Weakly Supervised Action Recognition and Localization using Web Images

Cuiwei Liu, Xinxiao Wu and Yunde Jia

Beijing Laboratory of Intelligent Information Technology,
School of Computer Science, Beijing Institute of Technology,
Beijing100081,P.R.China
{liucuiwei, wuxinxiao, jiayunde}@bit.edu.cn

Abstract. This paper addresses the problem of joint recognition and localization of actions in videos. We develop a novel Transfer Latent Support Vector Machine (TLSVM) by using Web images and weakly annotated training videos. In order to alleviate the laborious and time-consuming manual annotations of action locations, the model takes training videos which are only annotated with action labels as input. Due to the non-available ground-truth of action locations in videos, the locations are treated as latent variables in our method and are inferred during both training and testing phrases. For the purpose of improving the localization accuracy with some prior information of action locations, we collect a number of Web images which are annotated with both action labels and action locations to learn a discriminative model by enforcing the local similarities between videos and Web images. A structural transformation based on randomized clustering forest is used to map Web images to videos for handling the heterogeneous features of Web images and videos. Experiments on two publicly available action datasets demonstrate that the proposed model is effective for both action localization and action recognition.

1 Introduction

Action recognition is an active research topic in computer vision and plays an important role in wide applications such as intelligent video surveillance, content-based video retrieval and human computer interaction. Most of the existing action recognition methods [1–4] focus on recognizing which action exists in a video, regardless of where the action really takes place. In recent years, action recognition and localization have attracted extensive research interests, and some literatures [5–8] engage in jointly predicting which action is performed (recognition) and where the action occurs (localization) in videos. However, most of the action recognition and localization methods require both the annotations of action classes and action locations in each frame for training.

In this work, we aim to build an action recognition and localization system which takes training videos only annotated with action labels as input for alleviating the arduous and time-consuming manual annotations of action locations.

Some recent literatures [9, 10] also consider to localize and recognize actions using weakly annotated training videos. These methods generate candidate spatiotemporal regions without supervision and take one or more spatiotemporal regions discriminative for action recognition as the results of action localization. These methods assume that the most discriminative parts of videos are actually the spatiotemporal regions of the actions. However, for many actions such as diving and bowling, instances usually share similar scenarios. Consequently, regions of background are more discriminative than regions of motions for action recognition, which would lead to incorrect localizations.

To address this problem, we propose a novel Transfer Latent Support Vector Machine (TLSVM) for jointly recognizing and localizing actions in videos by using training videos only annotated with action labels and Web images annotated with both action labels and action locations. The model takes the spatiotemporal regions of actions as latent variables and selects the best one from a set of region candidates in both training and test videos. During the training stage, the local similarities between spatiotemporal regions of interest from training videos and the annotated regions of interest from Web images are enforced to boost both action recognition and localization. At test time, the proposed model is able to automatically predict both the action label and location in an input video. In this paper, bag-of-words representations based on randomized clustering forest are adopted to characterize videos and Web images. Since videos and Web images are represented by heterogeneous features generated from different code books, we introduce a structural transformation based on randomized clustering forest to transform the image feature space to the video feature space. An overview of our approach is illustrated in Fig. 1.

The remainder of this paper is organized as follows. Section 2 reviews the related work. In Section 3, we describe the representation of videos and Web images, including the bag-of-words framework based on randomized clustering forest and the structural transformation from images to videos. The detailed implementation of the proposed TLSVM model is introduced in Section 4. In Section 5, we evaluate the proposed method on the UCF sports dataset and the Olympic sports dataset. Finally, Sections 6 gives the conclusions drawn from the experimental results .

2 Related Work

Some recent literatures [5–8] focus on simultaneously predicting the action label and localizing the action within a video. Yao et al. [5] presented an approach to classify and localize actions using a Hough transform voting framework. They annotated each frame of training examples with a bounding box, in order to obtain normalized action tracks to build a hough forest. An implicit representation of the spatiotemporal shape of an activity is proposed in [6] for localizing and recognizing human actions in unsegmented image sequences, in which the upper and lower bounds of the subjects are manually annotated at each frame. Lan et al. [7] proposed a discriminative model coupling action recognition with person

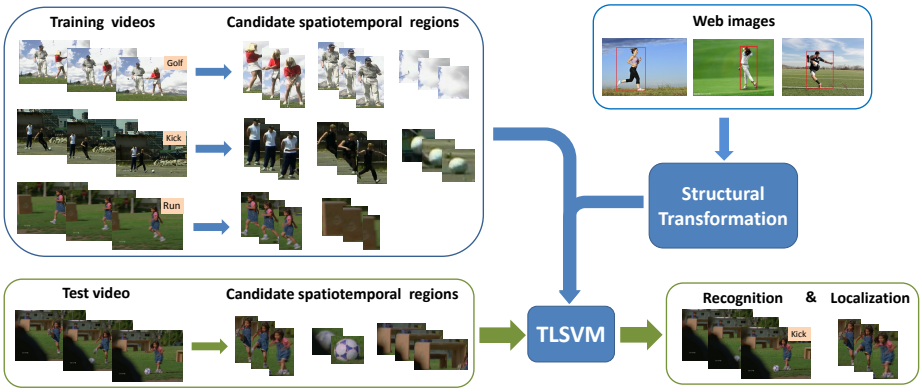


Fig. 1. Overview of the proposed method.

localization. Although this method utilizes a latent region of interest to indicate the action location, it still requires the supervision of latent region in each frame of training videos. Raptis et al. [8] focused on discovering discriminative action parts from clusters of local trajectories that are densely sampled from the videos for action recognition and localization. In their work, strongly supervised bounding boxes of all training frames are extracted to restrict the selection of action parts. All of the aforementioned approaches require the manual annotation of action location for each frame as well as the action label for the whole video.

Shapovalova et al. [9] proposed a SCLSVM model for weakly supervised action recognition and localization. This model aims to advance the recognition performance by enforcing the consistency of local regions among training data, and uses the regions that are most discriminative for recognition as localization results. Ma et al. [10] presented to generate Hierarchical Space-Time Segments in an unsupervised manner, and these segments are utilized as the action representation for classification. In their work, localization of the action is achieved by outputting space-time segments that have positive contributions to the classification. However, in many cases, a region from the background may be chosen as the action localization result due to the similar scenarios shared among training videos with the same action label. Our approach conquers this problem by introducing Web images which are annotated with both action labels and action locations. Local similarities between spatiotemporal regions of interest from training videos and annotated regions of interest from Web images are enforced to boost both action recognition and localization.

There has been recent interests in transferring visual knowledge from images to videos. Duan et al. [11] developed a multiple source domain adaptation method for event recognition in consumer videos by leveraging a large number of Web images from different sources. Chen et al. [12] proposed an event recognition model for consumer videos, using a large number of loosely labeled Web videos and Web images. Both of these methods focus on event recognition without con-

sidering the localization task, while the proposed approach can simultaneously recognize and localize the action in a video. Ikizler-Cinbis and Sclaroff [13] employed action pose classifiers trained with a large image dataset to detect actions in each frame of an input video. A key difference between our approach and [13] is that [13] focuses on transferring knowledge from images to images, while our model is able to transfer knowledge from Web images to videos for recognizing and localizing actions in videos.

3 Representation of Videos and Images

In this section, we first describe how to represent videos and Web images in a bag-of-words framework based on randomized clustering forest [14], and then we present a structural transformation to map images to videos.

3.1 Bag-of-words Representation Based on Randomized Clustering Forest

Bag-of-words model [2] is a popular and powerful method for classification and recognition, which quantizes the low-level local descriptors as a histogram of visual words to get a discriminative mid-level representation. In our work, we use the randomized clustering forest [14] to quantize low-level descriptors effectively.

Web images are characterized by a set of densely sampled low-level HOG descriptors [15] $\{z_l^{HOG}\}_{l=1:N_I}$, and videos are described by dense trajectories [16] $\{z_k^{traj}\}_{k=1:N_V}$. For trajectory k, a descriptor z_k^{traj} is extracted within a space-time volume around the trajectory, and a HOG descriptor z_k^{HOG} is extracted to characterize the spatial patch. The trajectory descriptors $\{z_k^{traj}\}_{k=1:N_V}$ are utilized to construct the randomized clustering forest for videos, while two sets of HOG descriptors $\{z_k^{HOG}\}_{k=1:N_V}$ and $\{z_l^{HOG}\}_{l=1:N_I}$ are integrated to build the randomized clustering forest for images. Moreover, the correspondence between z_k^{traj} and z_k^{HOG} are exploited to learn a transformation from images to videos, which will be described in detail in Section 3.2.

Randomized clustering forest is an ensemble of decision trees, and the tree hierarchies provide a means of clustering low-level local descriptors. Nodes of each tree constitute the hierarchical clusters, namely, the visual words in bag-of-words model. Histograms of visual words in videos are generated from clustering forests built upon trajectory descriptors, while histograms of visual words in images are created from forests built upon HOG descriptors.

Construction of trees. Each tree in a clustering forest is independently grown from a random subset D' of the labeled training low-level descriptors D in a top-down manner. We assume that low-level descriptors share the same label with the video or image they are sampled from. All the training data in D' are dropped down from the root of a tree. In order to split a node n, we randomly generate a set of N_H hypotheses $\{(c_k^n, t_k^n)_{k=1:N_H}\}$, where c_k^n denotes one feature candidate and t_k^n is the corresponding threshold for splitting. Each hypothesis divides the training data arriving at the node n into two subsets, and the one

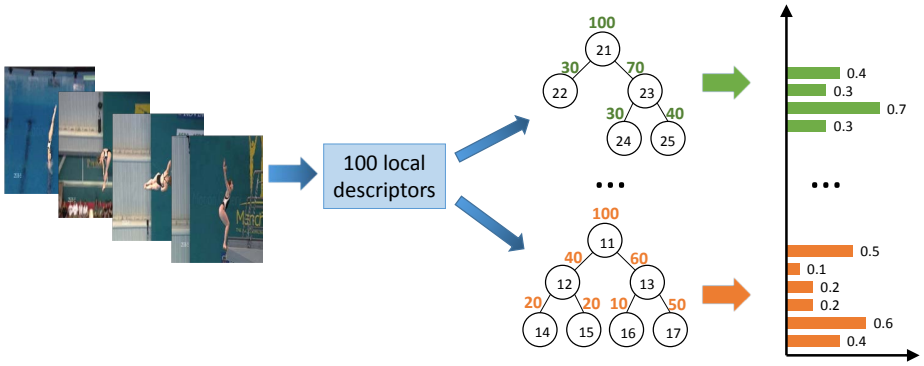


Fig. 2. Generating the mid-level representation for a video.

maximizing the expected information gain is chosen for node split. Growth of a tree is controlled by a maximum tree depth and a minimum amount of samples, so a node stops splitting in the following three cases: (1)The limited tree depth is reached; (2)There are not enough data for splitting; (3)All the data belong to the same class. If one of the above three conditions is satisfied, the node will be treated as a leaf.

Data coding. We take all the nodes (except the root) of each tree, including split nodes and leaf nodes as hierarchical visual words in our framework. Randomized forests for videos and images are built separately by their corresponding training low-level descriptors, we quantize the visual words for videos and images in the same way. Taking a video for example, all the extracted local trajectory descriptors are dropped down from the root of each tree, and the occurrences of nodes across all trees are concatenated to create a normalized histogram H, as shown in Fig.2. Suppose $\mathbf{H}(n)$ to be the occurrence of split node n, then $\mathbf{H}(n)$ can be calculated as

$$\mathbf{H}(n) = \mathbf{H}(n_L) + \mathbf{H}(n_R), \tag{1}$$

where n_L and n_R denote the left and right children nodes of node n, respectively. The hierarchical histogram encodes the structure of each tree, and the relationship among father node and children nodes (defined in Eq. 1) is employed to learn a linear transformation in the next section.

3.2 Structural Transformation

In order to cope with the heterogeneous features of images and videos, a class specific structural transformation is introduced to map the image feature space to the video feature space.

Assume that $RF^V = \{T_r^V\}_{r=1:N_T}$ and $RF^I = \{T_r^I\}_{r=1:N_T}$ are randomized clustering forests for videos and images, respectively, where T_r denotes the r th tree in a forest and N_T is the number of trees. Training trajectory descriptors of videos $\{z_k^{traj}\}_{k=1:N_V}$ are passed through T_r^V from the root, and the corresponding HOG descriptors $\{z_k^{HOG}\}_{k=1:N_V}$ are dropped down to T_r^I simultaneously.

We first learn a set of class specific mapping matrices $\{\mathbf{L}_r^y\}_{y\in Y} \in R^{Nl_r^V \times Nl_r^I}$ among the leaf nodes of T_r^I and T_r^V by using the correspondence between low-level descriptors, where Nl_r^V is the number of leaf nodes in tree T_r^V, and Nl_r^I is the number of leaf nodes in tree T_r^I. Each element $\mathbf{L}_r^y(p,q)$ in matrix $\mathbf{L}_r^y$ is obtained by calculating the amount of samples k of action y, that z_k^{traj} reaches leaf node p of tree T_r^V and z_k^{HOG} goes to leaf node q of tree T_r^I. Normalization is performed on each column of $\mathbf{L}_r^y$ afterwards.

Suppose $\mathbf{H}_r^I$ to be the histogram of an image with action label y, generated by tree T_r^I, and $\mathbf{Hl}_r^I \in R^{Nl_r^I \times 1}$ to be a sub-histogram of $\mathbf{H}_r^I$ corresponding to leaf nodes, we can get a transformed sub-histogram $\mathbf{Hl}_r^V \in R^{Nl_r^V \times 1}$ by defining each element in $\mathbf{Hl}_r^V$ as

$$\mathbf{Hl}_r^V(p) = \sum_{q=1:Nl}^{I_r} \mathbf{L}_r^y(p,q) \cdot \mathbf{Hl}_r^I(q). \tag{2}$$

With the transformed sub-histogram $\mathbf{Hl}_r^V$ of leaf nodes, we can create the transformed histogram $\mathbf{H}_r^V$ of all nodes according to Eq. 1.

Since both of the transformations defined by Eq. 1 and Eq. 2 are linear, the whole transformation from $\mathbf{H}_r^I$ to $\mathbf{H}_r^V$ is also a linear transformation. Transformed histograms of all trees $\{\mathbf{H}_r^V\}_{r=1:N_T}$ are concatenated to form the transformed mid-level representation of the Web image. In the following, we use a matrix $\mathbf{A}$ to represent the linear transformation from the feature space of images to that of videos, for convenience.

4 Transfer Latent SVM Model

The Transfer Latent SVM (TLSVM) Model is able to predict both which action happens and where this action locates in an action video. A few Web images annotated with both the action labels and action locations are employed to learn a discriminative model. Since the annotations of action locations are not available for training videos, the model takes the action location as a latent variable and could automatically select a region of interest from a set of spatiotemporal region candidates. In the rest of this section, we first describe the generation of candidate spatiotemporal regions of interest, and then we present the model formulation, the learning procedure and the inference.

4.1 Candidate Spatiotemporal Regions of Interest

Our goal is to generate a reduced set of candidate spatiotemporal regions of interest for a given video. One intuitive strategy is to extract global 3-dimensional bounding boxes covering the whole action. However, this constrained structure is only applicable for actions with stable locations in a video (i.e, boxing and handshake), and does not work well on drastic actions such as running and walking. In this paper, we independently detect bounding boxes from each frame

by using both the static appearance information and the motion information, and then a two-stage cluster algorithm is introduced to group the bounding boxes into different spatiotemporal regions of interest.

Given an input video, an "objectness" detector [17] is utilized to extract bounding boxes that are likely to contain an object of interest from each frame of the video. Appearance information characterizes the static pattern of an image, while motion information captures the focus of action and allows to discard some irrelevant parts from the background. In order to take advantage of both the static appearance information and the motion information, we compute the boundary map [18] for each frame by merging six appearance channels (i.e, color and soft-segmentation [18]) and two optical flow [19] channels, then the "objectness" detector operates on the boundary maps and returns the potential bounding boxes.

With the detected bounding boxes from each frame, we utilized a two-stage cluster algorithm based on Affinity Propagation [20] to group the bounding boxes into different spatiotemporal regions of interest. Affinity Propagation is an exemplar based cluster algorithm, taking a similarity matrix between samples as input.

In the first stage, Affinity Propagation cluster algorithm is employed to group the bounding boxes into hundreds of clusters based on their appearance similarities and spatiotemporal distances. Intuitively, bounding boxes that are both similar in appearance and adjacent in space and time fall in the same cluster. Given two bounding boxes $B_i = (\mathbf{h}_i, \mathbf{a}_i, \mathbf{c}_i, \mathbf{t}_i)$ and $B_j = (\mathbf{h}_j, \mathbf{a}_j, \mathbf{c}_j, \mathbf{t}_j)$, where $\mathbf{h}_i$ is the color histogram, $\mathbf{a}_i$ denotes the area, $\mathbf{c}_i$ denotes the spatial coordinates for the center point, and $\mathbf{t}_i$ represents the temporal coordinate. The similarity between B_i and B_j is defined as

$$S_B(B_i, B_j) = -\mathcal{D}_h(\mathbf{h}_i, \mathbf{h}_j) - \mathcal{D}_a(\mathbf{a}_i, \mathbf{a}_j) - \mathcal{D}_s(\mathbf{c}_i, \mathbf{c}_j) - \mathcal{D}_t(\mathbf{t}_i, \mathbf{t}_j), \tag{3}$$

where $\mathcal{D}_h$, $\mathcal{D}_a$, $\mathcal{D}_s$ and $\mathcal{D}_t$ denote the χ^2 distance between two color histograms, the difference between the area, the spatial Euclidean distance between two center points and the temporal distance between two bounding boxes, respectively. Due to the temporal distance $\mathcal{D}_t$, bounding boxes extracted from temporally distant frames will fall into different clusters, and each cluster is composed of similar bounding boxes from adjacent frames.

In the second stage, Affinity Propagation cluster algorithm is performed on the first-stage clusters, according to the similarities between bounding boxes in different first-stage clusters. This results in tens of second-stage clusters, and bounding boxes appear in the same second-stage cluster form a spatiotemporal region of interest. The similarity between two first-stage clusters C_k^1 and C_l^1 is defined as

$$S_C(C_k^1, C_l^1) = \max_{i,j:B_i \in C_k^1, B_j \in C_l^1} -\mathcal{D}_h(\mathbf{h}_i, \mathbf{h}_j) - \mathcal{D}_a(\mathbf{a}_i, \mathbf{a}_j) - \mathcal{D}_s(\mathbf{c}_i, \mathbf{c}_j). \tag{4}$$

Different from Eq.3, the similarity measure in Eq.4 does not take the temporal distance $\mathcal{D}_t$ of bounding boxes into consideration. Accordingly, similar bounding

boxes from adjacent frames are grouped into clusters in the first stage, and then distant first-stage clusters with similar appearances are allowed to be clustered together in the second stage.

4.2 Model Formulation

Let $\mathcal{D}^V = \{(x_i, y_i)_{i=1:N}\}$ be the training videos, where $y_i \in Y$ is the action label of video x_i, and the unobserved action locations $\{h_i\}_{i=1:N}$ of videos are treated as latent variables in our model. The latent variable h_i specifies a local spatiotemporal region in video x_i. Our method aims to learn a discriminative compatibility function $F(x, y)$ which measures how compatible the action label y is suited to an input video x:

$$F(x, y) = \max_h f_\omega(x, y, h),$$

$$f_\omega(x, y, h) = \omega^T \Phi(x, y, h),$$

where ω is the learned parameter of the model, and $\Phi(x, y, h)$ is a joint feature vector which describes the relationship between the action video x, the action label y, and the latent action location h.

The model parameter includes two parts $\omega = \{\alpha; \beta\}$. The relationship between an action video x, an action label y and the latent region h is formulated as

$$\omega^T \Phi(x, y, h) = \alpha^T \varphi_1(x, y) + \beta^T \varphi_2(x, h, y), \tag{5}$$

$$\alpha^T \varphi_1(x, y) = \sum_{t=1}^{N_y} \alpha_t^T \cdot \phi(x) \cdot \mathbf{I}(y = t),$$

$$\beta^T \varphi_2(x, h, y) = \sum_{t=1}^{N_y} \beta_t^T \cdot \psi(x, h) \cdot \mathbf{I}(y = t),$$

where $\mathbf{I}(y = t)$ is an indicator function, with $\mathbf{I}(y = t) = 1$ if $y = t$ and 0 otherwise. The potential function $\alpha^T \varphi_1(x, y)$ captures the global relationship between an action video x and the action label y, where $\phi(x)$ denotes a mid-level representation obtained by the random clustering forest using low-level trajectory descriptors extracted from the whole video. The potential function $\beta^T \varphi_2(x, h, y)$ measures the compatibility between a local region h and the action label y, where $\psi(x, h)$ is also a mid-level feature vector, but only using low-level trajectory descriptors extracted from a local region of x specified by the latent variable h.

4.3 Learning

Given a set of weakly labeled training videos $\mathcal{D}^V = \{(x_i, y_i)_{i=1:N}\}$ and a few Web images $\mathcal{D}^I = \{(x_j^I, y_j^I, h_j^I)_{j=1:M}\}$, where $y_j^I \in Y$ is the action label of image x_j^I and h_j^I indicates the spatial location of the person, our goal is to learn the

model parameter ω. Since the unobserved action locations of training videos $\{h_i\}_{i=1:N}$ are treated as latent variables, the model is formulated in a latent structural SVM framework for learning:

$$\min_{\omega,\xi_i,\xi_j^I,\xi_i^s} \frac{1}{2}\|\omega\|^2 + C_1\sum_{i=1}^{N}\xi_i + C_2\sum_{j=1}^{M}\xi_j^I + C_3\sum_{i=1}^{N}\xi_i^S, \tag{6}$$

$$\text{s.t.} \quad f_\omega(x_i,y_i,h_i) - f_\omega(x_i,y',h') \geq \Delta(y_i,y') - \xi_i; \forall y', \forall h', \forall i; \tag{7}$$

$$g_\omega(x_j^I,y_j,h_j) - g_\omega(x_j^I,y',h_j) \geq \Delta(y_j,y') - \xi_j^I; \forall y', \forall j; \tag{8}$$

$$\min_{j:y_i=y_j} \frac{1}{Z_{x_i}} \cdot \Theta((x_i,h_i),(x_j^I,h_j)) \leq \xi_i^S \tag{9}$$

where ξ_i and ξ_i^S are slack variables for training video x_i, and ξ_j^I is the slack variable for Web image x_j^I. The normalization factor Z_{x_i} for video x_i is defined by

$$Z_{x_i} = \max_h \min_{j:y_i=y_j} \Theta((x_i,h),(x_j^I,h_j)). \tag{10}$$

Eq. 7 represents the usual latent SVM max margin constraints which optimize ω by classifying training videos correctly. The loss function $\Delta(y,y')$ measures the cost of predicting the truth label y as action label y'. We define $\Delta(y,y')$ as a simple Hamming loss: $\Delta(y,y')$ is 1 if $y \neq y'$ and 0 otherwise.

Eq. 8 denotes the max margin constraints for the transferred Web images. The constraints defined in Eq. 7 and Eq. 8 compel the model to classify both the Web images and the training videos. Different from the training videos, the Web images are annotated with the regions of actions, therefore Eq. 8 does not include any latent variables. $g_\omega(x^I,y,h)$ is the score function for Web images, defined by

$$g_\omega(x^I,y,h) = \sum_{t=1}^{N_y}\alpha_t^T \cdot \mathbf{A} \cdot \phi(x^I) + \sum_{t=1}^{N_y}\beta_t^T \cdot \mathbf{A} \cdot \psi(x^I,h).$$

where $\mathbf{A}$ is a learned mapping matrix transforming the image feature space to the video feature space, as Web images and videos are represented by heterogeneous features with different dimensions.

Eq. 9 enforces the local similarities between training videos and Web images, which means that the latent regions of training videos should resemble the regions of actions annotated in Web images. According to this constraint, TLSVM model is inclined to choose latent regions with more similarity or less distance to the annotated local regions of images, which benefits both classification and localization. Here we define the loss function $\Theta((x_i,h_i),(x_j^I,h_j^I))$ as a pair-wise distance to estimate the similarity between a local region of image and a latent region of video, which can be directly calculated using mapping matrix $\mathbf{A}$ as

$$\Theta((x_i,h_i),(x_j^I,h_j^I)) = d(\psi(x_i,h_i),\mathbf{A}\cdot\psi(x_j^I,h_j^I)),$$

A variety of distance functions can be employed to measure the similarity between a video and an image, and we adopt the χ^2 distance which is suitable for histogram similarity estimation.

The optimization problem in Eq. 6 is non-convex since the latent variables $\{h_i\}_{i=1:N}$ are not observed during learning. Therefore we employ the non-convex bundle optimization algorithm [21]. This algorithm iteratively builds a gradually accurate piecewise quadratic approximation, and converges to an optimal solution of parameter ω. At each iteration, calculation of the subgradient is required to add a new linear cutting plane to the piecewise quadratic approximation.

The objective function in Eq. 6 can be rewritten in an unconstrained form:

$$O(\omega) = \min_{\omega} \frac{1}{2}\|\omega\|^2 + \sum_{i=1}^{N}(L_i - R_i) + \sum_{j=1}^{M} P_j^I, \tag{11}$$

where L_i, R_i and P_j^I are defined by

$$\begin{aligned}
L_i &= C_1 \max_{y',h'}[f_\omega(x_i, y', h') + \Delta(y', y_i)], \\
R_i &= \max_{h_i}[C_1 f_\omega(x_i, y_i, h_i) - \frac{C_3}{Z_{x_i}} \min_j \Theta((x_i, h_i), (x_j^I, h_j))], \\
P_j^I &= C_2\{\max_{y'}[g_\omega(x_j^I, y', h_j^I) + \Delta(y', y_j)] - g_\omega(x_j^I, y_j, h_j^I)\}.
\end{aligned}$$

Assume that (y_i^*, h_i^*), h_i and y_j^* are solutions to L_i, R_i, and P_j^I, respectively, the subgradient of $O(\omega)$ in Eq. 11 can be calculated by

$$\begin{aligned}
\partial_\omega(O(\omega)) =& C_1 \sum_{i=1}^{N}(\Phi(x_i, y_i^*, h_i^*) - \Phi(x_i, y_i, h_i)) \\
&+ C_2 \sum_{j=1}^{M}(\Phi(x_j^I, y_j^*, h_j^I) - \Phi(x_j^I, y_j, h_j^I)).
\end{aligned}$$

We enumerate y', h' and h_i to find the optimal (y_i^*, h_i^*), h_i and y_j^*.

4.4 Inference

With the learned parameter ω, the inference problem is to simultaneously find the best action label y^* and the best latent region h^* given an input video x. The inference is equal to the following optimization problem:

$$(y^*, h^*) = \arg\max_{y,h} \omega^T \Phi(x, y, h). \tag{12}$$

We can solve Eq. 12 by enumerating all the possible action labels y and latent regions h for a test video x, as the set of possible values for y and h is limited.

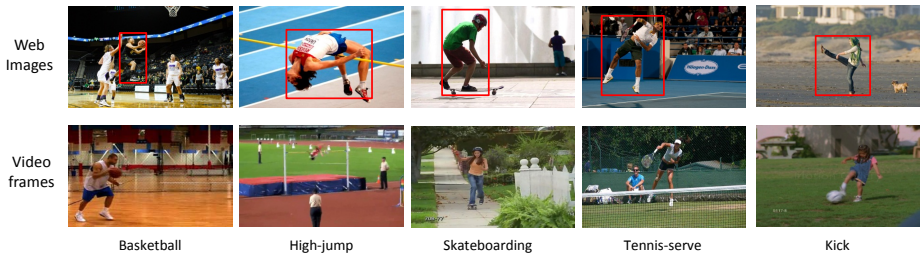


Fig. 3. Examples of the Web images and the Video frames.

5 Experiments

5.1 Dataset and Settings

We evaluate our method on the UCF sports dataset [22] and the Olympic sports dataset [23]. The UCF sports datase contains 150 sports videos of 10 different human actions. Videos in this dataset are extracted from sports broadcasts, and bounding boxes of the person performing the action are provided for each frame. The test strategy proposed in [7] is adopted, in which one third of the videos are selected for testing, leaving the rest for training. The Olympic sports dataset consists of 783 sports videos of 16 action classes. Complex sports actions, drastic camera motions, poor light and large variations of human appearance augment the difficulty of both action recognition and localization. The whole dataset is split into 649 videos for training and 134 videos for testing. We annotate the Olympic sports dataset with bounding boxes in order to quantify our localization performance. We use Image Search Engine to download images from the Web taking the action class labels as query keywords, and annotate a bounding box around the person of interest for each Web image. Examples of the Web images are shown in Fig. 3.

In our implementation, HOG and MBH descriptors of dense trajectory [16] are extracted from videos, and HOG descriptors are densely sampled from the Web images. We randomly select 100,000 training descriptors to build the clustering forests for videos and images. The clustering forest of Web images consists of five trees, and the depth of each tree is limited to 12. The clustering forest of videos consists of five trees, and the depth of each tree is limited to sixteen and eleven for the UCF sports dataset and the Olympic sports dataset, respectively.

We compare the proposed approach with three baseline methods:

Global linear SVM model without images. It only considers the first potential function $\alpha^T \varphi_1(x, y)$ in Eq. 5, which captures the global relationship between a video x and the action label y. A linear SVM classifier is trained on the global representations of training videos. Note that this method can only assign an action label to a test video, without predicting the location of person.

Latent SVM model without images. It is similar to our method, except that no Web images are employed. Regions of interest are also treated as latent

Table 1. Action recognition accuracy comparison with three baselines.

Method	UCF	Olympic
Linear SVM	0.711	0.643
Latent SVM	0.794	0.695
TLSVM (Video frames)	0.844	0.715
TLSVM (Web images)	0.869	0.727

variables, but the local similarities between training videos and Web images are not enforced in this model. Particularly, only the parameter ω under the constraint in Eq. 7 is optimized, and the constraints in Eq. 8 and Eq. 9 are neglected.

TLSVM model using frames from the training videos. Instead of using Web images, this baseline method employs frames randomly selected from the training videos to learn the model. With this baseline method, we aim to assess the benefit of introducing Web images for training.

5.2 Experimental Results

Action Recognition. The proposed approach is compared with the three baseline methods, and the results are summarized in Table 1. It is observable that the proposed approach significantly improves the recognition accuracy compared with the first two baseline methods, which demonstrates the effectiveness of leveraging annotated images for training the model. Meanwhile, our method performs slightly better than the third baseline method, in which the Web images are replaced by images selected from the training videos. A major cause of the performance improvement is that our method avoids the problem of overfitting. Furthermore, the Latent SVM method achieves better performance than the Linear SVM method, which leads a conclusion that incorporating local spatiotemporal information benefits the recognition of action videos.

Table 2 compares the proposed approach with state-of-the-art methods [7, 9, 8, 10] on the UCF sports dataset. As is shown in Table 2, our approach achieves the best result among all the listed methods. We also compare our method with other methods on the Olympic sports dataset. We evaluate the mean average precision for all categories and show the results of different methods in Table 3. It is observable that the proposed method achieves better performance than the methods listed in Table 3.

Action Localization. We adopt the evaluation criterion in [7] and compute the ROC curves of each action class. Given a video, the IOU (intersection-over-union) score is computed for each frame, and the average IOU score over all test frames is compared to a predefined threshold ν to decide whether this video is successfully localized. A test video is considered to be correctly predicted if it is correctly classified and the average IOU score is larger than ν. The action localization results on the UCF sports dataset and the Olympic sports dataset

Table 2. Mean action recognition accuracy of each class for different methods on the UCF sports dataset.

Method	Accuracy
Lan et al. [7]	0.731
Shapovalova et al. [9]	0.753
Raptis et al. [8]	0.794
Ma et al. [10]	0.817
Our method	0.869

Table 3. Mean Average Precision (MAP) of each class for different methods on the Olympic Sports dataset.

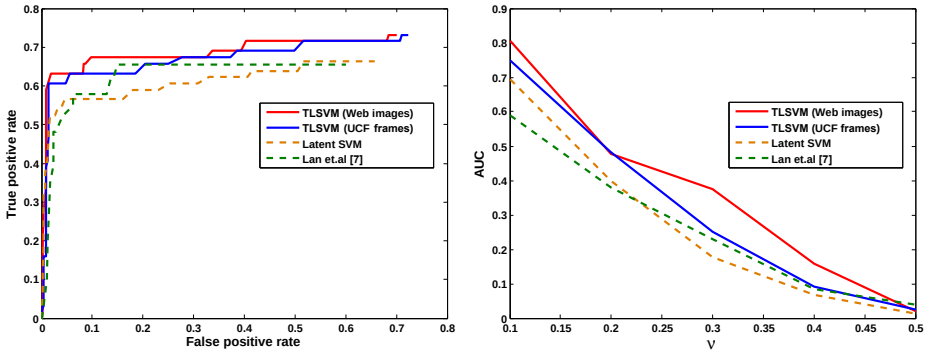
Method	MAP
Niebles et al. [23]	0.625
Tang et al. [24]	0.668
Liu et al. [25]	0.743
Li et al. [26]	0.765
Our method	0.771

are shown in Fig. 4 and Fig. 5, respectively. Fig. 4(a) and Fig. 5(a) depict the average ROC curves for all action classes with $\nu = 0.2$. The Area Under ROC curve (AUC) is evaluated with ν varying from 0.1 to 0.5, and the curves are shown in Fig. 4(b) and Fig. 5(b).

From Fig. 4 and Fig. 5, we can see that our method outperforms the last two baseline methods, which demonstrates the effectiveness of introducing the Web images for learning. Our method is also compared with the method of [7] on the UCF sports dataset. As is shown in Fig. 4, although [7] is trained on videos annotated with bounding boxes for each frame, our method could outperform [7] by using a few annotated images. Moreover, in many cases, the proposed approach using Web images performs better than the third baseline method which employs images from training data, especially for $\nu = 0.2$. These results demonstrate the positive effect of introducing Web images into training for action localization.

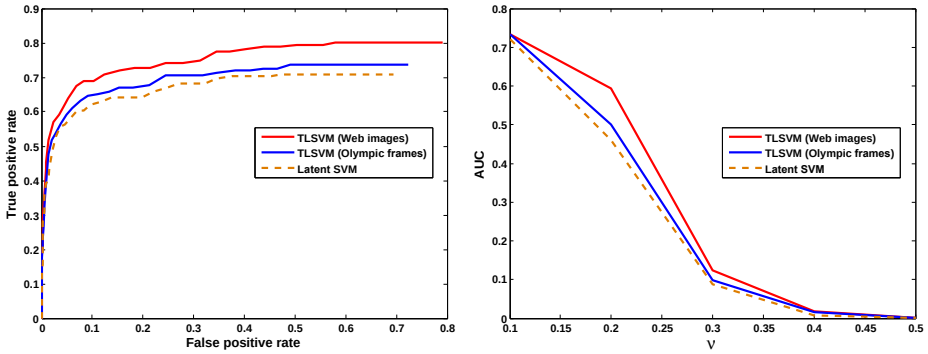
6 Conclusions

We have presented a discriminative Transfer Latent Support Vector Machine (TLSVM) for jointly recognizing and localizing actions in videos. The model is trained on videos only annotated with action labels, and a few Web images annotated with both action labels and action locations are introduced into the learning framework. The spatiotemporal region capturing the action being performed is treated as a latent variable in the proposed model. Since images and videos are represented by different types of features, we introduce a structural transformation that maps images to videos. Experimental results on the UCF



(a) ROC curves for $\nu = 0.2$. (b) Area Under ROC for different ν.

Fig. 4. Comparison of action localization performance on the UCF sports dataset.



(a) ROC curves for $\nu = 0.2$. (b) Area Under ROC for different ν.

Fig. 5. Comparison of action localization performance on the Olympic sports dataset.

sports dataset and the Olympic sports dataset demonstrate that our model can effectively recognize and localize actions in videos.

Acknowledgement.

The research was supported in part by the Natural Science Foundation of China (NSFC) under Grant 61203274, the Specialized Research Fund for the Doctoral Program of Higher Education of China (20121101120029), the Specialized Fund for Joint Building Program of Beijing Municipal Education Commission and the Excellent Young Scholars Research Fund of Beijing Institute of Technology.

References

1. Efros, A.A., Berg, A.C., Mori, G., Malik, J.: Recognizing action at a distance. In: Computer Vision, IEEE International Conference on. (2003) 726–733
2. Niebles, J.C., Fei-Fei, L.: A hierarchical model of shape and appearance for human action classification. In: Computer Vision and Pattern Recognition (CVPR), IEEE Conference on. (2007) 1–8
3. Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: Computer Vision and Pattern Recognition (CVPR), IEEE Conference on. (2008) 1–8
4. Wu, X., Xu, D., Duan, L., Luo, J., Jia, Y.: Action recognition using multilevel features and latent structural svm. In: Circuits and Systems for Video Technology, IEEE Transactions on. Volume 23. (2013) 1422–1431
5. Yao, A., Gall, J., Van Gool, L.: A hough transform-based voting framework for action recognition. In: Computer Vision and Pattern Recognition (CVPR), IEEE Conference on. (2010) 2061–2068
6. Oikonomopoulos, A., Patras, I., Pantic, M.: An implicit spatiotemporal shape model for human activity localization and recognition. In: Computer Vision and Pattern Recognition Workshops (CVPR Workshops), IEEE Computer Society Conference on. (2009) 27–33
7. Lan, T., Wang, Y., Mori, G.: Discriminative figure-centric models for joint action localization and recognition. In: Computer Vision (ICCV), IEEE International Conference on. (2011) 2003–2010
8. Raptis, M., Kokkinos, I., Soatto, S.: Discovering discriminative action parts from mid-level video representations. In: Computer Vision and Pattern Recognition (CVPR), IEEE Conference on. (2012) 1242–1249
9. Shapovalova, N., Vahdat, A., Cannons, K., Lan, T., Mori, G.: Similarity constrained latent support vector machine: an application to weakly supervised action classification. In: ECCV. (2012) 55–68
10. Ma, S., Zhang, J., Ikizler-Cinbis, N., Sclaroff, S.: Action recognition and localization by hierarchical space-time segments. In: Computer Vision, IEEE International Conference on. (2013)
11. Duan, L., Xu, D., Chang, S.F.: Exploiting web images for event recognition in consumer videos: A multiple source domain adaptation approach. In: Computer Vision and Pattern Recognition (CVPR), IEEE Conference on. (2012) 1338–1345
12. Chen, L., Duan, L., Xu, D.: Event recognition in videos by learning from heterogeneous web sources. In: Computer Vision and Pattern Recognition (CVPR), IEEE Conference on. (2013) 2666–2673
13. Ikizler-Cinbis, N., Sclaroff, S.: Web-based classifiers for human action recognition. In: Multimedia, IEEE Transactions on. Volume 14. (2012) 1031–1045
14. Moosmann, F., Nowak, E., Jurie, F.: Randomized clustering forests for image classification. Pattern Analysis and Machine Intelligence, IEEE Transactions on **30** (2008) 1632–1646
15. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition (CVPR), IEEE Conference on. (2005) 886–893
16. Wang, H., Klaser, A., Schmid, C., Liu, C.L.: Action recognition by dense trajectories. In: Computer Vision and Pattern Recognition (CVPR), IEEE Conference on. (2011) 3169–3176

17. Alexe, B., Deselaers, T., Ferrari, V.: Measuring the objectness of image windows. In: Pattern Analysis and Machine Intelligence, IEEE Transactions on. Volume 34. (2012) 2189–2202
18. Leordeanu, M., Sukthankar, R., Sminchisescu, C.: Efficient closed-form solution to generalized boundary detection. In: ECCV. (2012) 516–529
19. Liu, C.: Beyond pixels: exploring new representations and applications for motion analysis. PhD thesis, Massachusetts Institute of Technology (2009)
20. Frey, B.J., Dueck, D.: Clustering by passing messages between data points. In: Science. Volume 315., American Association for the Advancement of Science (2007) 972–976
21. Do, T.M.T., Artières, T.: Large margin training for hidden markov models with partially observed states. In: Proceedings of the 26th Annual International Conference on Machine Learning. (2009) 265–272
22. Rodriguez, M., Ahmed, J., Shah, M.: Action mach: A spatio-temporal maximum average correlation height filter for action recognition. In: Computer vision and pattern recognition (CVPR), IEEE Conference on. (2008) 1–8
23. Niebles, J.C., Chen, C.W., Fei-Fei, L.: Modeling temporal structure of decomposable motion segments for activity classification. In: ECCV. (2010) 392–405
24. Tang, K., Fei-Fei, L., Koller, D.: Learning latent temporal structure for complex event detection. In: Computer Vision and Pattern Recognition (CVPR), IEEE Conference on. (2012) 1250–1257
25. Liu, J., Kuipers, B., Savarese, S.: Recognizing human actions by attributes. In: Computer Vision and Pattern Recognition (CVPR), IEEE Conference on. (2011) 3337–3344
26. Li, W., Vasconcelos, N.: Recognizing activities by attribute dynamics. In: NIPS. (2012) 1115–1123