Discriminative Figure-Centric Models for Joint Action Localization and Recognition

Tian Lan
School of Computing Science
Simon Fraser University
tla58@sfu.ca

Yang Wang
Dept. of Computer Science
UIUC
yangwang@uiuc.edu

Greg Mori
School of Computing Science
Simon Fraser University
mori@cs.sfu.ca

Abstract

In this paper we develop an algorithm for action recognition and localization in videos. The algorithm uses a figure-centric visual word representation. Different from previous approaches it does not require reliable human detection and tracking as input. Instead, the person location is treated as a latent variable that is inferred simultaneously with action recognition. A spatial model for an action is learned in a discriminative fashion under a figure-centric representation. Temporal smoothness over video sequences is also enforced. We present results on the UCF-Sports dataset, verifying the effectiveness of our model in situations where detection and tracking of individuals is challenging.

1. Introduction

At a broad level, there are two prevalent approaches for action recognition from video sequences. The first is statistical, gathering histograms of local space-time interest points over videos. The second is structural, using human figure-centric representations which maintain spatial arrangements of features with respect to the person. In terms of output, many action recognition approaches do not answer the question of **where** an action takes place in a video, just that it does exist somewhere in the video. In this paper we argue two points, that this action localization is an important component of action recognition, and that doing so can lead to better action classification accuracy.

Impressive action recognition results have been achieved using bag-of-words statistical representations [14]. However, there are limitations in this representation of an action, lacking important cues about the spatial arrangement of features. Recent efforts have been made to extend this representation, using relative arrangements of visual words [20, 13]. Yet these representations still lack explicit modeling of the human figure, and are limited to higher-order statistics of visual word co-occurrences.

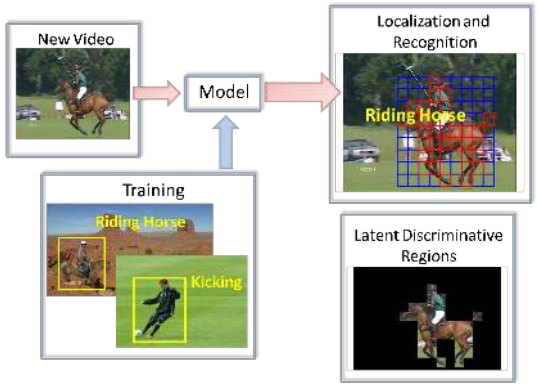


Figure 1: The goal of this paper is to learn a model using training videos annotated with action labels and bounding boxes around people performing the action. Given a new video, we can use the model to simultaneously predict the action label of the video and localize the person performing the action in each frame, including a detailed discriminative region representation.

Structural approaches that use figure-centric representations rely on either a template matching strategy [21] or human detection and tracking as input [5]. Template matching is arguably too brittle for unconstrained videos. Reliable human detectors exist [8], though only for canonical upright poses without occlusion. For recognizing more diverse categories of actions (e.g the UCF Sports dataset [19]), this type of human detector is unreliable. Hence, many action recognition approaches forego a human-centric representation in favour of a purely statistical approach.

In this paper we argue that building figure-centric structural information into an action representation is advantageous, if it can be done robustly. Further, if one is to go beyond just recognizing an action and proceed to localize it within a video, it is essential to actually determine where the action is taking place. The main contribution of this paper is a representation that bridges between a bag-of-words style statistical representation and this type of figure-centric structural representation. To overcome the challenge of per-

son detection in complex actions, we do not pre-suppose the existence of a human detector capable of producing an accurate figure-centric representation. Instead, we treat the position of the human as a latent variable in a discriminative latent variable model, and infer it while simultaneously recognizing an action. We also move beyond a simple bounding box representation for the human figure and learn a more detailed spatial model of an action, discriminatively selecting which cells of a bounding box representation should be included as a model for an action. This detailed action-aware representation explicitly models the discriminative region within a bounding box and thus can be more robust to background clutter. A similar representation is proposed recently to handle occlusion in object detection [11].

2. Previous Work

The literature on vision-based action recognition is immense. Weinland et al. [23] provide a recent survey; we review closely related work here. Visual word representations, in the form of quantized interest point descriptors, are common in the action recognition literature [18, 4, 14]. Beyond first-order bag-of-words representations, Ryoo and Aggarwal [20] developed a matching kernel that considers spatial and temporal relations between space-time interest points. Kovashka and Grauman [13] consider higher-order relations between visual words, each with discriminatively selected spatial arrangements. Instead, our work endows visual word representations with a figure-centric frame of reference. We maintain a spatial coordinate system specifying where features are with respect to the person, while these other bag-of-words methods do not. Klaser [12] evaluates bags-of-words features in different datasets with and without a figure-centric representation obtained using a person detector. Our work also uses a figure-centric visual word representation, though with latent modeling of the person location and a finer level of discriminatively chosen spatial details.

Figure-centric template matching approaches based on shape and motion features are another common strategy [21, 5]. As noted above, assuming reliable human detection and tracking as input to action recognition is problematic. Lu et al. [15] attempt to address the tracking component with a generative probabilistic model for simultaneous tracking and action recognition. Impressive results are shown for small-scale human figures, performing actions in far-field hockey videos. Our work is similar in spirit, though does not require manual initialization of tracks and learns discriminative models for actions with larger shape variations.

In this paper we develop an algorithm to both recognize and localize actions in videos. Similar work has been explored in the object recognition literature. The Implicit Shape Model (ISM) and its extensions [10] have been widely used in object category detection and segmentation. ISM is a generative model that determines possible object locations and scales in a probabilistic Hough voting procedure. ISM can be first used to find hypotheses and a discriminative model is then applied to verify them and filter out false positives. Maji and Malik [16] develop a max-margin formulation of the Hough Transform. Felzenszwalb et al. [8] develop the latent SVM that aligns parts while learning a model. Fergus et al. [9] encode spatial information of interest points and simultaneously localize and recognize objects, using a generative model (pLSA). Deselaers et al. [2] learn models for objects using features inside a bounding box. In our work, we use a latent variable to discriminatively select which information inside the bounding box is useful for our task. In addition, in contrast with all the aforementioned object recognition work, we consider video sequences and we explicitly enforce the temporal coherence of actions across time.

(a) (b) (c)

Figure 2: Background context for action recognition. The three images taken from the UCF-Sports dataset [19] are from the classes: diving, swinging (at the high bar) and swinging (on the pommel) respectively. Context is unhelpful for distinguishing between the actions in (b) and (c), but essential to distinguish the actions in (a) and (b).

3. Recognizing and Localizing Actions in Video

Our goal is to learn a model to both recognize and localize actions in videos, depicted in Fig. 1. We assume we are given a training set of videos with action labels and person locations. We train a model that can determine whether an action occurs in an input video, and the spatio-temporal locations in the video at which the action occurs. Robust solutions will require that many sources of information are brought to bear on the problem. In this paper we incorporate two sources in a unified model – a statistical scene context representation and a structural representation of the individual person.

Global statistical models for scene context are now part of the standard toolkit for action recognition (e.g. Marszalek et al. [17]). Much can be gained from background scene recognition via a global statistical model – in our experiments we show that this effect seems to dominate in a widely-used dataset. However, as Fig. 2 illustrates, global statistical models are not enough, and descriptors on the person may be overwhelmed by the cloud of background features. Clearly, a structural representation that considers a

figure-centric representation is needed here. It is needed not only for differentiating actions where background scenes are similar, but also for localizing the actions.

In this work, we combine a global scene model with a figure-centric representation. One of the challenges in using a figure-centric representation is the need for human detection and tracking in order to form an aligned representation. We take inspiration from the commonly used LSVM-based object detector [8], which implicitly searches for an alignment of images via latent variables. However, that detector is not directly applicable to action recognition in videos. In our work, we make several important modifications to adapt it to this problem domain.

First, analysis of video sequences naturally leads one to consider temporal continuity, or tracking constraints. We use latent variables to represent the location of the person performing the action in each frame. Our tracking constraint enforces the region of the video corresponding to the person (represented by those latent variables) should be consistent in appearance over time. Second, unlike the relatively rigid objects (pedestrians, cars, etc.) for which the LSVM-based detector works well, human figures performing various actions undergo drastic changes in shape. The global "root filter" template and set of parts described with HOG features are insufficient to capture this variation. Instead, we deploy a figure-centric bag-of-words representation combined with a flexible latent sub-region model to capture this variation. Exact learning and inference in our model are intractable. We develop efficient approximate learning and inference algorithms.

3.1. Figure-Centric Video Sequence Model

We propose a discriminative figure-centric model that jointly captures the relationship between the action label of a video and the location of the person performing the action in each frame. The location is represented by a bounding box around the person, and a detailed shape mask that selects certain regions within the bounding box. We divide a bounding box into R cells, and introduce a latent variable to discriminatively select which cells of a bounding box representation are "on". The action label of a video and the bounding boxes of the person performing the action are observed on training data (but not on test data). The discriminative cells of each bounding box are treated as latent variables in the model for both training and test data.

Each training video $\mathbf{I}$ is associated with an action label y. Suppose the video contains τ frames represented as $\mathbf{I} = (I_1, I_2, \ldots, I_\tau)$, where I_i denotes the i-th frame of the video. We use $L = (l_1, l_2, \ldots, l_\tau)$ to denote the set of bounding boxes in the video, one per each frame. The i-th bounding box l_i is a 4-dimensional vector representing location, height, and width of the bounding box. We use $\lambda(l_i; I_i)$ to denote the feature vector extracted from the patch defined by l_i in the frame I_i. We assume $\lambda(l_i; I_i)$ is the concatenation of three vectors, i.e. $\lambda(l_i; I_i) = [\mathbf{x}_i; \mathbf{g}_i; c_i]$. Here $\mathbf{x}_i$ and $\mathbf{g}_i$ denote the appearance feature (codeword from k-means quantized HOG3D descriptors [12]) and spatial locations of interest points in the bounding box l_i respectively. c_i denotes the holistic feature of the patch, here we use a color histogram.

To encode which sub-regions of the bounding box are important for an action, we introduce a matrix $\mathbf{z} = \{z_{ij}\}$, $1 \le i \le \tau, 1 \le j \le R$. An entry $z_{ij} = 1$ means the j-th cell in the i-th frame is discriminative and thus "turned on", and $z_{ij} = 0$ otherwise. In other words, $\mathbf{z}$ specify the discriminative regions inside each bounding box. We treat $\mathbf{z}$ as latent variables and infer them automatically when learning model parameters.

The configurations of bounding boxes in a video are not independent. For example, bounding boxes of neighboring frames tend to be similar in terms of image appearance, location and size. To capture this intuition, we assume that there are connections between bounding boxes in neighboring frames. Intuitively speaking, this will enforce a tracking constraint that states bounding boxes on the action of interest should move smoothly over time. We use a chain-structured undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ to represent the configurations of bounding boxes L in a video. A vertex $v_i \in \mathcal{V}$ corresponds to the configuration l_i of the bounding box in the i-th frame. An edge $(v_i, v_{i+1}) \in \mathcal{E}$ corresponds to the dependency between two neighboring bounding boxes l_i and l_{i+1}.

Inspired by the latent SVM [8, 25], we use the following scoring function to measure the compatibility between a video $\mathbf{I}$, an action label y and the configurations of bounding boxes L that localize the person performing the action in each frame of the video: $f_\theta(L, y, \mathbf{I}) = \max_{\mathbf{z}} \theta^\top \Phi(\mathbf{z}, L, y, \mathbf{I})$, where θ are the model parameters, and $\Phi(\mathbf{z}, L, y, \mathbf{I})$ is a feature vector defined on $\mathbf{z}, L, y, \mathbf{I}$. The model parameters have three parts $\theta = \{\alpha, \beta, \gamma\}$, and $\theta^\top \Phi(\mathbf{z}, L, y, \mathbf{I})$ is defined as:

$$\theta^\top \Phi(\mathbf{z}, L, y, \mathbf{I}) = \sum_{i \in \mathcal{V}} \alpha^\top \phi(l_i, \mathbf{z}_i, y, I_i) + \sum_{i, i+1 \in \mathcal{E}} \beta^\top \psi(l_i, l_{i+1}, \mathbf{z}_i, \mathbf{z}_{i+1}, I_i, I_{i+1}) + \gamma^\top \eta(y, \mathbf{I}) \quad (1)$$

The details of the potential functions in Eq. 1 are described in the following.

Unary Potential $\alpha^\top \phi(l_i, \mathbf{z}_i, y, I_i)$: For the i-th frame I_i, this potential function measures the compatibility between the action label y, the configuration of the bounding box l_i, and the discriminative cells of the bounding box $\mathbf{z}_i$. Recall that we divide a bounding box into R cells, $\mathbf{z}_i$ is a vector that denotes whether each cell in the bounding box l_i is dis-

criminative or not. We define the potential function as:

$$\alpha^\top \phi(l_i, \mathbf{z}_i, y, I_i) = \sum_{a=1}^{Y}\sum_{j=1}^{R}\sum_{w=1}^{K}\sum_{v\in\mathcal{N}(j)} \alpha_{aw}^\top \cdot \mathbb{1}(y=a)\cdot\mathbb{1}(x_{iv}=w)\cdot \text{bin}(g_{iv})\cdot z_{ij} \quad (2)$$

where $\mathcal{N}(j)$ denotes the set of interest points in the j-th cell, we use $\text{bin}(\cdot)$ to denote the feature vector that bins the relative location of an interest point with respect to the center of the bounding box. Hence $\text{bin}(g_{iv})$ is a sparse vector of all zeros with a single one for the bin occupied by g_{iv}. Let Y denote the number of action labels, K denote the number of codewords, B denote the number of bins, then the parameter α is a matrix of size $Y \times K \times B$, where an entry α_{awr} can be interpreted as how much the model prefers to see a "discriminative" interest point in the r-th bin when its codeword is w and the action label is a.

Pairwise Potential $\beta^\top \psi(l_i, l_{i+1}, \mathbf{z}_i, \mathbf{z}_{i+1}, I_i, I_{i+1})$: This potential function measures the compatibility between two neighboring frames and assesses how likely they are to contain the same person. The compatibility is measured in terms of three factors: similarity of bounding boxes, similarity of discriminative regions and similarity of patch appearances. More formally, it is written as:

$$\beta^\top \psi(l_i, l_{i+1}, \mathbf{z}_i, \mathbf{z}_{i+1}, I_i, I_{i+1}) = \beta_1 \cdot s(\mathbf{z}_i, \mathbf{z}_{i+1}) + \beta_2 \cdot m(c_i, c_{i+1}) + \beta_3 \cdot m(l_i, l_{i+1}) \quad (3)$$

where the first term $s(\mathbf{z}_i, \mathbf{z}_{i+1})$ describes the shape similarity between the discriminative regions in the i-th and $i+1$-th bounding box, which is computed as $s(\mathbf{z}_i, \mathbf{z}_{i+1}) = 1 - \frac{1}{R}\sum_{j=1}^{R} |z_{ij} - z_{i+1,j}|$. The second term $m(c_i, c_{i+1})$ measures the similarity between the color histograms of the patches defined by the i-th and $i+1$-th bounding box, where m is a similarity function, here we use the reciprocal value of L_2 distance. The third term denotes the similarity between the i-th and the $i+1$-th bounding box in terms of three cues: location, aspect ratio and area. β is a vector of model parameters that control the relative contributions of these three terms. In essence, this potential function tries to enforce a tracking constraint that two neighboring frames should have similar bounding boxes (in terms of location, aspect ratio and area), similar shaped discriminative regions, and the image patches from the two bounding boxes are also similar.

Global Action Potential $\gamma^\top \eta(y, \mathbf{I})$: Many action classes can be distinguished by scene context. To capture this, we also include a potential function that is a global template model measuring the compatibility between the action label y and a global feature vector of the whole video. It is parameterized as:

$$\gamma^\top \eta(y, \mathbf{I}) = \sum_{a\in\mathcal{Y}} \gamma_a^\top \mathbb{1}(y=a)\cdot x_0 \quad (4)$$

where x_0 is a feature vector extracted from the whole video $\mathbf{I}$. Here we use a statistical bag-of-words style representation for the whole video. The parameter γ_a is a template for the action class a.

4. Learning and Inference

We now describe how to infer the action label given the model parameters (Sec. 4.1), and how to learn the model parameters from a set of training data (Sec. 4.2). In training, the action labels and bounding boxes around people performing the action are provided. At test time, this information is unavailable and must be inferred.

4.1. Inference

Given the model parameters θ, the inference problem is to find the best action label y^* and the best configurations of bounding boxes L^* that localize the person performing the action in each frame of a video $\mathbf{I}$. The inference problem requires solving the following optimization problem:

$$\max_y \max_L f_\theta(L, y, \mathbf{I}) = \max_y \max_L \max_{\mathbf{z}} \theta^\top \Phi(\mathbf{z}, L, y, \mathbf{I}) \quad (5)$$

We can enumerate all the possible $y \in \mathcal{Y}$. For a fixed y, we need to solve an inference problem of maximizing L and $\mathbf{z}$ as follows:

$$\max_L \max_{\mathbf{z}} \theta^\top \Phi(\mathbf{z}, L, y, \mathbf{I}) \quad (6)$$

The optimization problem in Eq. 6 is in general NP-hard since it involves a combinatorial search. One possible solution is to use an iterative method as follows: (1) holding L fixed, find the optimal $\mathbf{z}$; (2) holding $\mathbf{z}$, finding the optimal L. These two steps are repeated until convergence. However, this solution is still not efficient in practice, since the second step of the method requires searching over all the locations and scales of the bounding boxes in each frame of the video. Further, we have to do it during every iteration. This is very computationally expensive. So we further develop an approximation scheme to speed up this step. The intuition of the approximation scheme is to start the configuration space L to be $\mathcal{L}$, which denotes all possible locations and scales of bounding boxes in each frame. During each iteration, we gradually shrink $\mathcal{L}$ by picking the search spaces that are most likely to contain the true locations/scales/aspect ratios of bounding boxes. At the beginning, we still need to do an exhaustive search of all possible locations/scales/aspect ratios. But in subsequent iterations, we only need to search over smaller and smaller sets of possible locations/scales/aspect ratios. The details of the inference method are as follows.

We initialize $\mathbf{z}$ to be a matrix with every entry equals to one, which means all the cells in the bounding boxes are "turned on" in the model. Initially, we set the search

space of the configurations L to be $\mathcal{L}$, i.e. all possible locations, scales, and aspect ratios of the bounding boxes in each frame of a video. We then iterate the following two steps, and reduce the search space of L in each iteration until a final configuration L^* is achieved:

1. Holding the variables $\mathbf{z}$ fixed, find the set $\mathcal{L}'$ of the top $\kappa\%$ scored configurations L according to the score function: $\theta^\top \Phi(\mathbf{z}, L, y, \mathbf{I})$ by exhaustively searching over $\mathcal{L}$ and sorting the scores. Then shrink the search space as $\mathcal{L} \leftarrow \mathcal{L}'$.

2. Enumerate all possible configurations $L \in \mathcal{L}$, and optimize the variable $\mathbf{z}$ according to: $\mathbf{z} = \arg\max_{L \in \mathcal{L}, \mathbf{z}'} \theta^\top \Phi(\mathbf{z}', L, y, \mathbf{I})$.

These two steps are repeated until only one configuration is left in the search space $\mathcal{L}$. The optimization problem in step 1 is done by exhaustive search over $\mathcal{L}$. Since the search space $\mathcal{L}$ shrinks at every iteration, the exhaustive search is expensive only in the first few iterations. The optimization problems in step 1 and step 2 are standard MAP inference problems in undirected graphical models. We use standard belief propagation to solve them.

In practice, in order to reduce the initial search space $\mathcal{L}$, we train a HOG detector [1] on our training set that detects people of any action class. We evaluate the scores returned by the HOG detector for all possible bounding boxes in a video frame I and then use the top 100 bounding boxes according to their scores as the initial search space $\mathcal{L}$ for a video frame. Essentially this very rough person detector (with 100 false positives per frame) acts as a saliency operator. Typically, the top 100 bounding boxes cover all the regions that could possibly contain people and allow us to quickly discard many background regions. This procedure greatly reduces the running time of inference.

4.2. Learning

Given a set of N training examples $\langle \mathbf{I}^n, L^n, y^n \rangle (n = 1, 2, \ldots, N)$, we would like to train the model parameter θ that tends to produce the correct action label y and localize the person performing the action for a new test video $\mathbf{I}$. Note that the bounding boxes L of the person performing the action are observed on training data, but the discriminative regions in each bounding box (or equivalently the variables $\mathbf{z}$) are unobserved and will be automatically inferred.

We adopt the latent SVM framework [8, 25] for learning.

$$\min_{\theta, \xi \geq 0} \frac{1}{2}||w||^2 + C \sum_{n=1}^{N} \xi^n \tag{7a}$$

$$\text{s.t.} \quad f_\theta(y^n, L^n, \mathbf{I}^n) - f_\theta(y, L, \mathbf{I}^n) \geq \Delta(y, y^n, L, L^n) - \xi^n, \forall n, \forall y, \forall L \tag{7b}$$

where $\Delta(y, y^n, L, L^n)$ measures the joint loss between the ground-truth action label and bounding boxes (y^n, L^n) compared with the hypothesized ones (y, L). The joint loss $\Delta(y, y^n, L, L^n)$ should reflect how well the hypothesized action label y and bounding boxes L match the ground truth y^n and L^n. We define the joint loss as a weighted combination of recognition loss and localization loss $\Delta(y, y^n, L, L^n) = \mu \Delta_{0/1}(y, y^n) + (1 - \mu)\Delta(L, L^n)$, where $0 \leq \mu \leq 1$ balancing the relative contributions of these two terms. In our experiments, we set μ to be 0.5. The recognition loss $\Delta_{0/1}$ is a 0-1 loss that measures the difference between the ground-truth action label y^n and a hypothesized action label y, i.e. $\Delta_{0/1}(y, y^n) = 1$ if $y \neq y^n$, and $\Delta_{0/1}(y, y^n) = 0$ otherwise.

The localization loss $\Delta(L, L^n)$ measures the difference between the scales/locations/aspect ratios of the ground-truth bounding boxes L^n and the hypothesized bounding boxes L. This loss function is in turn defined as the sum over a set of local losses on each frame: $\Delta(L, L^n) = \frac{1}{\tau} \sum_i \Delta_i(L_i, L_i^n)$, where τ is the number of frames in a video. We use the intersection over union loss used in the PASCAL VOC challenge [6] as the localization loss: $\Delta_i(L_i, L_i^n) = 1 - \frac{Area(L_i \cap L_i^n)}{Area(L_i \cup L_i^n)}$, where the quality of localization is measured based on the amount of area overlap between the predicted bounding box L_i and the ground truth bounding box L_i^n in the i-th frame. $Area(L_i \cap L_i^n)$ is the area of intersection of the two bounding boxes, while $Area(L_i \cup L_i^n)$ is the area of their union.

We use the non-convex bundle optimization in [3] to solve Eq. 7. In a nutshell, the algorithm iteratively builds an increasingly accurate piecewise quadratic approximation to the objective function. During each iteration, a new linear cutting plane is found via a subgradient of the objective function and added to the piecewise quadratic approximation. We omit the details due to space constraints.

5. Experiments

We present results of both action recognition and localization on the UCF-Sports dataset [19] to demonstrate the effectiveness of our model, especially in situations where detection and tracking of individuals are challenging.

5.1. Experimental Settings and Baselines

The UCF-Sports dataset [19] contains 150 videos from 10 action classes: diving, golf swinging, kicking, lifting, horse riding, running, skating, swinging (on the pommel horse and on the floor), swinging (at the high bar), and walking. The videos are taken from real sports broadcasts. Bounding boxes around the person performing the action of interest in each frame are also available.

Reported results [19, 22, 24, 13, 12] on this dataset use Leave-One-Out (LOO) cross validation, cycling each example as a test video one at a time. There are two issues with using LOO on this dataset. First, it is not clear how parameters (e.g. regularizer weightings) are set. Second, there are strong scene correlations among videos in certain

C value	0.1	1	10	100	1000
accuracy	0.434	0.466	0.643	0.819	0.819

Table 1: Accuracies of the bag-of-words model with different C parameters (LOO).

Method	Accuracy
global bag-of-words	63.1
local bag-of-words	65.6
spatial bag-of-words with $\Delta_{0/1}$	63.1
spatial bag-of-words with Δ_{joint}	68.5
latent model with $\Delta_{0/1}$	63.7
our approach	**73.1**

Table 2: Mean per-class action recognition accuracies (splits).

Method	Accuracy
Kovashka et al. [13]	87.3
Klaser [12]	86.7
Wang et al. [22]	85.6
Yeffet et al. [24]	79.3
Rodriguez et al. [19]	69.2
global bag-of-words	81.9
our approach	**83.7**

Table 3: Mean per-class action recognition accuracies (LOO).

classes; many videos are captured in exactly the same location. With LOO, the learning method can exploit this correlation and memorize the background instead of learning the action. As evidence, we tested the performance of a bag-of-words model [22] using a linear kernel with different C parameters (Table 1). We can see the regularizer weighting C greatly affects the results. The best accuracy is achieved when the learning method focuses on the training error instead of a large margin ($C = 1000$). Essentially, the focus is on memorizing the training examples. To help alleviate these problems, we split the dataset by taking one third of the videos from each action category to form the test set, and the rest of the videos are used for training. This will reduce the chances of videos in the test set sharing the same scene with videos in the training set[1].

In order to comprehensively evaluate the performance of the proposed model in terms of both action recognition and localization, we define the following baseline methods to compare with. The first two baseline methods only do action recognition, while the last two baseline methods can do both action recognition and localization. We extract the HOG3D features for interest points detected by dense sampling, using the code from the author's website[2], with the parameter settings following [22]. For all methods we use a 4000 word codebook, and the number of cells (R) is set to 81.

Global bag-of-words: The first baseline is an SVM classifier on the global feature vector x_0 with a bag-of-words representation for a video, which is similar to [22].

Local bag-of-words: The second baseline is similar to the first one. The only difference is that the bag-of-words histogram is computed using only the interest point descriptors within the person's bounding box in a video. Note that in this baseline, ground truth bounding boxes are also used at test time. So strictly speaking, this is not a practical method. We include it here only for comparison purposes.

Spatial bag-of-words: The third baseline is similar to our proposed method. The difference is that it does not use the latent variables $\mathbf{z}$ to select the discriminative regions. In other words, $\mathbf{z}$ are fixed to be all ones. For this baseline, we report the results of using both the classification loss ($\Delta_{0/1}$) and the joint loss of localization and classification (Δ_{joint}).

Latent model with $\Delta_{0/1}$: The last baseline is equivalent to our proposed method, except that it uses the classification loss ($\Delta_{0/1}$) in learning. The comparison with this baseline will demonstrate that it is helpful to choose a loss function that jointly considers action recognition and localization.

Since the first two baseline methods do not perform action localization, we only evaluate them for the action recognition task. The other baselines are evaluated in terms of both action recognition and localization. For simplicity, we use a linear kernel for both SVM and latent SVM, and the SVM classifier is implemented using LIBLINEAR [7].

5.2. Experimental Results

Action Recognition: We summarize mean per-class action recognition accuracies in Table 2. We can see that our method significantly outperforms the baseline methods. Baselines using the $0/1$ classification loss (spatial bag-of-words with $\Delta_{0/1}$ and latent model with $\Delta_{0/1}$) do not show much improvement over the global bag-of-words. We believe this is due to the fact that $\Delta_{0/1}$ does not enforce correct localizations during training. The localization results at test time can be arbitrary, which in turn do not provide much useful information for recognition. We can see significant improvement when optimizing the joint loss of recognition and localization (spatial bag-of-words with Δ_{joint} and our approach). In addition, by introducing the latent variables to suppress the non-discriminative regions inside each bounding box, our approach works significantly better than the baseline (spatial bag-of-words with Δ_{joint}) in the same framework but without using latent variables. Per-class accuracies of our method and the baseline *global bag-of-words* are compared in Fig. 3.

In order to compare our methods with the state-of-the art on the UCF sports dataset, we also report our results in the LOO setup in Table 3 (though this setup has problems as stated previously). Our method still outperforms

[1]The training-testing split is available at our website http://www.sfu.ca/~tla58

[2]http://lear.inrialpes.fr/people/klaeser/software

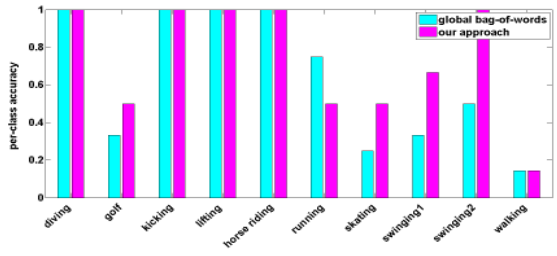
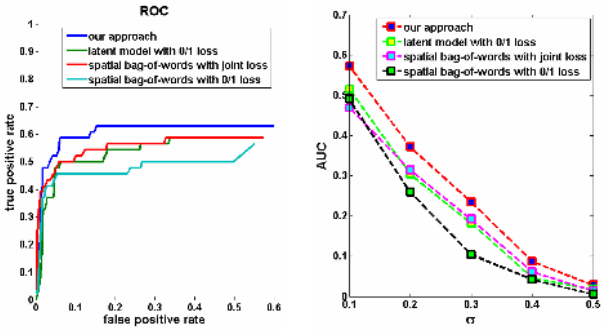


Figure 3: (Best viewed in color) Per-class action classification accuracies of our approach and *global bag-of-words*. *Swinging1* and *swinging2* represent the action classes *swinging (on the pommel horse and on the floor)* and *swinging (at the high bar)* respectively.



(a) (b)

Figure 4: (Best viewed in color) Comparison of action localization performance. (a) ROC curves, when σ is set to 0.2. (b) The comparison of area under ROC (AUC) measures in terms of different σ. σ is the threshold that determines whether a video is correctly localized (see text for explanation).

the baseline of global bag-of-words, with a smaller gap than splitting the dataset. The accuracy of our method is slightly lower than [13] and the best result reported in [22] and [12], we think it is for two reasons: 1) the strong scene correlation between training and test videos in LOO induces the learning method to recognize the background instead of the action, 2) the use of linear versus complex kernels.

Action Localization: Since the first two baselines (global bag-of-words and local bag-of-words) cannot perform action localization, we only evaluate the last three baselines (spatial bag-of-words with $\Delta_{0/1}$, spatial bag-of-words with Δ_{joint} and latent model with $\Delta_{0/1}$) and our approach for action localization in Table 2. Our evaluation criterion is as follows: we compute an "intersection-over-union" score for each frame in a video according to: $O(L_i, L_i^n) = \frac{Area(L_i \cap L_i^n)}{Area(L_i \cup L_i^n)}$, where L_i denotes the predicted bounding box for the i-th frame, and L_i^n is the ground truth bounding box for the i-th frame. The localization score $O(L, L^n)$ for a video is computed by taking an average of the scores $O(L_i, L_i^n)$ of all the frames in a video: $O(L, L^n) = \frac{1}{\tau}\sum_i O(L_i, L_i^n)$, where τ is the number of frames in a video. if the score $O(L, L^n)$ is larger than σ, then the video is considered as correctly localized.

Given a test video $\mathbf{I}$, our model returns $|\mathcal{Y}|$ scores according to $f_\theta(y, L, \mathbf{I})$, where $y \in \mathcal{Y}$. We take each action class as the positive class at one time, and we use the scores $f_\theta(y, L, \mathbf{I})$ to produce ROC curves for each positive class. A video is considered as being correctly predicted if both the predicted action label y^* and the localization L^* match the ground truth. Due to space limitations, we only visualize the average action localization performance of all the action categories in terms of ROC curves, which is computed when σ is set to 0.2. We also evaluate the area under ROC (AUC) measure, with σ varying from 0.1 to 0.5, in a step of 0.1. The curves are shown in Fig. 4.

The localization score $O(L, L^n)$ for each video is computed by taking an average over the "intersection-over-union" scores of all the frames. If we consider a person as being correctly localized based on "intersection-over-union" score larger than 0.5 (see the evaluation criterion in the PASCAL VOC [6]), the threshold $\sigma = 0.2$ can be roughly interpreted as a video is correctly localized if on average 40% of the frames are correctly localized. $\sigma = 0.5$ means almost all the frames in a video are correctly localized, which is a very stringent criterion.

We can draw similar conclusions for the action localization results: optimizing a joint loss leads to better localization and using the latent variables to suppress the non-discriminative regions inside each bounding box also improves the action localization performance.

We visualize the localization results and the learnt discriminative regions inside each bounding box (or equivalently latent variables $\mathbf{z}$) for each action category in Fig. 5 (see the figure caption for detailed explanation).

6. Conclusion

We have presented a discriminative model for joint action localization and recognition in videos. We use a figure-centric visual word representation. Different from previous approaches that require human detection and tracking as input, we treat the position of the human as a latent variable in a discriminative latent variable model, and infer it while simultaneously recognizing an action. We also move beyond a simple bounding box representation for the human figure and learn a more detailed spatial model of an action, discriminatively selecting which cells of a bounding box representation should be included as a model for an action. Our experimental results demonstrate that our proposed model outperforms other baseline methods in both action localization and recognition.

References

[1] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005. 5

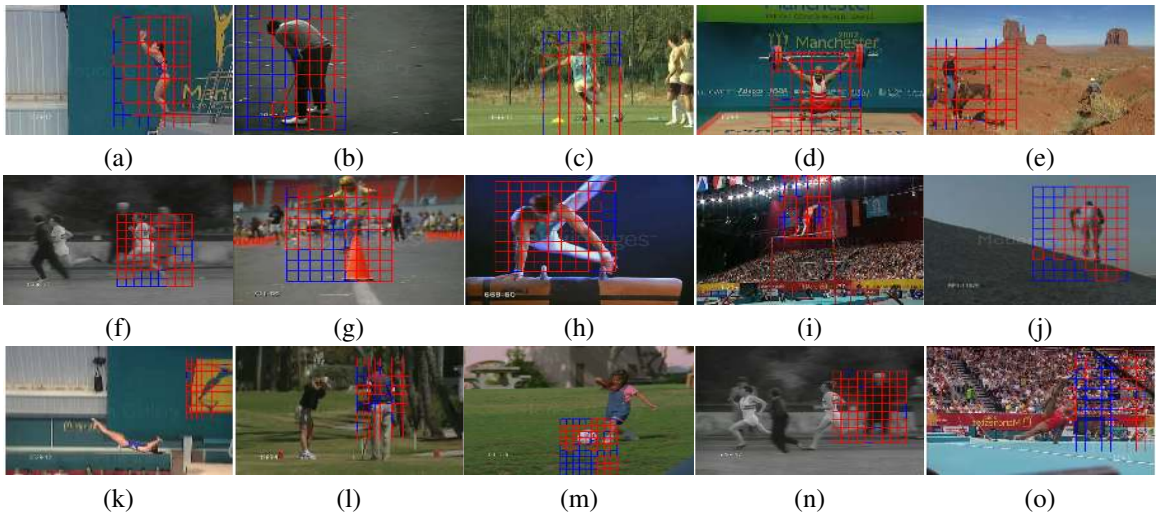


Figure 5: (Best viewed in color) Visualization of localization results and the learnt discriminative cells of each bounding boxes (or equivalently latent variables $\mathbf{z}$) for each action category. The red cells indicate the regions are inferred as discriminative regions by our model, the blue cells indicate the regions are not discriminative. The first two rows show correct examples. We can see that most of the discriminative regions (red cells) are on the person performing the action of interest or discriminative context such as the golf club, soccer ball, barbell, horse and bar in (b)-(e) and (i) respectively. The last row shows incorrect examples. We can see that most of the incorrect localizations are due to background clutter (e.g. the person shaped logo in (k)), or strong scene context (e.g. the soccer ball in (m)).

[2] T. Deselaers, B. Alexe, and V. Ferrari. Localizing objects while learning their appearance. In *ECCV*, 2010. 2

[3] T.-M.-T. Do and T. Artieres. Large margin training for hidden markov models with partially observed states. In *ICML*, 2009. 5

[4] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *VS-PETS*, 2005. 2

[5] A. Efros, A. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *ICCV*, 2003. 1, 2

[6] M. Everingham, L. Gool, C. Williams, J. Winn, and A. Zisserman. The PASCAL visual object classes (VOC) challenge. *IJCV*, 88(2):303–338, 2010. 5, 7

[7] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *JMLR*, 2008. 6

[8] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, 2008. 1, 2, 3, 5

[9] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman. Learning object categories from google's image search. In *ICCV*, 2005. 2

[10] M. Fritz, B. Leibe, B. Caputo, and B. Schiele. Integrating representative and discriminant models for object category detection. In *ICCV*, 2005. 2

[11] T. Gao, B. Packer, and D. Koller. A segmentation-aware object detection model with occlusion handling. In *CVPR*, 2011. 2

[12] A. Klaser. *Learning human actions in videos*. PhD thesis, Universit de Grenoble, 2010. 2, 3, 5, 6, 7

[13] A. Kovashka and K. Grauman. Learning a hierarchy of discriminative space-time neighborhood features for human action recognition. In *CVPR*, 2010. 1, 2, 5, 6, 7

[14] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*, 2008. 1, 2

[15] W.-L. Lu, K. Okuma, and J. J. Little. Tracking and recognizing actions of multiple hockey players using the boosted particle filter. *IVC*, 27(1-2):189–205, 2009. 2

[16] S. Maji and J. Malik. Object detection using a max-margin hough transform. In *CVPR*, 2009. 2

[17] M. Marszalek, I. Laptev, and C. Schmid. Actions in context. In *CVPR*, 2009. 2

[18] J. C. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. *IJCV*, 2008. 2

[19] M. D. Rodriguez, J. Ahmed, and M. Shah. Action MACH: A spatial-temporal maximum average correlation height filter for action recognition. In *CVPR*, 2008. 1, 2, 5, 6

[20] M. Ryoo and J. Aggarwal. Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities. In *ICCV*, 2009. 1, 2

[21] E. Shechtman and M. Irani. Space-time behavior based correlation. In *CVPR*, 2005. 1, 2

[22] H. Wang, M. M. Ullah, A. Klaser, I. Laptev, and C. Schmid. Evaluation of local spatio-temporal features for action recognition. In *BMVC*, 2009. 5, 6, 7

[23] D. Weinland, R. Ronfard, and E. Boyer. A survey of vision-based methods for action representation, segmentation and recognition. *CVIU*, 115:224–241, 2011. 2

[24] L. Yeffet and L.Wolf. Local trinary patterns for human action recognition. In *ICCV*, 2009. 5, 6

[25] C.-N. Yu and T. Joachims. Learning structural SVMs with latent variables. In *ICML*, 2009. 3, 5