Human Activity Recognition in Videos

Vignesh Ramanathan Ankur Sarin Rishabh Goel
Stanford University
{rishgoel, vigneshr, asarin1}@stanford.edu

1. Introduction

Our project deals with the problem of classifying real-world videos by human activity. Such videos usually have a large variation in background and camera motion. This makes the performance of models using low-level appearance and motion features unsatisfactory, particularly in the case of video classes sharing similar objects and background (e.g. ”snatch” and ”clean-jerk” weightlifting actions).

Objects present in a video, and the event label are mutually related. For instance, the presence of a “barbell” in a video would help classify it as a “weightlifting” event. Similarly, we would expect to see a “barbell” in a “weightlifting” video . Thus, recognizing the object presence and motion in a video should aid the event classification task. However, this requires an accurate detection of object tracks, which is a highly challenging task in itself. Works like [12], use humans in the loop to obtain good quality object tracks. However this requires significant human effort. We address this issue by extracting candidate tracks from a video, and modeling the choice of correct tracks as latent variables in a Latent SVM (LSVM) [14]. This formulation enables us to peform action recognition and weakly supervised object tracking in a joint framework. This leads to a more robust as well as discriminative choice of object tracks for event classification. Candidate object tracks are extracted using Deformable Part based Models (DPM) [2] and a tracking algorithm from [12]. We capture the object appearance and motion in a video through features extracted from the object tracks to use in our LSVM model. Finally, we test the performance of our model against different baselines, and show imporvement over the state-of-the-art method on the Olympic Sports Dataset introduced in [7].

2. Related work

While a lot of work has been carried out in the field of action recognition, most of the past works [6, 9, 4, 5] focus on using local shape and motion features to model the video semantics. A few of the methods have also tried to explicitly model object motion in the video by extracting coherent motion tracks and representing them with the aid of local motion and appearance features [3, 13]. [13] has also explicitly extracted human tracks from videos, however unlike our approach, they represent the human track with local features and consider a bag of features collected from all tracks in a video. [8] uses high level human object trajectories similar to our method, however they require an explicit annotation of objects in training videos. In this context, it is to be noted that our method is weakly supervised. We model the spatio-temporal object paths in a video without requiring human or object annotations in the training videos.

3. The Model Formulation

In this section, we present our method for modeling object behavior to recognize actions in videos. The spatio-temporal position of between object tubes provides a good description of actions at a high level. Extraction of correct object tubes from a video is a challenging task in its own right, especially when the annotations on training videos are not available. Our method allows video recognition while facilitating this extraction. Given a video sequence, we first extract a set of candidate tubes for each object. We then adopt a latent SVM framework to model the spatio-temporal object motion where the choice of object tubes acts as latent variables. We elaborate our model formulation in Sec.3.1, and then describe how to extract candidate tubes in Sec.3.2 and perform model learning in

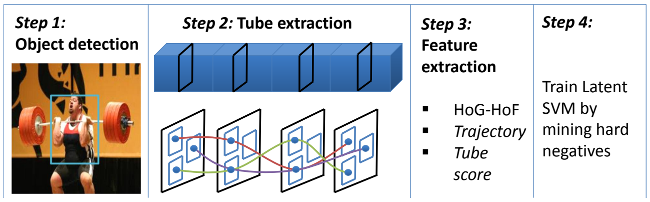

Figure 1. An overview of the iterative training procedure is shown.

Sec.3.3.

3.1. Modeling Object Motion

Given a video sequence, we have a set of object tubes $\tau_O = \{\tau_{O_1}, \cdots, \tau_{O_M}\}$ that correspond to M different objects. A tube is the result of tracking an object across the video. It is represented by a sequence of boxes across frames of the video. Tube τ_i in frame t is represented by $\{x_i(t), a_i(t)\}$, where $x_i(t)$ is the normalized location and size of the corresponding box and $a_i(t)$ is the appearance information of the box.

However, object tracking is difficult on real world videos. Extracting and obtaining reliable tubes itself is challenging. Therefore, instead of assuming that the object tubes are given, we first extract a set of candidate tubes $\mathcal{T}$ for each object. Our method then selects the best tubes τ_O from the candidate set $\mathcal{T}$. Further, we also consider the overall video context through low-level space-time interest point (STIP) [5] features extracted from the video. The recognition score for a video V is

$$s_{\mathbf{w}}(V) = \max_{\tau_O \in \mathcal{T}} \sum_{t=1}^{T} \left\{ \sum_{i=1}^{M} [\alpha_{i,t} \cdot \phi(\tau_{O_i}, t)] \right\} + \eta \cdot \Psi_B, \tag{1}$$

where Ψ_B denotes the vector of STIP features, T is the total number of video frames, $\mathbf{w} = \{\alpha, \eta\}$ denote the feature weights corresponding to different components. $\phi(\tau_{O_i}, t)$ is composed of 3 sets of features that capture the change in appearance and motion of the object over time as given below. We use a χ^2 kernel.

Histogram of Gradient(HoG) and Flow (HoF) Captures the change in appearance and local motion of the object over time.

Trajectory Captures object motion over time by binning the flow vector across 8 directions. The trajectory feature extraction process is illustrated in Fig. 2.

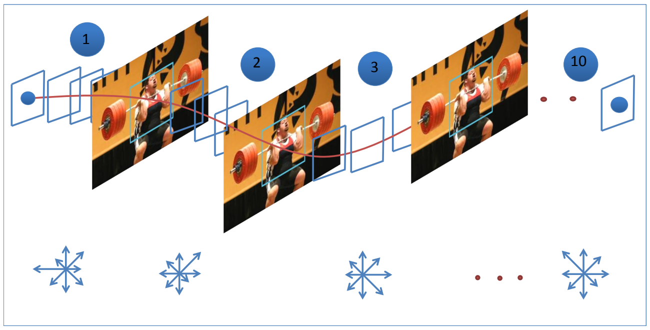

Figure 2. The trajectory features extracted from an object tube are illustrated, when we break the tube into 10 segments.

Tube Score Mean of the object detection scores using [2] for the boxes along the tube.

3.2. Extraction of object tubes

In this section, we discuss the process for extracting candidate tubes from a video for each object. As shown by Fig. 1, we start by running a standard object detector from [2] on each frame of the video. The video is then split into 4 equal segments. From each segment we pick the frame that has the maximum score. Score is defined as the sum of the top 3 object detector scores for the frame, where each box is non-overlapping. Using these 4 frames, with 3 object detections per frame, we initialize a [12] model to extract 81 candidate tubes. Note that the method described above can be adjusted to capture finer variation in tracks.

3.3. Model Learning

In the training stage, the goal is to jointly learn a set of optimal feature weights $\mathbf{w}$ as well as select the best tubes τ_O from the candidates $\mathcal{T}$. We are given a set of training video sequences $\{V_1, \ldots, V_N\}$ where each V_j is assigned a class label y_j. The learning problem can be formulated as an optimization problem similar to latent SVM [2], where the latent variables are the choice of object tubes τ_O. The discriminative cost function is minimized with respect to classifier weights $\mathbf{w}$ as in

$$\min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|_2^2 + C \cdot \sum_{j=1}^{N} \ell\left(s_{\mathbf{w}}(V_j), y_j\right), \tag{2}$$

where $\ell(\cdot)$ is a loss function and $s_{\mathbf{w}}(V_j)$ is specified in Eq.1.

The minimization problem is semi-convex as described in [2]. When the choice of tubes is fixed for positive examples, the optimization in Eq. 2 becomes convex. When the weight vector w is fixed, Eq.1 can be solved to identify the best tubes for positive examples. Hence, the complete opimimzation problem is solved iteratively, where at each iteration $\mathbf{w}$ and (τ_O) for positive example are optimized alternatively, assuming the other variables to be fixed. However, since the search space for object tubes is large, optimization of Eq. 2 with fixed tubes for positives is still very expensive through Stochastic Sub-Gradient (SSG) descent. To speed up the calculations, we use hard-mining for negatives as explained in [2]. We use the CVX optimization package [1] to implement the SSG algorithm. (τ_O) is initialized with tubes corresponding to the best detection scores.

Since the search space for τ_O grows exponentially with the number of objects, a naive search would be very time-consuming. In order to reduce the computation, we use the method of additive kernels [11] to approximate the χ^2 kernel in Eq.1 with linear weights. This reduces the search complexity to be linear in the number of objects considered.

4. Experiments

We present results on 16 events from the Olympic sports dataset for complex event classification. The dataset contains events involving object motion as well as plain human actions without objects. The experiments considered as part of the project only extract human objects from each video.

4.1. Olympic sports dataset

The Olympic sports dataset contains 800 sports videos collected from YouTube. We use the same training and testing splits as used in [10] for easy comparison with the previous works. Each video segment depicts a single event and is temporally localized. Since the videos are well localized, the results are presented for classification of these videos into different event classes.

4.2. Results

The classification results are presented in Tab. 1. We use average precision as a measure to evalutate the model for each event class. Our results are compared with 3 baseline methods as well as a control method explained below:

1. **Bag of Words (BoW)** In this method, low level HoG and HoF features extracted from a video are vector quantized and used in a SVM model. χ^2 kernel is used.

2. **Niebles et al. [7]** This method tries to find the temporal alignment of different low level action segments in a video to classify it into an event class.

3. **Tang et al. [10]** A duration HMM model is used to classify the videos by finding the semantic temporal segments in an event.

4. **No Latent** This corresponds to a control setting, where we present results without a LSVM model. We use features from the highest scoring tubes in each video to train and test a SVM model.

Note that the results for [7, 10] are taken from [10].

We observe that results obtained by using the highest scoring tubes performs worse than a simple BoW model. This confirms our intuition that the highest scoring tubes are not necessarily the correct ones. By treating the choice of object tubes as a latent variable, we are able to identify the most informative tube in a video. This enables us to perform a more accurate classification as demonstrated by the results. Further, we outperform the state-of-the-art method on this dataset. Our method is seen to do better by a significant margin for events where the human motion carries significant information like "high-jump", "long-jump", "diving-springboard" and "javelin-throw". However, the performance drops for "pole-vault" and "tennis-serve", where the performance of the initial DPM detector is bad and leads to poor candidate tubes. This can be accounted to significant deformation of the human in these events.

We further show qualitative results in Fig. 3, where the highest scoring tube in a video as well as the tube chosen by our method are shown for a few test videos from different event classes. Interestingly, we observe that even in the presence of multiple people in a video, our method chooses the person performing the relevant action in the event. The ability to pick the most

Table 1. Average Precision (AP) values for classification on the Olympic dataset. The best performance for each class is highlighted in bold.

Sports class	Baseline (BoW)	Niebles et al. [7]	Tang et al. [10]	Our method	
				No Latent	Full model
high-jump	0.3488	0.2700	0.184	0.2994	**0.4653**
long-jump	0.7602	0.7170	0.8180	0.6864	**0.8571**
triple-jump	0.0864	0.1010	0.161	**0.2925**	0.1424
pole-vault	0.8105	**0.9080**	0.849	0.6659	0.7756
gymnastics-vault	0.8506	0.861	0.857	0.6161	**0.8724**
shot-put	0.4410	0.3730	0.433	0.182	**0.5331**
snatch	0.5853	0.5420	**0.8860**	0.5294	0.6429
clean-jerk	0.7989	0.7060	0.7820	0.6424	**0.8005**
javelin-throw	0.5506	0.8500	0.7950	0.5403	**0.9455**
hammer-throw	0.6444	0.7120	0.7050	0.6747	**0.8219**
discuss-throw	0.4847	0.4730	**0.4890**	0.2643	0.4592
diving-platform	0.9366	**0.9540**	0.937	0.9116	0.9339
diving-springboard	0.7922	0.8430	0.7930	0.6520	**0.8818**
basketball-layup	0.8080	0.8210	**0.8550**	0.5310	0.8225
bowling	0.5460	0.5300	**0.6430**	0.4052	0.5202
tennis-serve	**0.5208**	0.3340	0.4960	0.2703	0.4902
mean AP	0.6228	0.6250	0.6680	0.5102	**0.6853**

discriminative human tube in an event class can be accounted to the max-margin training of our model.

5. Conclusion

In the current work, we developed a method to perform human activity recognition in videos by modeling the object motion through a LSVM framework. Our method performs weakly supervised object tracking in addition to video event classification by treating the choice of object tracks from a candidate pool as the latent variable. We developed an efficient way to restrict the choice of candidate object tracks by using a DPM based initialization. Results from experiments demonstre the increase in performance obtained by the LSVM model over the state-of-the-art event classification scheme. As a next step we wish to quantitatively evaluate the object tracking performed by our algorithm. We would also further experiment with a wider range of objects and datasets.

References

[1] I. CVX Research. CVX: Matlab software for disciplined convex programming, version 2.0 beta.

[2] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *Transactions on PAMI*, 32(9), September 2010.

[3] N. Ikizler-Cinbis and S. Sclaroff. Object, scene and actions: Combining multiple features for human action recognition. In *ECCV*, 2010.

[4] A. Kläser, M. Marszałek, and C. Schmid. A spatio-temporal descriptor based on 3d-gradients. In *BMVC*, pages 995–1004, 2008.

[5] I. Laptev and T. Lindeberg. Space-time interest points. In *ICCV*, 2003.

[6] Q. Le, W. Zou, S. Yeung, and A. Ng. Learning hierarchical spatio-temporal features for action recognition with independent subspace analysis. In *CVPR*, 2011.

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

[8] A. Prest, V. Ferrari, and C. Schmid. Explicit modeling of human-object interactions in realistic videos. Technical report, 2011.

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

[10] K. Tang, L. Fei-Fei, and D. Koller. Learning latent temporal structure for complex event detection. In *CVPR*, 2012.

[11] A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. In *CVPR*, 2010.

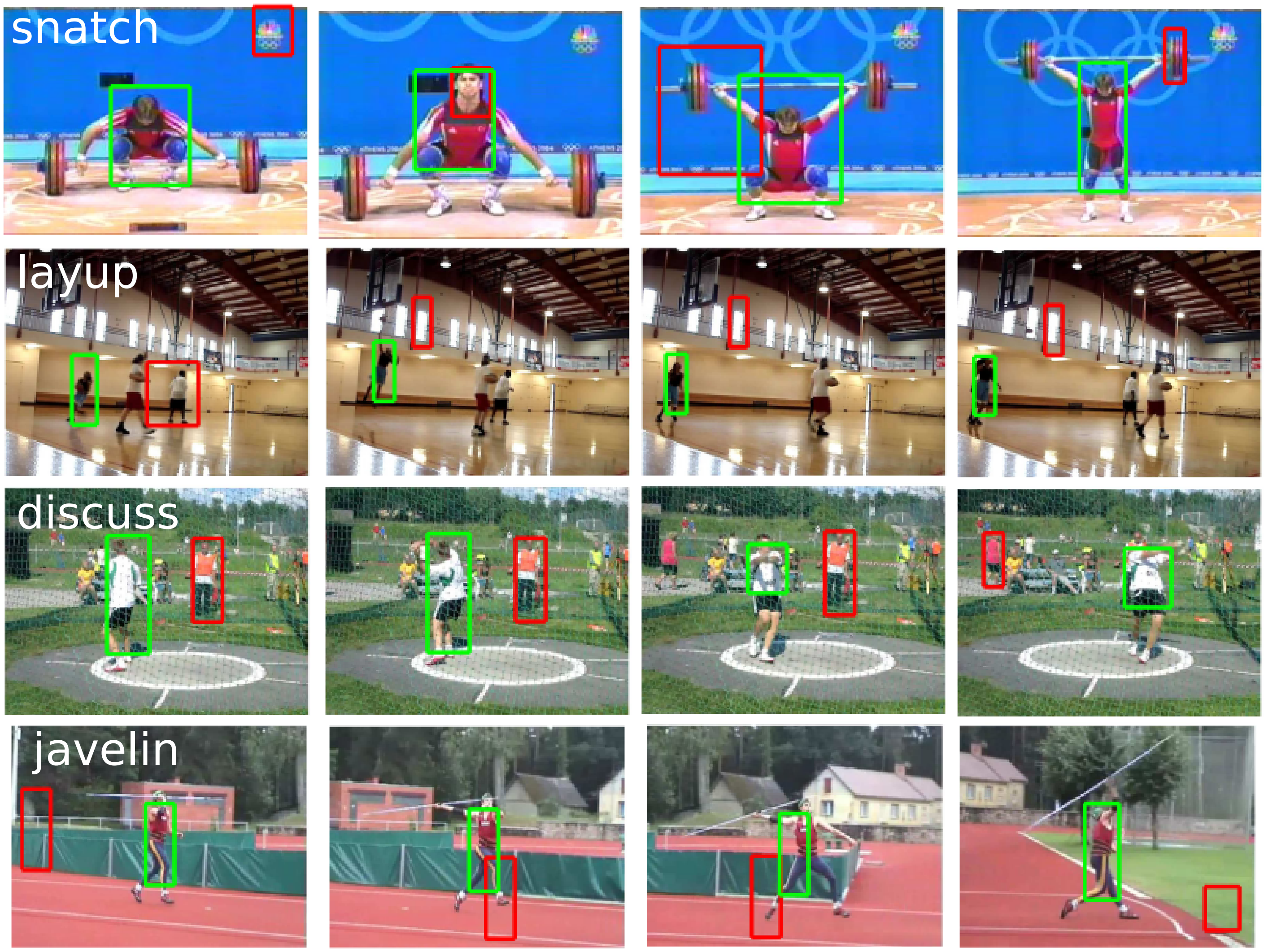

Figure 3. The tubes extracted (human tubes) by our algorithm for different video classes are shown from Olympic dataset. The green boxes indicate the tubes selected after identification of latent variables. The red boxes indicate the tubes selected based on best detection score. It can be seen that the green tube captures the correct object relevant to the event class even in the presence of multiple object instances. (Best viewed in color)

[12] C. Vondrick and D. Ramanan. Video annotation and tracking with active learning. In *NIPS*, 2011.

[13] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu. Action recognition by dense trajectories. In *CVPR*, 2011.

[14] C.-N. Yu and T. Joachims. Learning structural svms with latent variables. In *ICML*, 2009.