Action Localization in Videos through Context Walk

Khurram Soomro, Haroon Idrees, Mubarak Shah
Center for Research in Computer Vision (CRCV), University of Central Florida (UCF)
{ksoomro, haroon, shah}@eecs.ucf.edu

Abstract

This paper presents an efficient approach for localizing actions by learning contextual relations, in the form of relative locations between different video regions. We begin by over-segmenting the videos into supervoxels, which have the ability to preserve action boundaries and also reduce the complexity of the problem. Context relations are learned during training which capture displacements from all the supervoxels in a video to those belonging to foreground actions. Then, given a testing video, we select a supervoxel randomly and use the context information acquired during training to estimate the probability of each supervoxel belonging to the foreground action. The walk proceeds to a new supervoxel and the process is repeated for a few steps. This "context walk" generates a conditional distribution of an action over all the supervoxels. A Conditional Random Field is then used to find action proposals in the video, whose confidences are obtained using SVMs. We validated the proposed approach on several datasets and show that context in the form of relative displacements between supervoxels can be extremely useful for action localization. This also results in significantly fewer evaluations of the classifier, in sharp contrast to the alternate sliding window approaches.

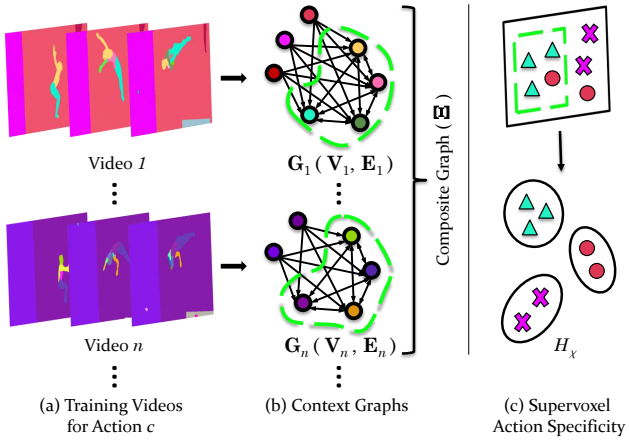

Figure 1. This figure illustrates the idea of using context in the form of spatio-temporal displacements between supervoxels. (a) Given N_c videos for an action c which have been over-segmented into supervoxels, we construct a context graph for each video as shown in (b). Each graph has edges emanating from all the supervoxels to those that belong to foreground action (circumscribed with dashed green contours). The color of each node in (b) is the same as that of the corresponding supervoxel in (a). Finally, a composite graph (Ξ) from all the context graphs is constructed, implemented efficiently using a kd-tree. (c) We also quantify 'supervoxel action specificity' which returns the likelihood of a particular supervoxel belonging to an action and use it in conjunction with context to localize actions.

1. Introduction

The most challenging problems associated with automated analysis of videos are related to actions, with a variety of approaches in computer vision [2, 33] developed to address them. One of the problems is *action recognition* which entails classification of a given video in terms of a set of action labels. With the introduction of uncontrolled datasets, consisting of videos captured in realistic non-experimental settings and longer durations such as those from YouTube [24, 16], *action detection* has emerged as a new problem where the goal is to determine the location of an action in addition to its class. Action detection, which may refer to temporal detection [16] or spatio-temporal action localization [6, 23, 4, 13], is especially difficult when background is cluttered, videos are untrimmed or contain multiple actors or actions. Applications include video search, action retrieval, multimedia event recounting [1], and many others related to video understanding.

Many existing approaches [26, 31] learn an action detector on trimmed training videos and then exhaustively search for each action through the testing videos. However, with realistic videos having longer durations and higher resolutions, it becomes impractical to use sliding window approach to look for actions or interesting events [38, 13, 18]. Analyzing the videos of datasets used for evaluation of action localization such as UCF-Sports [24], JHMDB [15], and THUMOS [16] reveals that, on average, the volume occupied by an action (in pixels) is considerably small com-

pared to the spatio-temporal volume of the entire video (around 17%, using ground truth). Therefore, it is important that action localization is performed through efficient techniques which can classify and localize actions without evaluating at all possible combinations of spatio-temporal volumes.

The use of context has been extensively studied for object detection in images through modeling of the relationships between the objects and their surroundings including background [10, 3, 5], which significantly reduce search space of object hypotheses. However, it is non-trivial to extend such approaches to actions in videos due to the fact that the temporal dimension is very different from the spatial dimensions. An image or a video is confined spatially, but the temporal dimension can be arbitrarily long. The differences in spatial and temporal dimensions also affects the optimal representation of actions in videos [26]. Cuboid, which is the 3D extension of a bounding box in images, is not appropriate for action localization due to the following two reasons: (i) Actions have a variable aspect ratio in space and time as they capture articulation and pose changes of actors. Furthermore, instances of repetitive actions (such as running) can have different lengths depending on the number of cycles captured in the video. (ii) The nature of an action or simply the camera motion can cause an actor to move spatially in a video as time progresses. In such a case, a cuboid would include large parts of the background. Accordingly, the ground truth in action localization datasets consists of a sequence of bounding boxes which change in size and move spatially with respect to time. Each such sequence can be visualized as a rectangular tube with varying height, width and spatial location.

On the same grounds, the results of action localization will be more useful if they contain minimal background, which cannot be achieved with cuboid or sliding window approaches [28, 32, 26, 38]. However, such a powerful representation of actions comes with a cost. Generating tight tubes around the actors makes the task of action localization even more challenging as the action hypotheses not only depend on space and time, but also on tube deformations. An exhaustive search over all possible combinations is wasteful and impractical. In this paper, we formulate the problem of action localization in such a way that the issues associated with cuboid and sliding window approaches are circumvented and use context to significantly reduce the search space of hypotheses resulting in fewer number of evaluations during testing.

For the proposed approach, we over-segment the videos into supervoxels and use context as a spatial relation between supervoxels relative to foreground actions. The relations are modeled using three dimensional displacement vectors which capture the intra-action (foreground-foreground) and action-to-scene (background-foreground) dependencies. These contextual relations are represented by a graph for each video, where supervoxels form the nodes and directed edges capture the spatial relations between them (see Fig. 1). During testing, we perform a context walk where each step is guided by the context relations learned during training, resulting in a probability distribution of an action over all the supervoxels.

There are a few approaches that reduce the search space to efficiently localize actions. To the best of our knowledge, we are the first to explicitly model foreground-foreground and background-foreground spatial relationships for action localization. The proposed approach requires only a few nearest neighbor searches in a testing video followed by a single application of CRF that gives action proposals. The action confidences of proposals are then obtained through SVM. This is in contrast to most of the existing methods [26, 31], which require classifier evaluations several order of magnitudes higher than the proposed approach.

2. Related Work

Action recognition in realistic videos has been an active area of research with several recent surveys [2, 33] published on the subject. With significant progress in action recognition over the past few years, researchers have now started focussing on the more difficult problem of action localization [36, 35, 11, 6, 23, 31, 14]. Ke *et al*. [17] presented an approach for detecting simple events in crowded videos. Yuan *et al*. [38] used branch-and-bound and dynamic programming for action localization using cuboids. Lan *et al*. [18] treated the position of human in the video as a latent variable, inferred simultaneously while recognizing the action, which also helps in localization of the action. Since our representation is constructed from supervoxels, it can provide more accurate localization mimicking segmentation of objects in images.

Jain *et al*. [13] recently proposed a method that extends selective search approach [29] to videos by using supervoxels instead of superpixels. Supervoxels are merged using appearance and motion costs producing multiple layers of segmentation. Selective search yields category-independent hypotheses that are then evaluated for different actions. There have been few similar recent methods for quantifying actionness [4, 37] which yield fewer regions of interest in videos. Similar to these methods, our output is more precise than cuboids, however, we focus on localization through context by learning the relations between background and foreground action supervoxels. Furthermore, our proposed approach generates fewer but class-dependent hypotheses, and the hypotheses for each action are the result of context walk where new observations depend on past observations.

Zhou *et al*. [40] used a split-and-merge algorithm to obtain action segments and classify the segments with LatentSVM [7]. Tran *et al*. [27] used Structured SVM to local-

ize actions with inference performed using Max-Path search method. Me *et al*. [19] automatically discovered spatio-temporal root and part filters for action localization. Tian *et al*. [26] developed Spatiotemporal Deformable Parts Model [7] to detect actions in videos. They use a sliding window approach that can handle deformities in parts, both in space and time. Unlike [19], who used a simple linear SVM on a bag of hierarchical space-time segments representation, they build a spatio-temporal feature pyramid followed by LatentSVM. In contrast to these methods, we propose an efficient approach that requires significantly fewer evaluations for localizing actions.

Context has been used extensively for object detection [10, 3, 5]. Heitz and Koller [10] reduced the number of false positives using context between background and objects. Similarly, Alexe *et al*. [3] used context for object detection in images by learning relations betweens windows in training images to the ground truth bounding boxes. There are several works that use context for action recognition using different significations of the word 'context'. Gupta and Davis [8] attempted to understand the relationship between actions and the objects used in those actions. Han *et al*. [9] also used object context for action recognition . However, both the methods assume that detectors for multiple objects are available. Ikizler-Cinbis and Sclaroff [12] used a variety of features associated with objects, actions and scenes to perform action recognition. They also required person detection using [7]. Marszalek *et al*. [20] used movie scripts as automatic supervision for scene and action recognition in movies. Zhang *et al*. [39] extracted motion words and utilized the relative locations between the motion words and a reference point in local regions to establish the spatio-temporal context for action recognition. Sun *et al*. [25] presented a hierarchical structure to model the context information of SIFT points, and their model consists of point-level, intra-trajectory, and inter-trajectory relationships. Wu *et al*. [34] incorporated context through spatio-temporal coordinates for action recognition.

Our proposed approach is inspired from Alexe *et al*. [3], and differs in several key aspects: First, for precise detection of actions in videos, we cannot use windows or cuboids which can contain significant amounts of background due to articulation, actor/camera movement and naturally from cyclic actions. Furthermore, due to inherent differences between images and videos and the extra degree of freedom due to time, we segment the video into *supervoxels* to reduce the search space of candidate hypotheses. Second, instead of pixels in images, our proposed approach operates on a *graph* where nodes represent supervoxels. Third, since we localize actions using supervoxels instead of 3D windows, we have to infer action locations using a Conditional Random Field on the graph created for the testing video. In summary, ours is the first work that explicitly relies on both foreground actions and background for *action localization* with an emphasis on fewer number of classifier evaluations.

3. Action Localization through Context Walk

The proposed approach for action localization begins by over-segmenting the training videos into supervoxels and computing the local features in the videos. For each training video, a graph is constructed that captures relations from all the supervoxels to those belonging to action foreground (ground truth). Then, given a testing video, we initialize the context walk with a randomly selected supervoxel and find its nearest neighbors using appearance and motion features. The displacement relations from training supervoxels are then used to predict the location of an action in the testing video. This gives a conditional distribution for each supervoxel in the video of belonging to the action. By selecting the supervoxel with the highest probability, we make predictions about location of the action again and update the distribution. This *context walk* is executed for several steps and is followed by inferring the action proposals through Conditional Random Field. The confidences for the localized action segments (proposals) are then obtained through Support Vector Machine learned using the labeled training videos (see Fig. 2).

3.1. Context Graphs for Training Videos

Let the index of training videos for action $c = 1 \dots C$ range between $n = 1 \dots N_c$, where N_c is number of training videos for action c. The ith supervoxel in the nth video is represented by $\mathbf{u}_n^i, i = 1 \dots I_n$, where I_n is the number of supervoxels in video n. Each supervoxel either belongs to a foreground action or the background. Next, we construct a directed graph $\mathbf{G}_n(\mathbf{V}_n, \mathbf{E}_n)$ for each training video across all the action classes. The nodes in the graph are represented by the supervoxels while edges $\mathbf{e}^{ij}$ emanate from all the nodes (supervoxels) to those belonging to the foreground, i.e., supervoxels spatio-temporally contained within the ground truth tube.

Let each supervoxel $\mathbf{u}$ be represented by its spatio-temporal centroid, i.e., $\mathbf{u}_n^i = (x_n^i, y_n^i, t_n^i)$. The features associated with $\mathbf{u}_n^i$ are given by $\mathbf{\Phi}_n^i = ({}_1\phi_n^i, {}_2\phi_n^i, \dots, {}_F\phi_n^i)$, where F is the total number of features. Then, for a particular action c, the graphs $\mathbf{G}_n$ and features $\mathbf{\Phi}_n^i, \forall n = 1 \dots N_c$ are represented by the composite graph $\mathbf{\Xi}_c$ which constitutes all the training information necessary to localize an action during testing. The following treatment is developed for each action class, therefore, we drop the subscript c for clarity and use it when necessary.

3.2. Context Walk in the Testing Video

For a testing video, we obtain supervoxels ($\sim 200 - 300$ per video) with each supervoxel and its local features represented by $\mathbf{v}$ and $\mathbf{\Phi}$, respectively. Then, we construct

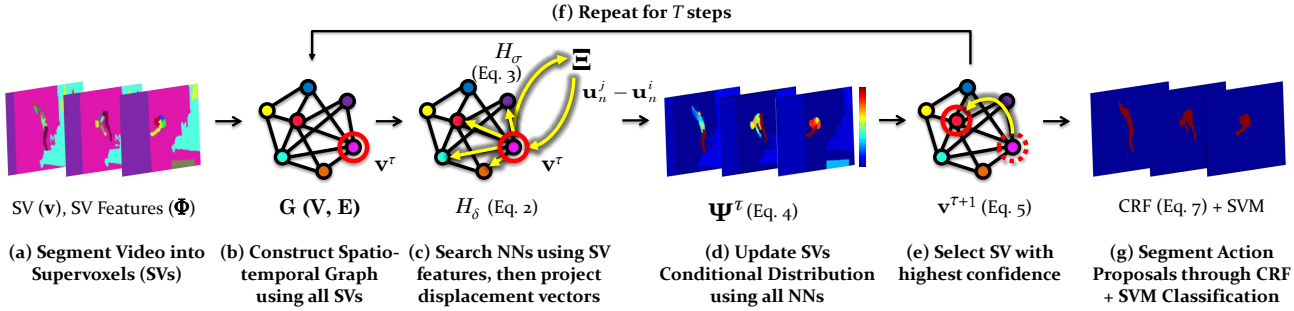

Figure 2. This figure depicts the testing procedure of the proposed approach. (a) Given a testing video, we perform supervoxel (SV) segmentation. (b) A graph $\mathbf{G}$ is constructed using the supervoxels as nodes. (c) We find the nearest neighbors of the selected supervoxel ($\mathbf{v}^\tau$; initially selected randomly) in the composite graph $\mathbf{\Xi}$ which returns the displacement vectors learned during training. The displacement vectors are projected in the testing video as shown with yellow arrows. (d) We update the foreground/action confidences of all supervoxels using all the NNs and their displacement vectors. (e) The supervoxel with the highest confidence is selected as $\mathbf{v}^{\tau+1}$. (f) The walk is repeated for T steps. (g) Finally, a CRF gives action proposals whose action confidences are computed using SVM.

an undirected graph $\mathbf{G}(\mathbf{V}, \mathbf{E})$ where $\mathbf{V}$ contains the supervoxels represented with spatio-temporal centroids, and $\mathbf{E}$ contains edges between neighboring supervoxels. Our goal is to find a contiguous subsets of nodes in this graph that form action proposals. We achieve this by making sequential observations based on context. Given the composite graph $\mathbf{\Xi}$, we traverse the supervoxels in testing video in a sequence, referred to as *context walk*. The sequence till step $\tau \leq T$ is given by $\mathbf{S}_\mathbf{v}^\tau = (\mathbf{v}^1, \mathbf{v}^2, \dots \mathbf{v}^\tau)$, and $\mathbf{S}_\mathbf{\Phi}^\tau = (\mathbf{\Phi}^1, \mathbf{\Phi}^2, \dots, \mathbf{\Phi}^\tau)$. Each observed supervoxel during the walk independently proposes candidate supervoxels which are later visited if they accumulate enough support during the course of the walk. Next, we describe the procedure of generating such a sequence on a given testing video.

The initial supervoxel $\mathbf{v}^1$ is selected randomly in the testing video. We find similar supervoxels from the training data and project their displacement vectors to the selected supervoxel $\mathbf{v}^\tau$ in the testing video. The following function $\psi(.)$ with the associated parameters $\mathbf{w}_\psi$ generates a conditional distribution over all the supervoxels in the testing video given only the current supervoxel $\mathbf{v}^\tau$, its features $\mathbf{\Phi}^\tau$, and the composite graph $\mathbf{\Xi}$, i.e.,

$$\psi(\mathbf{v}|\mathbf{v}^\tau, \mathbf{\Phi}^\tau, \mathbf{\Xi}; \mathbf{w}_\psi) = Z^{-1} \sum_{n=1}^{N_c} \sum_{i=1}^{I_n} \sum_{j|e_{ij}\in\mathbf{E}_n} H_\sigma(\mathbf{\Phi}^\tau, \mathbf{\Phi}_n^i; \mathbf{w}_\sigma) \cdot H_\delta(\mathbf{v}, \mathbf{v}^\tau, \mathbf{u}_n^i, \mathbf{u}_n^j; w_\delta), \quad (1)$$

where H_σ computes the similarity between features of current supervoxel in testing video $\mathbf{\Phi}^\tau$, and all the training supervoxels ($\mathbf{\Phi}_n^i$). H_δ transfers displacements between supervoxels in training videos to a supervoxel in the testing video. Both functions have weight parameters $\mathbf{w}_\sigma$ and w_δ, respectively, and Z is the normalization factor. Theoretically, Eq. 1 loops over all displacement vectors in all the training videos, and is computationally prohibitive. Therefore, we only consider the nearest neighbors for the selected supervoxel during testing using kd-trees (one per action). In Eq. 1, the function H_δ assigns a confidence to each supervoxel $\mathbf{v}$ in the testing video whether it is part of the action or not. This is achieved by computing proximity of a supervoxel in the testing video to the displacement vector projected onto the current supervoxel $\mathbf{v}^\tau$. If $\mathbf{u}_n^j - \mathbf{u}_n^i$ defines the displacement vector from the supervoxel $\mathbf{u}_n^i$ to the foreground action supervoxel $\mathbf{u}_n^j$, then H_δ is given by:

$$H_\delta(\mathbf{v}, \mathbf{v}^\tau, \mathbf{u}_n^i, \mathbf{u}_n^j; w_\delta) = \exp\Big(- w_\delta \|\mathbf{v} - \big(\mathbf{v}^\tau + \mathbf{u}_n^j - \mathbf{u}_n^i\big)\|\Big). \quad (2)$$

Furthermore, the function H_σ in Eq. 1 is simply the weighted sum of distances between the different features:

$$H_\sigma(\mathbf{\Phi}^\tau, \mathbf{\Phi}_n^i; \mathbf{w}_\sigma) = \exp\Big(- \sum_{f=1}^{F} \Big(w_{\sigma_f} \Gamma_{\sigma_f}({}_f\phi^\tau, {}_f\phi_n^i)\Big)\Big), \quad (3)$$

where Γ_{σ_f} with the associated weight parameter w_{σ_f} defines the distance function for the fth feature. For the proposed method, we used the following features: (i) ${}_1\phi = (x, y, t, s)$, i.e., centroid of the supervoxel in addition to scale (or volume) s with each dimension normalized between 0 and 1 relative to the video, (ii) appearance and motion descriptor ${}_2\phi = \mathbf{d}$ using improved Dense Trajectory Features (iDTF) [30], and (iii) the supervoxel action specificity measure, as described in §3.3.

At each step τ, we compute the non-parametric conditional distribution $\psi(.)$ in Eq. 1 and use it to update $\Psi(.)$ in the following equation, which integrates the confidences that supervoxels gather during the context walk:

$$\Psi^\tau(\mathbf{v}|\mathbf{S}_\mathbf{v}^\tau, \mathbf{S}_\mathbf{\Phi}^\tau, \mathbf{\Xi}; \mathbf{w}) = w_\alpha \psi(\mathbf{v}|\mathbf{v}^\tau, \mathbf{\Phi}^\tau, \mathbf{\Xi}; \mathbf{w}_\psi) + (1 - w_\alpha)\Psi^{\tau-1}(\mathbf{v}|\mathbf{S}_\mathbf{v}^{\tau-1}, \mathbf{S}_\mathbf{\Phi}^{\tau-1}, \mathbf{\Xi}; \mathbf{w}), \quad (4)$$

where $\mathbf{w}$ are the parameters associated with Ψ. In the above equation, the conditional distribution Ψ is updated with exponential decay at the rate w_α. Finally, the supervoxel with the highest probability from Eq. 4 is selected to be visited in the next step of the context walk:

$$\mathbf{v}^{\tau+1} = \arg\max_{\mathbf{v}} \Psi^{\tau}(\mathbf{v}|\mathbf{S}^{\tau}_{\mathbf{v}}, \mathbf{S}^{\tau}_{\mathbf{\Phi}}, \mathbf{\Xi}; \mathbf{w}). \quad (5)$$

Each video typically contains several hundred supervoxels. Although kd-tree significantly speeds up the Eq. 1, the efficiency of nearest neighbor search can be further improved using feature compression techniques [21].

3.3. Measuring Supervoxel Action Specificity

In a testing video, some supervoxels are distinct and discriminative towards one action while other supervoxels might be discriminative for other actions. We quantify this observation using a simple technique where we cluster all the descriptors (iDTF [30]) from the training videos of a particular action c into $k_c = 1 \dots K$ clusters. Our goal is to give each supervoxel an action specificity score. Let $\xi(k_c)$ represent the ratio of number of supervoxels from foreground (ground truth) of action c in cluster k_c to all the supervoxels from action c in that cluster. Then, given the appearance/motion descriptors $\mathbf{d}$, if the supervoxel belongs to cluster k_c, its action specificity $H_\chi(\mathbf{v}^i)$ is quantified as:

$$H_\chi(\mathbf{v}^i) = \xi(k_c) \cdot \exp\Big(\frac{\|\mathbf{d}^i - \mathbf{d}_{k_c}\|}{r_{k_c}}\Big), \quad (6)$$

where $\mathbf{d}_{k_c}$ and r_{k_c} are the center and radius for the kth cluster, respectively.

3.4. Inferring Action Locations using 3D-CRF

Once we have the conditional distribution $\mathbf{\Psi}^T(.)$, we merge the supervoxels belonging to actions so that resulting action proposals have contiguous supervoxels without any gaps or voids. For this, we use a Conditional Random Field where nodes form the supervoxels while edges link neighboring supervoxels. We minimize the negative log-likelihood over all supervoxel labels $\mathbf{a}$ in the video:

$$-\log\big(Pr(\mathbf{a}|\mathbf{G}, \mathbf{\Phi}, \mathbf{\Psi}^T; w_\Upsilon)\big) = \sum_{\mathbf{v}^i \in \mathbf{V}} \Big(\Theta\big(a^i|\mathbf{v}^i, \mathbf{\Psi}^T\big) + \sum_{\mathbf{v}^j | \mathbf{e}^{ij} \in \mathbf{E}} \Upsilon\big(a^i, a^j|\mathbf{v}^i, \mathbf{v}^j, \mathbf{\Phi}^i, \mathbf{\Phi}^j; w_\Upsilon\big)\Big), \quad (7)$$

where $\Theta(.)$ captures the unary potential and depends on the conditional distribution in Eq. 4 after T steps and action specificity measure computed through Eq. 6, both of which are normalized between 0 and 1:

$$\Theta\big(a^i|\mathbf{v}^i, \mathbf{\Psi}^T\big) = -\log\Big(H_\chi(\mathbf{v}^i) \cdot \mathbf{\Psi}^T(\mathbf{v}^i)\Big). \quad (8)$$

If Ω^i is the volume of the ith supervoxel, then the binary potential $\Upsilon(.)$ between neighboring supervoxels with parameter w_Υ is given by:

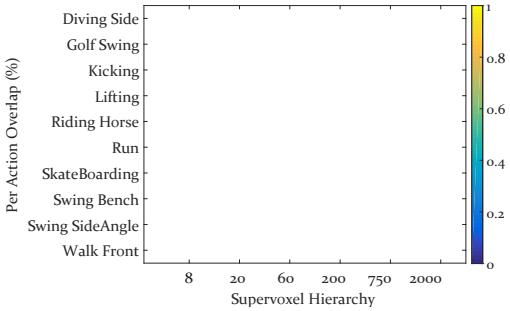

Figure 3. This figure shows the average of maximum supervoxel overlap in every training video of different actions as a function of segmentation level. Using the correct level from the hierarchy reduces the number of potential supervoxels we have to handle while testing. This speeds up the method without sacrificing performance.

$$\Upsilon\big(a^i, a^j|\mathbf{v}^i, \mathbf{v}^j, \mathbf{\Phi}^i, \mathbf{\Phi}^j; w_\Upsilon\big) = w_\Upsilon \Gamma_d(\mathbf{d}^i, \mathbf{d}^j)\Big(|\log(\Omega^i/\Omega^j)| + |\Omega^i - \Omega^j|\Big). \quad (9)$$

Once we have the actions segmented in the testing video, we use Support Vector Machine to obtain the confidence for each action segment using the appearance/motion descriptors of all supervoxels in each segment.

4. Experiments

We evaluate the proposed approach on three challenging action localization datasets: UCF-Sports [24], sub-JHMDB [15, 31] and THUMOS'13 [16]. First, we provide experimental details about the three datasets followed by detailed analysis of the performance and complexity of the proposed algorithm.

Experimental Setup: For each video in the training and testing data, we obtain a supervoxel based segmentation using [22]. This is followed by extraction of improved Dense Trajectory Features (iDTF: HOG, HOF, MBH, Traj) [30]. Every supervoxel in the video is encoded using bag-of-words (BoW) representation on iDTFs. For all our experiments, we use Top-20 nearest neighbors using kd-trees with context walk executed for $T = 5$ steps. Once we obtain segments using CRF, an SVM with histogram-intersection kernel is used to classify each action segment. We train a one-vs-all SVM per action class using ground truth bounding boxes from training videos as positive samples, while negative samples are randomly selected from the background and other action classes. Each sample is a supervoxel based BoW histogram and we consider supervoxels as positive samples only if they overlap ($\geq 80\%$) with the ground truth. Features from all the supervoxels within the ground truth are accumulated to form one representative descriptor for SVM training. Furthermore, since we used normalized features, the parameters for $\psi(.)$ did not require tuning and were set to 1, i.e., $w_\delta = w_{\sigma_1} = w_{\sigma_2} = w_{\sigma_3} = 1$. The decay

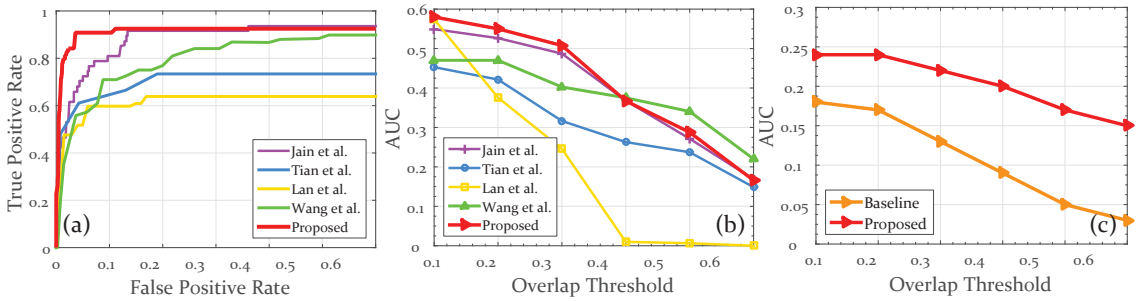

Figure 4. The ROC and AUC curves on UCF Sports Dataset [24] are shown in (a) and (b), respectively. The results are shown for Jain *et al*. [13] (orchid), Tian *et al*. [26] (blue), Lan *et al*. [18] (amber), Wang *et al*. [31] (green) and Proposed Method (red). (c) shows the AUC for THUMOS'13 dataset [16], for which we are the first to report results.

rate was set to $w_\alpha = 0.5$ and the weight for CRF was set to $w_\Upsilon = 0.1$ using training data.

Selecting Segmentation Level: Supervoxel methods [22, 13] generate different levels of a segmentation hierarchy. Each level has a different number of segmented supervoxels and may or may not cover an action. Searching for an action over the entire hierarchy is computationally inefficient and can also significantly hurt the performance of localization if an incorrect level in the hierarchy is selected. Manually choosing the correct level for a dataset is cumbersome since every action has its own complexity characterized by variation in scale, background clutter, and actor/camera movement. To automatically choose the right hierarchy level, we sample training videos from each action, and within every level of the hierarchy we find the overlap of the supervoxels with the ground truth bounding boxes. We take the maximum supervoxel overlap for each video and average it for all training videos of an action at a particular level of the segmentation hierarchy. Fig. 3 shows the average of maximum supervoxel overlap for each action at different levels of the hierarchy. The overlap peaks at a certain level and reduces thereafter. The average maximum supervoxel overlap varies for every action and selecting a unique level of segmentation for each action using this technique helps in correctly localizing an action in testing videos.

4.1. Experiments on UCF-Sports

The UCF Sports dataset [24] consists of 150 videos collected from broadcast television channels. The dataset includes 10 action classes: *diving, golf swing, kicking, etc*. Videos in the dataset are captured in a realistic setting with intra-class variations, camera motion, background clutter, scale and viewpoint changes. We follow evaluation methodology of Lan *et al*. [18] using the same train-test splits with intersection-over-union criterion at an overlap of 20%.

We construct a codebook ($K = 1000$) of iDTFs [30] using all the training videos. The quantitative comparison with state-of-the-art methods using ROC and Area Under Curve (AUC) for overlaps of $10\%, 20\%, 30\%, 40\%, 50\%$ and 60% is shown in Fig. 4(a,b). The ROC curve highlights that the proposed method performs better than the state-of-the-art methods [18, 26, 31, 13]. Although, we evaluated the classifier on very few segments of supervoxels, we are still able to achieve better results at an overlap of 20%. The comparison using AUC measure (Fig. 4(b)) also shows that we are able to achieve comparable results for different overlaps. We accredit this level of performance to avoiding background clutter and irrelevant camera motion through the use of context which allows the proposed method to ignore the potential false positive regions in the videos.

4.2. Experiments on THUMOS'13

THUMOS'13 action localization dataset was released as part of the THUMOS Challenge workshop [16] in 2013. This dataset is a subset of UCF-101 and has 3207 videos with 24 action classes such as *basketball, biking, cliff diving, etc*. The dataset is quite challenging and is currently the largest dataset for action localization. It includes several complex interactive actions such as *salsa spin, fencing, cricket bowling* with multiple action instances in the same video. We are the first to report action localization results on THUMOS'13. We also evaluated a competitive baseline using iDTFs with BoW ($K = 4000$), and trained a one-vs-all SVM-HIK for each action. Given a test video, we perform an exhaustive multi-scale spatio-temporal sub-volume search. The results are shown in Fig. 4(c).

4.3. Experiments on sub-JHMDB

The sub-JHMDB dataset [31] is a subset of the JHMDB [15] dataset where all the joints for humans in the videos have been annotated. Similar to [31], we use the box encompassing the joints as the ground truth. This dataset contains 316 clips over 12 action classes: *catch, climb stairs, golf, etc*. Jhuang *et al*. [15] have shown that this subset is far more challenging in recognizing actions compared to the entire dataset. The probable reason is the presence of the entire human body which exhibits complex variations in appearance and motion.

We used $K = 4000$ codebook centers for bag-of-words representation of the supervoxels. We report our results us-

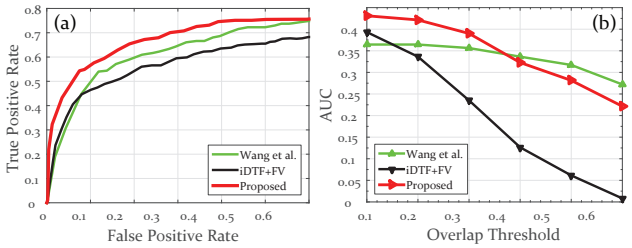

Figure 5. The ROC and AUC curves for sub-JHMDB dataset [31, 15] are shown in (a) and (b), respectively. Green and black curves are from the method by Wang *et al*. [31] and their iDTF + Fisher Vector baseline. Red curve shows the performance of the proposed method which is better than [31].

ing both ROC and AUC curves as shown in Fig. 5. At an overlap of 20%, we perform better than the state-of-the-art and achieve competitive results at other overlapping thresholds. Note that Wang *et al*. [31] also evaluated a competitive baseline over this dataset. This baseline uses iDTF features with a Fisher Vector encoding (black curves in Fig. 5) to exhaustively scan at various spatio-temporal locations at multiple scales in the video. Performing better than the baseline in a far more efficient manner emphasizes the strength of the proposed approach and reinforces that context does make a significant impact in understanding and predicting the locations of actions.

4.4. Analysis and Discussion

In Table 1, we report the percentage AUC on UCF-Sports [24] and sub-JHMDB [31] datasets. These numbers are computed at an overlap of 20% and show we perform competitively or better than existing approaches.

Computational Efficiency: Our approach achieves competitive results compared to the state-of-the-art methods on multiple datasets. However, in certain cases, some existing methods show better accuracy at higher overlaps, but this does come with a price of evaluating classifiers at a significantly higher number of locations. Note that, the BOW framework in our approach is only a matter of choice and efficiency, and results are expected to improve further through Fisher Vectors [31, 23].

Component's Contributions: The proposed approach has several aspects that contribute to its performance. We quantify their relative contributions to overall performance in Fig. 6, which shows both the ROC and AUC curves computed on UCF-Sports dataset. The grey curves represent the output using just supervoxel action specificity (§3.3). Here, we assign confidences using Eq. 6 to each supervoxel, followed by a fixed threshold. Each segment is considered as an action segment and evaluated using the ground truth. Next, we incorporate context walk as shown with green curves in Fig. 6. In this case, the confidence for supervoxels are obtained using Eq. 8. The difference between grey and

Table 1. Quantitative comparison of proposed approach with existing methods at 20% overlap.

Method	UCF-Sports	sub-JHMDB
Wang *et al*. [31]	47%	36%
Wang *et al*. (iDTF+FV) [31]	-	34%
Jain *et al*. [13]	53%	-
Tian *et al*. [26]	42%	-
Lan *et al*. [18]	38%	-
Proposed	**55%**	**42%**

red curves highlights the importance of context for action localization. Next, we show improvement in performance obtained by using CRF (Eq. 7) in blue curves, which helps in obtaining contiguous and complete action segments. Finally, the performance obtained with all aspects of the proposed approach (including SVM) is shown with red curves. The reason SVM gives a large boost is that the evaluation of action localization simultaneously quantifies action classification. Correctly localizing the action but assigning it an incorrect label is treated as incorrect localization. Since each SVM is trained on both background and negative samples from other classes, it significantly contributes to the correct classification of the localized actions. Note that for non-linear kernels, the summation of scores from supervoxels does not equal that of an action volume, thus, necessitating classification using an SVM. Nevertheless, this is an inexpensive step since we require very few SVM evaluations.

Action Contours: The proposed approach uses over-segmented supervoxels, therefore, it produces action segments which can be used for video segmentation as well. Since the local features (iDTF) are based on motion, the segments are heavily dependent on the motion of actors. Such results are an improvement over cuboid representation which can contain significant quantities of background. Some qualitative results of the proposed approach with segmented actors are presented in Fig. 7. Since the proposed method uses supervoxels to segment the video, we are able to capture the entire human body contours after CRF. These results show that supervoxels indeed help in obtaining fine contours while reducing the complexity of the problem. However, there are certain cases where the proposed ap-

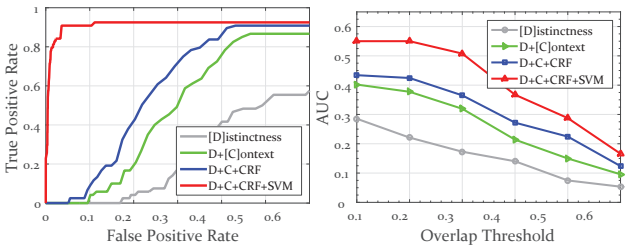

Figure 6. This figure shows the contributions of the fours aspects of the proposed approach towards overall performance, in terms of ROC (left) and AUC of Precision-Recall curve as a function of overlap threshold (right).

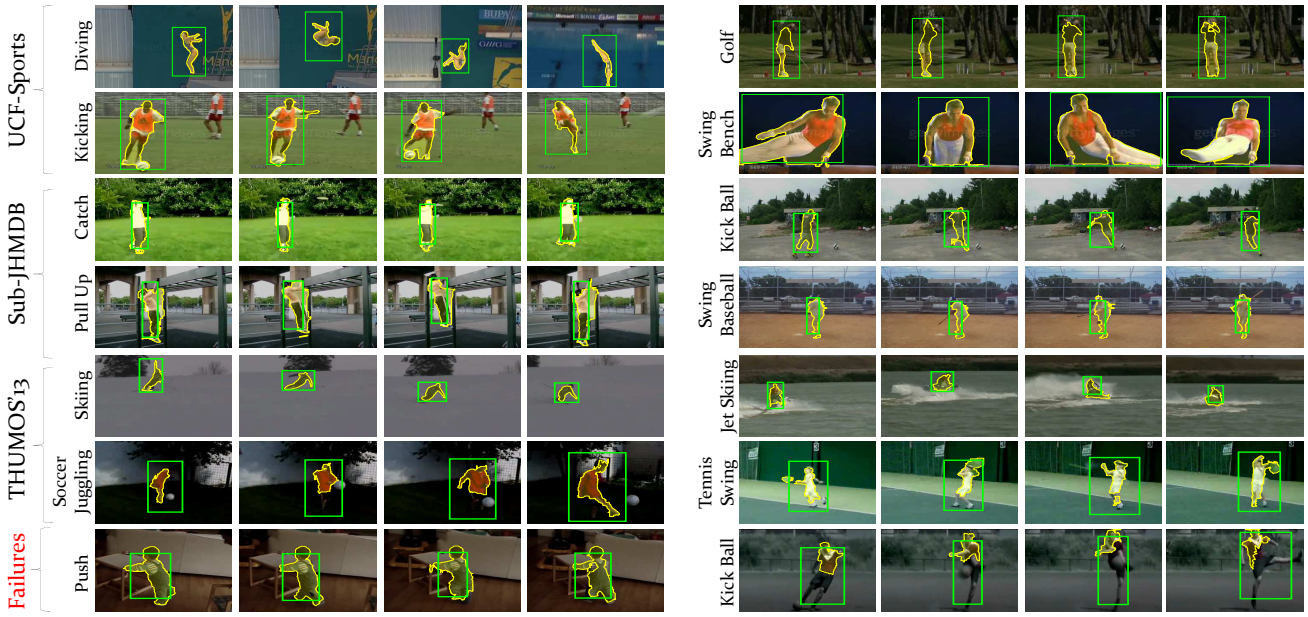

Figure 7. This figure shows qualitative results of the proposed approach (yellow contours) against ground truth (green boxes) on selected frames of testing videos. The first two rows are from UCF-Sports [24], third and fourth are from sub-JHMDB [31], while fifth and sixth rows are from THUMOS'13 [16] datasets. Last row shows two failure cases from sub-JHMDB.

proach fails as shown in the last row of Fig. 7. The action depicted on the left of the last row shows the case where the action *push* was classified as *walk*, even though it was localized correctly. The second set of images on the right shows incorrect localization of the action *kick-ball*. For this particular case, the large motion of the actor resulted in a large number of supervoxels on the lower body as compared to training videos. Many supervoxels had different distances (from Eq. 2) as compared to the ones seen during training. This caused lower confidences for such supervoxels resulting in only upper-body localization.

Complexity Analysis: We offer an analysis of the number of classifier evaluations of the proposed approach on the number of supervoxels or subvolumes with two other state-of-the-art methods. Table 2 shows Tian *et al*. [26] who learn a Spatio-temporal Deformable Parts Model detector that is used to search for an action over width (X), height (Y), time (T) and different aspect ratios (S) within the video. This requires enormous computations which can incur many false positives as well. We also compare the effectiveness of the proposed approach to Jain *et al*. [13], who also use supervoxels to reduce computation. Given N supervoxels at the lowest level, they apply an agglomerative hierarchical clustering, which merges supervoxels at each level of the hierarchy followed by an application of SVM classifier on each supervoxel. Compared to these approaches we localize in constant time (context-walk with 5 steps and one inference through CRF followed by an execution of SVM). Note that this table only shows the complexity of localizing the action, assuming the features have been computed and models have been learnt in advance.

Table 2. Number of classifier evaluations as a function of supervoxels / subvolumes in a video.

Method	Evaluated Volumes	Complexity
SDPM [26]	*XYTS*	$\mathcal{O}(n^4)$
Action Tubelets [13]	*N + (N-1) + ... + 1*	$\mathcal{O}(n^2)$
Proposed	5 (+ CRF)	$\mathcal{O}(c)$

5. Conclusion

We presented an efficient and effective approach to localize actions in videos. We use context to make a series of observations on supervoxels, such that the probability of predicting the location of an action increases at each step. Starting with a random supervoxel, we find similar supervoxels from the training data, and transfer the knowledge about relative spatio-temporal location of an action to the test video. This gives a conditional distribution over the graph formed by supervoxels in the testing video. After selecting the supervoxel with highest probability, we repeat the steps. The conditional distribution at the end of Context Walk over supervoxel graph is used in a CRF to infer the number and location of action proposals. Finally, each of the proposals is evaluated through an SVM. Due to both supervoxels and context, the proposed approach requires very few classifier evaluations. The future work will aim at action localization in longer videos, which will need multiple random initializations and increased number of steps for the context walk.

References

[1] Trecvid multimedia event recounting evaluation track. http://www.nist.gov/itl/iad/mig/mer.cfm.

[2] J. K. Aggarwal and M. S. Ryoo. Human activity analysis: A review. *ACM Computing Surveys (CSUR)*, 43(3), 2011.

[3] B. Alexe, N. Hees, Y. Teh, and V. Ferrari. Searching for objects driven by context. In *NIPS*, 2012.

[4] W. Chen, C. Xiong, R. Xu, and J. J. Corso. Actionness ranking with lattice conditional ordinal random fields. In *CVPR*, 2014.

[5] M. J. Choi, J. J. Lim, A. Torralba, and A. S. Willsky. Exploiting hierarchical context on a large database of object categories. In *CVPR*, 2010.

[6] C. Desai and D. Ramanan. Detecting actions, poses, and objects with relational phraselets. In *ECCV*. 2012.

[7] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE TPAMI*, 32(9), 2010.

[8] A. Gupta and L. S. Davis. Objects in action: An approach for combining action understanding and object perception. In *CVPR*, 2007.

[9] D. Han, L. Bo, and C. Sminchisescu. Selection and context for action recognition. In *ICCV*, 2009.

[10] G. Heitz and D. Koller. Learning spatial context: Using stuff to find things. In *ECCV*. 2008.

[11] Y. Hu, L. Cao, F. Lv, S. Yan, Y. Gong, and T. S. Huang. Action detection in complex scenes with spatial and temporal ambiguities. In *ICCV*, 2009.

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

[13] M. Jain, J. Gemert, H. Jegou, P. Bouthemy, and C. Snoek. Action localization with tubelets from motion. In *CVPR*, 2014.

[14] M. Jain, J. C. van Gemert, and C. G. Snoek. What do 15,000 object categories tell us about classifying and localizing actions? In *CVPR*, 2015.

[15] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. J. Black. Towards understanding action recognition. In *ICCV*, 2013.

[16] Y.-G. Jiang, J. Liu, A. Roshan Zamir, I. Laptev, M. Piccardi, M. Shah, and R. Sukthankar. THUMOS challenge: Action recognition with a large number of classes. http://crcv.ucf.edu/ICCV13-Action-Workshop/, 2013.

[17] Y. Ke, R. Sukthankar, and M. Hebert. Event detection in crowded videos. In *ICCV*, 2007.

[18] T. Lan, Y. Wang, and G. Mori. Discriminative figure-centric models for joint action localization and recognition. In *ICCV*, 2011.

[19] S. Ma, J. Zhang, N. Ikizler-Cinbis, and S. Sclaroff. Action recognition and localization by hierarchical space-time segments. In *ICCV*, 2013.

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

[21] M. Norouzi, A. Punjani, and D. J. Fleet. Fast search in hamming space with multi-index hashing. In *CVPR*, 2012.

[22] D. Oneata, J. Revaud, J. Verbeek, and C. Schmid. Spatio-temporal object detection proposals. In *ECCV*. 2014.

[23] D. Oneata, J. Verbeek, and C. Schmid. Efficient action localization with approximately normalized fisher vectors. In *CVPR*, 2014.

[24] M. Rodriguez, A. Javed, and M. Shah. Action mach: a spatio-temporal maximum average correlation height filter for action recognition. In *CVPR*, 2008.

[25] J. Sun, X. Wu, S. Yan, L.-F. Cheong, T.-S. Chua, and J. Li. Hierarchical spatio-temporal context modeling for action recognition. In *CVPR*, 2009.

[26] Y. Tian, R. Sukthankar, and M. Shah. Spatiotemporal deformable part models for action detection. In *CVPR*, 2013.

[27] D. Tran and J. Yuan. Max-margin structured output regression for spatio-temporal action localization. In *NIPS*, 2012.

[28] D. Tran, J. Yuan, and D. Forsyth. Video event detection: From subvolume localization to spatiotemporal path search. *IEEE TPAMI*, 36(2), 2014.

[29] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders. Selective search for object recognition. *IJCV*, 104(2), 2013.

[30] H. Wang and C. Schmid. Action recognition with improved trajectories. In *ICCV*, 2013.

[31] L. Wang, Y. Qiao, and X. Tang. Video action detection with relational dynamic-poselets. In *ECCV*. 2014.

[32] T. Wang, S. Wang, and X. Ding. Detecting human action as the spatio-temporal tube of maximum mutual information. *IEEE TCSVT*, 24(2), 2014.

[33] D. Weinland, R. Ronfard, and E. Boyer. A survey of vision-based methods for action representation, segmentation and recognition. *CVIU*, 115(2), 2011.

[34] X. Wu, D. Xu, L. Duan, and J. Luo. Action recognition using context and appearance distribution features. In *CVPR*, 2011.

[35] Y. Xie, H. Chang, Z. Li, L. Liang, X. Chen, and D. Zhao. A unified framework for locating and recognizing human actions. In *CVPR*, 2011.

[36] G. Yu, N. A. Goussies, J. Yuan, and Z. Liu. Fast action detection via discriminative random forest voting and top-k subvolume search. *Multimedia, IEEE Transactions on*, 13(3), 2011.

[37] G. Yu and J. Yuan. Fast action proposals for human action detection and search. In *CVPR*, 2015.

[38] J. Yuan, Z. Liu, and Y. Wu. Discriminative video pattern search for efficient action detection. *IEEE TPAMI*, 33(9), 2011.

[39] Z. Zhang, Y. Hu, S. Chan, and L.-T. Chia. Motion context: A new representation for human action recognition. In *ECCV*. 2008.

[40] Z. Zhou, F. Shi, and W. Wu. Learning spatial and temporal extents of human actions for action detection. *Multimedia, IEEE Transactions on*, 17(4), 2015.