{R}^{2 \times \bar{T}_{p_i}}, g_{p_j} \in \mathbb{R}^{2 \times \bar{T}_{p_j}}$, we first estimate if the mean group trajectories co-exist for a sufficiently long time interval. If this condition is not satisfied, their pairwise feature is set to $\psi(g_{p_i}, g_{p_j}, \sigma) = \mathbf{0}$. Otherwise, a feature describing the evolution of their relative positions is computed. First, the canonical relative position is estimated $d_{p_i, p_j, \sigma}[\bar{t}] = (g_{p_i}[\bar{t}] - g_{p_j}[\bar{t}]) / \sigma$, where σ is the latent scale variable, and \bar{t} takes values only in the coexisting time interval. Subsequently, a feature capturing the rate of the convergence/divergence of the trajectory pairs is calculated $v^{x,y}[\bar{t}] = |d_{p_i, p_j, \sigma}^{x,y}[\bar{t} + 1] - d_{p_i, p_j, \sigma}^{x,y}[\bar{t}]|$; we note that this measure is symmetric in i and j . We obtain a statistical description of this signal by soft-quantizing the individual coordinates x, y of $v^{x,y}[\bar{t}]$ using a $2\eta + 3$ -dimensional vector¹:

$$\begin{aligned} \bar{\phi}(x) &= [v_{-1}(x), \rho_{-\eta}(x), \dots, \rho_0(x), \dots, \rho_\eta(x), v_1(x)], \\ v_{\pm 1}(x) &= \left(1 + e^{\frac{\pm x + \mu_0}{s_0}}\right)^{-1}, \quad \rho_\eta(x) = e^{-\frac{(x - \mu_\eta)^2}{s_\eta^2}}. \end{aligned} \quad (3)$$

This ‘soft binning’ vector is extracted for each time instance \bar{t} where the trajectories coexist; this gives us two matrices $\bar{v}_x, \bar{v}_y \in \mathbb{R}^{(2\eta+3) \times \bar{T}_{p_i, p_j} - 1}$ for each coordinate, where \bar{T}_{p_i, p_j} is the length of the coexistence interval. Finally, we set our pairwise feature $\psi(g_{p_i}, g_{p_j}, \sigma)$ equal to the vectorized result of $\bar{v}_x \bar{v}_y^T$. This feature vector can indicate, for instance, whether the two groups are converging in the x coordinate and diverging in the y coordinate, as illustrated in Fig. 2. The pairwise potential is obtained as the inner product with a weight vector $w_{i,j}$, $u_{i,j} = \langle w_{i,j}, \psi(g_{p_i}, g_{p_j}, \sigma) \rangle$.

¹The parameters in this feature vector are fixed for all actions. The μ, s parameters are such that the Gaussian functions ρ_η tessellate the velocity axes geometrically, $\mu_\eta = r^\eta \mu_0, s_\eta = \alpha \mu_0$, while s_0 and μ_0 are set so that the sigmoidal functions v cover the extremes of the domain.

In sum, once the latent variables $z = \{\sigma, P\}$ are given, we can quantify the fit of a video to our action model in terms of a cost obtained by adding the corresponding unary and pairwise terms:

$$\begin{aligned} \text{score}(z) &= \langle w_0, h_0 \rangle + \sum_{i=1}^{|V|} \langle w_i, h_{p_i} \rangle + \\ &\sum_{i=1}^{|V|} \sum_{j=i+1}^{|V|} \langle w_{i,j}, \psi(g_{p_i}, g_{p_j}, \sigma) \rangle. \end{aligned} \quad (4)$$

Having formulated our model, we now turn to the two main problems of (i) estimating the optimal z , given a video and the model parameters and (ii) estimating the model parameters from training data.

3.2. Classification by Subgraph Matching

Classifying a video based on the score described in Eq. 4 entails maximizing it over $z = (P, \sigma)$; i.e., our discriminant function is $s = \text{argmax}_z \text{score}(z)$. As the number of clusters is larger than the number of parts, finding the best cluster-part assignment amounts to solving subgraph matching, a combinatorial optimization problem.

If the cost function contained only unary terms, estimating P for known σ would amount to solving a linear assignment problem, which can easily be solved using Linear Programming. As our cost function includes pairwise terms, we face the NP-hard Quadratic Integer Programming problem. Approximate solutions for this problem in vision include LP/SDP relaxations [24], spectral methods [18, 8], and MRF inference [29]. We follow this last thread and use the TRW-S method [15] which was shown to perform marginally worse than the state-of-the-art method [29] in substantially less time - for our problem it takes a fraction of a second.

To formulate subgraph matching as an MRF labeling problem for each node i , we consider the unary cost $u_i(p_i)$ incurred if we assign to it a label $p_i \in \{1, \dots, N\}$ and the pairwise cost $u_{i,j}(p_i, p_j)$ paid for each of its neighbors j and their possible nodes j . Using the following expressions for the unary and pairwise terms:

$$\begin{aligned} u_i(p_i) &= \langle w_i, h_{p_i} \rangle, \\ u_{i,j}^\sigma(p_i, p_j) &= \begin{cases} \langle w_{i,j}, \psi(g_{p_i}, g_{p_j}, \sigma) \rangle, & p_i \neq p_j \wedge \psi \neq \mathbf{0} \\ -\infty, & p_i = p_j \vee \psi = \mathbf{0} \end{cases}, \end{aligned}$$

we obtain from Eq. (4) $\text{score}(z) = \sum_{i \in V} u_i(p_i) + \sum_{(i,j) \in E} u_{i,j}^\sigma(p_i, p_j)$, a standard MRF energy, apart from the scale variable. To optimize over $z = (\sigma, P)$, we consider a discrete set of scale values, $\sigma \in \{\sigma_1, \dots, \sigma_{N'}\}$; for each σ_k , we estimate $P_k^* = \text{argmax}_P \text{score}(P, \sigma_k)$ using TRW-S. We finally choose the scale index $k^* = \text{argmax}_k \text{score}(P_k^*, \sigma_k)$ that yields the smallest energy. The

output of this process is the latent variable vector $z^* = (P_{k^*}^*, \sigma_{k^*})$ that best fits the action model to a video.

Note that based on our definition of pairwise term $w_{i,j}^\sigma(p_i, p_j)$, groups of trajectories that do not co-exist cannot be simultaneously assigned as parts of the model. This restriction is motivated by our observation that actions of interest on most public available datasets tend to span only one scene of the video.

3.3. Learning

We now address the issue of learning the parameters of our score function. Our score for a video \mathbf{x}_i is the inner product $\langle \mathbf{w}, \phi(\mathbf{x}_i, z_i) \rangle$ of the feature $\phi(\mathbf{x}_i, z_i)$ and the parameter vector \mathbf{w} , where

$$\begin{aligned} \phi(\mathbf{x}_i, z_i) &= [h_0, h_{p_1}, \dots, h_{p_{|V|}}, \psi(g_{p_1}, g_{p_2}, \sigma), \dots, \\ &\quad \psi(g_{p_{|V|}}, g_{p_{|V|-1}}, \sigma)], \\ \mathbf{w} &= [w_0, w_1, \dots, w_{|V|}, w_{1,2}, \dots, w_{|V|,|V|-1}]. \end{aligned}$$

In the previous section, we described how to optimize the score $(z; \mathbf{w})$ over z for a known \mathbf{w} . Our task now is to find the \mathbf{w} that leads to the maximum margin classification. This is equivalent to minimizing the sum of a convex and a concave function, whose optimal solution can be approximated using an alternating optimization algorithm such as CCCP [35, 9], which we adopt. Specifically, the learning procedure alternates between maximizing the score function over the latent variables for each positive ($y_i = +1$) and negative sample ($y_i = -1$), and minimizing the SVM objective over the parameter vector \mathbf{w} . However, when we employed this scheme, we noticed that the SVM objective was affected by the imbalance between the number of positive and negative examples. Consequently, the algorithm focused on satisfying the constraints of the negative samples, neglecting the constraints on the positive samples. The same empirical observation has been reported previously [2]. Hence, we also address this issue by adopting the ranking SVM algorithm [12] in lieu of the traditional SVM objective:

$$\begin{aligned} &\underset{\mathbf{w}, \xi}{\text{minimize}} \quad \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i,j} \xi_{ij} \\ \text{s.t.} \quad &\langle \mathbf{w}, \phi(\mathbf{x}_i, z_i^*) \rangle - \langle \mathbf{w}, \phi(\mathbf{x}_j, z_j^*) \rangle \geq 1 - \xi_{ij}, \\ &\quad \forall i, j, y_j \in \mathcal{Y} \setminus \{y_i\} \\ &\xi_{ij} \geq 0 \quad \forall i, j. \end{aligned} \tag{5}$$

For initialization, we set the pairwise weights $w_{i,j}$ to zero and use the following initialization of the unary term weights w_i : each w_i is set equal to the center of a cluster produced by K-means on the collection of vectors h_k of all of the positive training videos.

4. Experiments

We validate our model on two benchmark datasets: Hollywood1 Human Action (HOHA) [17] and UCF-Sports [23]. HOHA contains 430 videos (240×450 , 24 fps). This dataset is extremely challenging; each video sequence, in addition to the action being performed, contains nuisances such as significant camera motion, rapid scene changes and occasionally significant clutter. Moreover, even though the included actions (e.g., “sit down” or “kiss”) can manifest themselves in a wide variety of conditions, only a tiny portion of them is sampled in the training set. This makes the classification task extremely challenging. Furthermore, many actions are not performed by a single agent (such as “sit down”) but involve interactions with other agents (“kiss”) or objects (“get out of car”). On the contrary, the UCF-Sports dataset consists of actions captured in more constrained environments compared to HOHA videos. In particular, it consists of 150 videos extracted from sports broadcasts that include actions such as “weight-lifting” and “swinging-bench”. However, this dataset poses many challenges as well due to the large displacements that most of the actions contain, the cluttered background, and the large intra-class variability. The ground-truth bounding boxes enclosing the person of interest at each frame are also provided [23] for all actions, except “weight-lifting”.

We annotated the HOHA² dataset with bounding boxes in order to be able to a) quantify our localization performance and b) aid the training phase. The latter is accomplished by restricting the possible selections of *parts* to trajectory groups relevant to the action. This weak supervision improves our algorithm’s ability to learn meaningful parts, enhancing our recognition performance. This can be attributed to the large variability among action instances and the limited size of the training set. We would like to emphasize that this weak annotation is *not* used in the testing phase of the algorithm.

Experimental settings: For all classes of the two datasets, we use $|V| = 3$ graph nodes. For the pairwise relations described in Eq. 3, we use $\eta = 3$ while μ_η, s_η are set to cover the relative velocity domain spanned by the database videos. The penalty parameter C of the regular and ranking SVM objectives is selected with 5-fold cross-validation in the training set. We consider 6 values for the latent variable σ , logarithmically spaced in the interval $[0.5, 2]$. The histograms h_o and h_k of our low-level features (HoG, HoF, HoMB) codeword occurrences are mapped via the approximate feature map for the χ^2 kernel [30]. This allows us to combine the increased discriminative ability of the χ^2 with the efficient training and testing of linear kernels. The computation of the trajectory groups (given the optical flow) and

²Our annotations are available at http://vision.ucla.edu/~raptis/action_parts.html

Table 1. Performance comparison on the UCF-Sports dataset. Mean per-class classification accuracies.

Method	BoW	Our Model	Lan <i>et al.</i> [16]
Accuracy	67.4%	79.4%	73.1%

their descriptors takes approximately 200 seconds in MATLAB/C on a 3GHz PC for a 100-frame video.

Action Recognition. For the HOHA dataset, we evaluate our model following the experimental setting previously proposed [17]. In particular, the test set has 211 videos with 217 labels and the training set has 219 videos with 231 labels, all manually annotated. For each action, we train our model and evaluate its performance considering the average precision (AP) on the precision/recall curve. As mentioned, we observe a boost in the performance (3% increase in mean AP) when we discard the candidate trajectory groups in the training set that have no overlap with the bounding boxes. Another observation is that if we use the regular SVM objective, the mean AP of our method is 38.2% as opposed to 40.1% using the ranking SVM. Table 2 summarizes the results of our model along with competing approaches for comparison. Using only our BoW representation of the videos coupled with a SVM with RBF χ^2 kernel, we obtain a mean AP of 33.4%. Our approach is competitive with most schemes and performs better than Laptev *et al.* [17] that use similar low-level features (HoG, HoF). The performance of our method is lower than the multi-kernel learning approach of Sun *et al.* [27] and the recent work of Shandong *et al.* [26]. This can be attributed to the use of linear kernels compared to RBF kernels, as well as the use of low-level features that do not capture long temporal information, such as the ones proposed by Shandong *et al.* [26]. Our scheme could easily incorporate the latter trajectory based features as part of our trajectory group description. However, we note that the multi-grid spatial binning approach [17, 27] cannot be used in support of other tasks, such as the localization that our model performs.

For the UCF-Sports dataset, we adopt the experimental setting proposed by the recent work of Lan *et al.* [16]. The dataset is split into 103 training and 47 test samples. This separation minimizes the strong correlation of background cues between the test and training set. We note here that earlier studies [23, 34] have used a leave-one-out cross validation setup, which retains the above correlation. Similar to the HOHA dataset experiment, we train our model for each of the ten actions and evaluate our performance using 1-versus-all classification. The mean per-class classification accuracies are summarized in Table 1 and Fig. 3. We notice a significant improvement over the baseline, namely bag-of-words, in several actions. Our model performs worse only in the “skating” action class. We assume this is because the dense trajectory grouping fails to segment distinctive re-

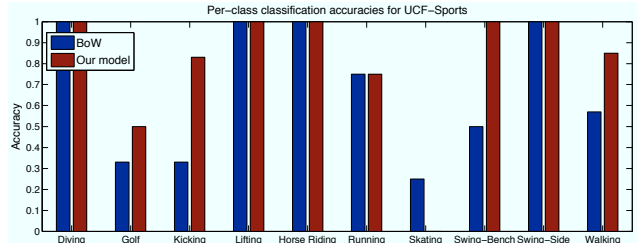


Figure 3. Per-class classification accuracy for UCF-Sports dataset.

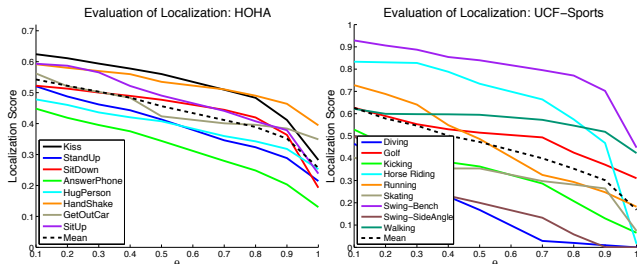


Figure 4. Localization scores for the trajectory groups selected by our algorithm as function of the overlap threshold (θ) for the HOHA dataset (left) and the UCF-Sports dataset (right). The mean localization scores using a high threshold ($\theta = 1$) are 28.6% and 17.1% for HOHA and UCF-Sports datasets respectively, whereas for a low threshold ($\theta = 0.1$) they are 54.3% and 62.6%.

gions of the actors from the background. Thus, we observe a confusion of this action with “golf” and “kicking” actions. All three actions are captured in highly textured outdoor environments.

Action Localization. To evaluate the relevance of the selected trajectory groups to the performed action, we define the localization score as $\frac{1}{|V| \cdot T} \sum_{i=1}^{|V|} \sum_{t=1}^T \mathbb{1}[\frac{|D_{i,t} \cap L_t|}{|D_{i,t}|} \geq \theta]$. L_t is the set of points inside the annotated bounding box, $\mathbb{1}[\cdot]$ is the zero-one indicator function, $D_{i,t}$ is the set of points belonging to the selected trajectory group, and θ is the threshold that defines the minimum overlap of trajectories of a group to consider it as a part of the bounding box. Essentially, we count the average number of utilized trajectories that have length-normalized overlap with the bounding box higher than a threshold θ . Fig. 4 illustrates the average localization score across the *test* videos of each action as well as the mean localization score across all actions of the two datasets. From this figure, we notice that for the overlap threshold $\theta = 0.5$ (i.e., half the points of the trajectory group lie inside the bounding box at a given time instance), we get average localization scores of 48.4% and 47.3% for HOHA and UCF-Sports, respectively. This shows that our method is able to select meaningful trajectory groups as *parts*. Fig. 5 shows the qualitative results of our method in sample frames from the two test sets; for better comprehension of the results, see also the accompanying video. In the UCF-Sports dataset, a significant decrease in

the localization performance is observed for a number of actions (e.g., “diving”, “skating”) while using a high overlap threshold. This can be attributed to the failure in the optical flow estimation and consequently the trajectories of a region, making the trajectories groups less compact. To the best of our knowledge, no such localization results have been reported before for the challenging HOHA dataset. In the case of the UCF-Sports dataset, our localization results are not directly comparable with the ones achieved using the previous approach [16] because our algorithm does not aim at identifying only one “holistic” rectangular region. Moreover, our algorithm does not require the use of a person detector for initialization.

5. Discussion

We have presented an approach to modeling spatio-temporal statistics of video for the purpose of classification of actions, events, or activities. Starting from local spatio-temporal descriptors and dense trajectories, we assemble a mid-level model of individual spatio-temporal regions and their pairwise relations. Our model lends itself for use in standard classification schemes; specifically, we use a latent SVM framework to simultaneously learn the parameters of our models and perform classification.

Testing such models is not straightforward. We use two benchmark datasets that pose several challenges, due to their limited number of training examples and the diversity of the actions. We demonstrate that our method outperforms all local models. Recent advances in multi-grid global schemes have been shown to yield excellent scores on the same benchmarks; based on the complementarity of their information, we expect that further gains can be obtained by a joint treatment in the future. Moreover, part-based models are desirable in action recognition because they support a variety of other tasks beyond straight classification of an entire video shot into one of a few action categories. For instance, we show that they enable localization by flagging the local components (parts) that are most discriminative.

Because our model relies on extracted descriptors, its performance degrades when the low-level data are uninformative, for instance in shots that are too short or too dark and thus yield no discriminative low-level features. Moreover, the huge variability of real-world actions barely can be captured by the small number of instances appearing in current benchmark datasets. We believe that as larger and richer datasets become available, featuring a sufficient number of training data for each action or action component, the localization power of our approach will pay off, especially when used to localize *not* entire complex actions such as “kiss”, but simpler action segments from which more complex actions can be composed.

Acknowledgments. This work was supported by NSF IIS-1018922, ARO W911NF-11-1-0391, DARPA FA8650-11-1-7156 and ANR-10-JCJC-0205.

References

- [1] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as Space-Time Shapes. *IEEE ICCV*, 2005. 2
- [2] M. B. Blaschko, A. Vedaldi, and A. Zisserman. Simultaneous object detection and ranking with weak supervision. In *NIPS*, 2010. 1, 5
- [3] A. Bobick and J. Davis. The recognition of human movement using temporal templates. *IEEE PAMI*, 2001. 2
- [4] W. Brendel and S. Todorovic. Activities as time series of human postures. In *ECCV*, 2010. 2
- [5] W. Brendel and S. Todorovic. Learning spatiotemporal graphs of human activities. In *IEEE ICCV*, 2011. 2
- [6] T. Brox and J. Malik. Object segmentation by long term analysis of point trajectories. In *ECCV*, 2010. 2
- [7] R. Cattell. The scree test for the number of factors. *Multivariate Behavioral Research*, 1966. 3
- [8] T. Cour, P. Srinivasan, and J. Shi. Balanced graph matching. In *NIPS*, 2007. 4
- [9] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *IEEE PAMI*, 2010. 1, 2, 5
- [10] B. Fulkerson, A. Vedaldi, and S. Soatto. Localizing objects with smart dictionaries. In *ECCV*, 2008. 3
- [11] N. Ikizler and D. Forsyth. Searching video for complex activities with finite state models. *IEEE CVPR*, 2007. 2
- [12] T. Joachims. Optimizing search engines using clickthrough data. In *KDD*, 2002. 1, 5
- [13] Y. Ke, R. Sukthankar, and M. Hebert. Event detection in crowded videos. In *IEEE ICCV*, 2007. 2
- [14] A. Kläser, M. Marszałek, and C. Schmid. A spatio-temporal descriptor based on 3D-gradients. In *BMVC*, 2008. 2, 8
- [15] V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *IEEE PAMI*, 2006. 1, 4
- [16] T. Lan, Y. Wang, and G. Mori. Discriminative figure-centric models for joint action localization and recognition. In *IEEE ICCV*, 2011. 2, 6, 7
- [17] I. Laptev, M. Marszałek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *IEEE CVPR*, 2008. 2, 3, 5, 6, 8
- [18] M. Leordeanu and M. Hebert. A spectral technique for correspondence problems using pairwise constraints. In *IEEE ICCV*, 2005. 4
- [19] P. Matikainen, M. Hebert, and R. Sukthankar. Trajectons: Action recognition through the motion analysis of tracked features. In *ICCV workshop on Video-oriented Objected and Event Classification*, 2009. 2, 8
- [20] J. C. Niebles, C.-W. Chen, and L. Fei-Fei. Modeling temporal structure of decomposable motion segments for activity classification. In *ECCV*, 2010. 2
- [21] K. Prabhakar, S. Oh, P. Wang, G. Abowd, and J. Rehg. Temporal causality for the analysis of visual events. In *IEEE CVPR*, 2010. 2

Table 2. Performance comparison on HOHA dataset.

Class	Our Model	Laptev et al. [17]		Yeffet et al. [34]	Raptis et al. [22] BoW	Matikainen et al. [19] BoW	Kläser et al. [14] BoW	Sun et al. [27]		Shandong et al. [26] BoW
		Single	Combined					TTD Combined	TTD-SIFT Combined	
Answer phone	29.5%	26.7%	32.1%	35.1%	26.7%	35.0%	18.6%			48.3%
Get out of car	51.0%	22.5%	41.5%	32.0%	28.1%	7.7%	22.6%			42.3%
Hand shake	35.4%	23.7%	32.3%	33.8%	18.9%	5.3%	11.8%			46.2%
Hug person	30.8%	34.9%	40.6%	28.3%	25.0%	23.5%	19.8%	N/A	N/A	49.3%
Kiss	58.4%	52.0%	53.3%	57.6%	51.5%	42.9%	47.0%			63.6%
Sit down	38.4%	37.8%	38.6%	36.2%	23.8%	13.6%	32.5%			47.5%
Sit up	18.9%	15.2%	18.2%	13.1%	23.9%	11.1%	7.0%			35.1%
Stand up	58.0%	45.4%	50.5%	58.3%	59.1%	42.9%	38.0%			47.3%
MAP	40.1%	32.9%	38.4%	36.8%	32.1%	22.8%	24.7%	30.3%	44.9%	47.6%

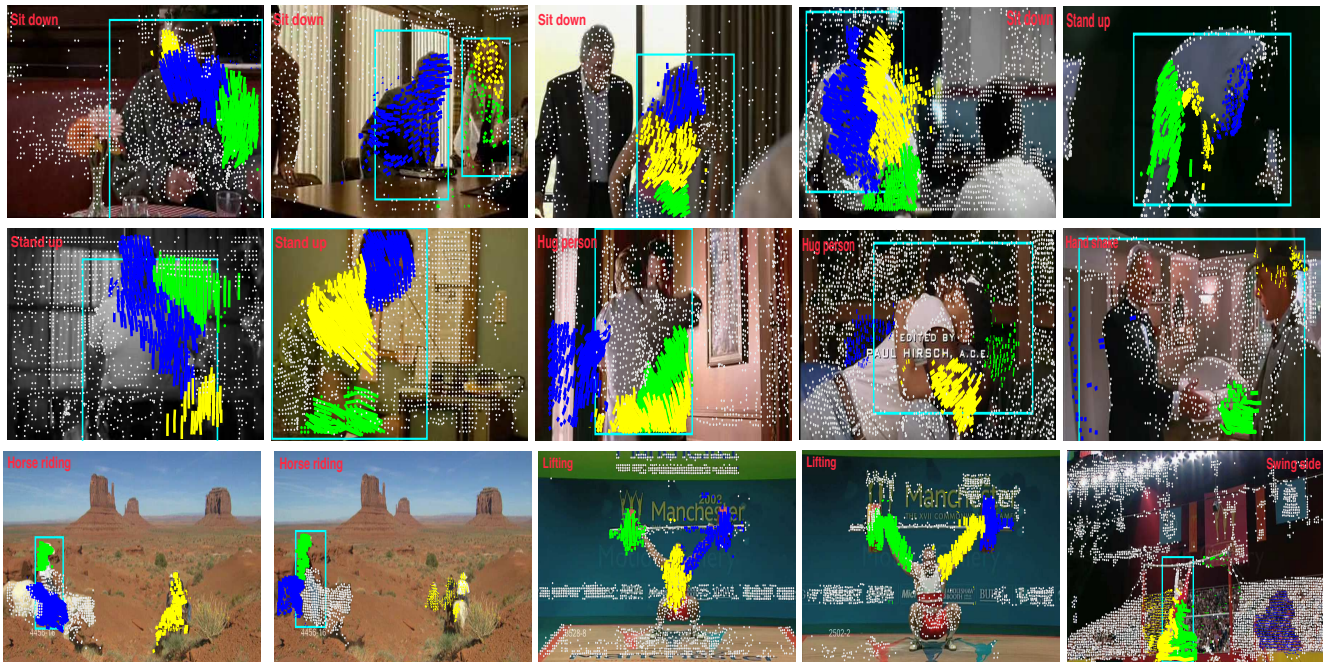


Figure 5. Sample frames from different video sequences of the test sets of the HOHA dataset (first two rows) and the UCF-Sports dataset (third row). The **colored trajectories** represent selected trajectory groups identified by our algorithm. The color indicates the node association in our model. Each trajectory is plotted using its current position and the two previous frame location history. The **white dots** illustrate the current location of the trajectories that were not selected as parts. From these figures, we can observe that the selected trajectory groups lie within the manually annotated bounding boxes, shown in cyan.

- [22] M. Raptis and S. Soatto. Tracklet descriptors for action modeling and video analysis. In *ECCV*, 2010. 2, 3, 8
- [23] M. Rodriguez, J. Ahmed, and M. Shah. Action mach a spatio-temporal maximum average correlation height filter for action recognition. In *IEEE CVPR*, 2008. 5, 6
- [24] C. Schellewald and C. Schnörr. Subgraph matching with semidefinite programming. *Electronic Notes in Discrete Mathematics*, 2003. 4
- [25] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local SVM approach. In *IEEE ICPR*, 2004. 2
- [26] W. Shandong, O. Oreifej, and M. Shah. Action recognition in videos acquired by a moving camera using motion decomposition of lagrangian particle trajectories. In *IEEE ICCV*, 2011. 2, 6, 8
- [27] J. Sun, X. Wu, S. Yan, L. Cheong, T. Chua, and J. Li. Hierarchical spatio-temporal context modeling for action recognition. In *IEEE CVPR*, 2009. 2, 6, 8
- [28] S. Tabatabaei, M. Coates, and M. Rabbat. Ganc: Greedy agglomerative normalized cut. *Arxiv preprint arXiv:1105.0974*, 2011. 3
- [29] L. Torresani, V. Kolmogorov, and C. Rother. Feature correspondence via graph matching: Models and global optimization. In *ECCV*, 2008. 4
- [30] A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. In *IEEE CVPR*, 2010. 5
- [31] H. Wang, A. Klaser, C. Schmid, and C. Liu. Action recognition by dense trajectories. In *IEEE CVPR*, 2011. 2, 3
- [32] Y. Wang and G. Mori. Hidden part models for human action recognition: Probabilistic vs. max-margin. *IEEE PAMI*, 2010. 1, 2
- [33] M. Weber, M. Welling, and P. Perona. Unsupervised learning of models for recognition. In *ECCV*, 2000. 1
- [34] L. Yeffet and L. Wolf. Local trinary patterns for human action recognition. In *IEEE ICCV*, 2009. 6, 8
- [35] A. Yuille and A. Rangarajan. The concave-convex procedure. *Neural Computation*, 15, 2003. 1, 5