This CVPR2015 paper is the Open Access version, provided by the Computer Vision Foundation.
The authoritative version of this paper is available in IEEE Xplore.

Fast Action Proposals for Human Action Detection and Search

Gang Yu, Junsong Yuan
School of Electrical and Electronic Engineering
Nanyang Technological University, Singapore
iskicy@gmail.com, jsyuan@ntu.edu.sg

Abstract

In this paper we target at generating generic action proposals in unconstrained videos. Each action proposal corresponds to a temporal series of spatial bounding boxes, i.e., a spatio-temporal video tube, which has a good potential to locate one human action. Assuming each action is performed by a human with meaningful motion, both appearance and motion cues are utilized to measure the actionness of the video tubes. After picking those spatiotemporal paths of high actionness scores, our action proposal generation is formulated as a maximum set coverage problem, where greedy search is performed to select a set of action proposals that can maximize the overall actionness score. Compared with existing action proposal approaches, our action proposals do not rely on video segmentation and can be generated in nearly real-time. Experimental results on two challenging datasets, MSRII and UCF 101, validate the superior performance of our action proposals as well as competitive results on action detection and search.

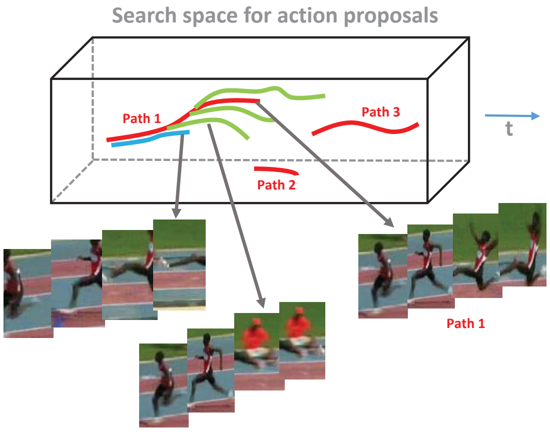

Figure 1. An illustration of action proposals. The red paths in the upper figure represent three detected action proposals, where each action proposal corresponds to a series of bounding boxes in the video space. The green and blue paths, which have large spatial-temporal overlap with the red paths, should be removed for the path diversity.

1. Introduction

Motivated by fast object detection and recognition using object proposals [8, 31, 33], we present an approach to efficiently propose action candidates of generic type in unconstrained videos. Each proposed action candidate corresponds to a temporal series of spatial bounding boxes, i.e., a spatio-temporal video tube, which locates the potential action in the video. For many video analytics tasks, e.g., action detection [21, 17, 25] and action search [26], we argue that a quick generation of action proposals is of great importance, because sophisticated action recognition can focus on the action proposals rather than the whole video to save computational cost and improve the performance, similar to the benefits of using object proposals for object detection and recognition.

Despite the success of object proposals, generating action proposals in videos is however a more challenging problem due to two reasons. First, different from objectness measure that relies on visual appearance only, action proposals need to take both appearance and motion cues into consideration. For example, actions should be coupled with human with meaningful motion. However, due to the diversity and variations of human actions, it is difficult to learn the actionness measure that can well differentiate human actions from the background clutters and other dynamic motions, which are quite common in unconstrained videos. Second, the candidate number of action proposals can be much larger than that of the object proposals. Given a video of size $M \times N \times T$, even with the fixed size bounding box, the candidate number of action proposals can be as large as $O(MNTk^T)$ [30], where k is the number of spatial neighbors a bounding box will consider to link in the next frame, which controls the smoothness of the action proposal tube. As the spatial extent of the action can vary across frames, if we consider a flexible bounding box size, it becomes an even much larger size of $O(M^2N^2Tk^T)$. As a result, it is computationally infeasible to explore the full candidate set to pick action proposals.

To address the above two challenges when generating action proposals, we first perform human and motion detection to generate candidate bounding boxes that may cover the human action in each frame. After picking up the bounding boxes of high "actionness" scores, we utilize the max sub-path search algorithm to locate the top-N maximal spatio-temporal paths based on "actionness" score. Due to the spatio-temporal redundancy in the video, many high quality paths may largely overlap with each other as the example shown in Fig. 1. The red paths illustrate three detected action proposals. But the green paths and blue path which significantly overlap with path 1 are redundant paths and should be removed. To pick the action proposals, we further formulate it as a maximum set coverage problem where each candidate path corresponds to a set, with each bounding box as an element. Such a maximum set coverage problem is NP-hard, but a greedy search algorithm [14] can achieve an approximation ratio of $1 - \frac{1}{e}$.

To evaluate the performance of our action proposals, we test two benchmark datasets, MSR II [4] and UCF 101 [5]. We notice that a small number of action proposals, e.g., 2000 proposals for all the 54 video clips in MSRII dataset, can already provide promising recall rate. Also, based on our action proposals, we can obtain competitive action detection and action search results in MSRII dataset compared with existing results. Moreover, the result on UCF 101 dataset validates that our action proposal can well track the actions in unconstrained videos. Last but not the least, compared with existing action proposal approaches, our action proposals do not rely on video segmentation and can be generated in nearly real-time on a normal desktop PC.

2. Related Work

Object proposals [8, 16, 31] have been actively studied recently. By generating a set of potential object bounding boxes with efficient computational speed and high recall, it can be utilized to replace the time-consuming sliding window approach for object detection so that sophisticated image feature and model can be evaluated [33]. A recent review of object proposal can be found from [32].

Compared with object proposals, action proposal is not sufficiently exploited in the video space, where action localization is a much more computational intensive step compared with object detection in the image case. Traditionally, action localization is handled by a sliding window based approach. For example, in [11], a weakly supervised model based on multiple instance learning is proposed to slide the spatial-temporal subvolumes for action detection. Spatio-temporal branch-and-bound algorithm is employed in [4] to reduce the computational cost. Other sliding-window based action detection approaches include [12, 22, 23]. Despite their successes, one limitation with those sliding-window based approach is that the detected action is usually captured by a video sub-volume, thus cannot handle the moving actions. Besides from sub-volume detection, spatial-temporal action tubes can be detected via structured output regression [13] but with a computationally intensive optimization. Although a linear complexity search algorithm has been proposed in [30], it only searches for the best video tube with bounding box size fixed. In [35], a near-optimal algorithm with linear complexity is presented for object tracking. In addition, Hough voting based approach can be applied for action localization with reasonable performance as in [40, 36].

Although there are a few possible solutions for action localization, the extremely large search space leads to intensive computational cost. Action proposal would be a good alternative to significantly reduce the search space. In [27], action proposals are generated by hierarchically merging super-voxels. Similarly, segmentation based proposal are generated in [20, 3]. However, these action proposal approaches highly rely on relatively accurate video segmentation [37, 38], which itself is a challenging problem. Moreover, it is difficult to efficiently and accurately segment the human action from the clutter video sequences. In [34], "actionness" is measured based on lattice conditional ordinal random fields. However, it does not address the action localization problem. In this paper, we present to formulate the action proposal based on the set coverage problem. In [41], maximum weight independent set is presented to solve the data association in multiple object tracking. Similar idea is applied to object segmentation in [39, 42].

Our work is also relevant to previous work [9, 10] that utilize human detector for human tracking and action localization. For example, [9] combines features of object, scene, and action for action recognition while [10] presents a deformable part model for human pose representation and action detection. A united framework based Hough forest is presented for human detection, tracking, and action recognition in [15]. Different from these human detection and tracking based algorithms, our focus is to generate generic action proposals which focus on the localization of potential actions by integrating the human and motion cues.

3. Action Proposals

Given a video sequence, our goal is to generate a number of action proposals $\mathbf{P} = \{\mathbf{p}^{(1)}, \mathbf{p}^{(2)}, \cdots, \mathbf{p}^{(K)}\}$, where each action proposal $\mathbf{p}^{(i)} = \{b_{t_s}^{(i)}, b_{t_s+1}^{(i)}, \cdots, b_{t_e}^{(i)}\}$ corresponds to a path from the t_s-th frame to t_e-th frame. Each element b_t in the path refers to a bounding box $[x, y, m, n]$, where (x, y) is the center, m is the width, and n is the height of the bounding box. As a smooth spatio-temporal path that may follow the actor, each action proposal should satisfy the following two requirements:

$$O(b_t^{(i)}, b_{t+1}^{(i)}) \geq \delta_o, \quad \forall b_t^{(i)} \in \mathbf{p}^{(i)} \tag{1}$$

$$||C(b_t^{(i)}) - C(b_{t+1}^{(i)})|| \leq \delta_c, \quad \forall b_t^{(i)} \in \mathbf{p}^{(i)}. \tag{2}$$

The first constraint in Eq. 1 requires the action proposal to be a smooth path, i.e., the intersection-over-union (IOU) of two consecutive bounding boxes $O(b_t^{(i)}, b_{t+1}^{(i)}) = \frac{\cap(b_t^{(i)}, b_{t+1}^{(i)})}{\cup(b_t^{(i)}, b_{t+1}^{(i)})}$ is large enough. The second constraint in Eq. 2 is to require the action proposal to correspond to a path of consistent appearance, thus it is more likely to track the same actor. In our implementation, $C(b)$ represents the color histogram of the bounding box b and δ_c is a threshold.

Each bounding box b_t will be associated with a discriminative actionness score $w(b_t)$, which will be explained in Section 3.4. The actionness score of a path is the summation of the actionness scores from all of its bounding boxes. It is worth noting as the actionness score can be either negative or positive. Thus it is not necessary a longer path will have a higher actionness score. To find a good set of action proposals $\mathbf{P}$, we prefer them to maximize the coverage of actionnness scores in the video space $\max_{\mathbf{P} \subset \mathcal{S}} \sum_{b_t \in \cup \mathbf{p}^{(i)}} w(b_t)$, where $\mathcal{S}$ is a collection of proposal candidates that satisfy the constraints in Eq. 1 and Eq. 2. In the next subsection, we will formulate it as a maximum set coverage problem [14] where each path $\mathbf{p}^{(i)}$ can be considered as a set with the bounding box $b^{(i)}$ as its element.

3.1. Problem Formulation

Formally, we want to find the path set $\mathbf{P}$ which maximizes the following set coverage problem:

$$\max_{\mathbf{P} \subset \mathcal{S}} \sum_{b_t \in \cup \mathbf{p}^{(i)}} w(b_t) \tag{3}$$

$$\text{s.t.} \quad |\mathbf{P}| \leq K, \tag{4}$$

$$\mathbf{O}(\mathbf{p}^{(i)}, \mathbf{p}^{(j)}) \leq \delta_p, \quad \forall \mathbf{p}^{(i)}, \mathbf{p}^{(j)} \in \mathbf{P}, i \neq j. \tag{5}$$

The first constraint (Eq. 4) is to set the maximum number of action proposals as K while the second constraint (Eq. 5) is to avoid generating redundant action proposals that are highly overlapped. The overlap of two paths is defined as

$$\mathbf{O}(\mathbf{p}_i, \mathbf{p}_j) = \frac{\sum_{\max(t_s^{(i)}, t_s^{(j)}) \leq t \leq \min(t_e^{(i)}, t_e^{(j)})} \cap(b_t^{(i)}, b_t^{(j)})}{\sum_{\max(t_s^{(i)}, t_s^{(j)}) \leq t \leq \min(t_e^{(i)}, t_e^{(j)})} \cup(b_t^{(i)}, b_t^{(j)})}, \tag{6}$$

where δ_p is a threshold.

3.2. Top-N Path Candidate Search

To solve the maximum set coverage problem in Eq. 3, we need to obtain the proposal candidate set $\mathcal{S}$ first. However, the search space for $\mathcal{S}$ in Eq. 3 is extremely large and increases exponentially along with the video duration. Thus, it is impossible to enumerate all the paths to obtain $\mathcal{S}$.

To address the computational issue, we present a max sub-path based candidate search algorithm [30] to collect the top-N path candidates as $\mathcal{S}$ which satisfy the constraints in Eq. 1 and Eq. 2. One solution is to apply the max sub-path search algorithm [30] N times with a non-maximum suppression each time to avoid those similar paths. In this paper, we propose a novel Top-N max path search algorithm which can locate the top-N max paths, in one round of search. There are two steps: forward search to locate the end of path and back-trace to recover the whole path.

The idea is to maintain a pool of the best N paths, denoting as $\mathcal{S} = \{(f_k, b^{(k)}), \ \ k = 1, 2, \cdots N\}$, where f_k is the actionness score of the k-th best path so far and $b^{(k)}$ records the end position of the corresponding path. Meantime, we keep the best score so far for each bounding box $f(b_t^i)$ during a forward search process:

$$f(b_t^i) = \max_{b_{t-1}^j} \{f(b_{t-1}^j) + w(b_t^i), 0\}, \tag{7}$$

where b_{t-1}^j and b_t^j should satisfy Eq. 1 and Eq. 2. If the accumulated score for the current bounding box $f(b_t^i)$ is larger than the N-th path score in the path pool $f(b_t^i) > f_N$, then the N-th path will be replaced by the new path with score $f(b_t^i)$ and ending at the position b_t^i (in the implementation, the bounding box at the previous frame b_{t-1}^j which leads to b_t^i should be saved as well). To facilitate the replacement, a min-heap structure can be utilized to maintain the best N paths based on the path scores. After the forward search, a back-tracing step can be performed to locate the whole path $\mathbf{p}_k$ for each candidate in the pool. More specifically, for each path $\mathbf{p}$, we can trace from the end of the path t_e back to the start of the path t_s by finding the corresponding b_t^* for each frame $t_s \leq t \leq t_e$ which satisfies:

$$f(b^*) = \sum_{t_s \leq t \leq t_e} w(b_t^*). \tag{8}$$

3.3. Greedy Search

Based on the path candidates $\mathcal{S}$, now we can solve the problem in Eq. 3 to obtain the action proposals. According to [14], the maximum set coverage problem is NP-hard but a greedy algorithm can achieve an approximation ratio of $1 - \frac{1}{e}$. Following [14], a greedy-based solution is presented to address the optimization problem in Eq. 3.

Initially, we add the path candidate $\mathbf{p}_1$ with the largest actionness score f_1 to the action proposal set $\mathbf{P}$. Suppose $k - 1$ action proposals have already been found: $\mathbf{P} = \{\mathbf{p}^{(1)}, \mathbf{p}^{(2)}, \cdots, \mathbf{p}^{(k-1)}\}$. To search the k-th action proposal, we enumerate the rest paths from the path candidates and

Algorithm 1 Action Proposal

```
Input: bounding box score $w(b_t^i)$,
Output: action candidates $\mathbf{P} = \{\mathbf{p}^{(1)}, \mathbf{p}^{(2)}, \cdots, \mathbf{p}^{(K)}\}$
 1: $f_k = 0, b^{(k)} = \emptyset, \quad k = 1, 2, \cdots N$
 2: for $t = 1 \to T$ do
 3:     for $i = 1 \to N_b^t$ do
 4:         $f(b_t^i) = \max_{b_{t-1}^j}\{f(b_{t-1}^j) + w(b_t^i), 0\}$ as Eq. 7
 5:         if $f(b_t^i) > f_N$ then
 6:             $f_N = f(b_t^i), \quad b^{(N)} = b_t^i$
 7:         end if
 8:     end for
 9: end for
10: back trace to obtain $\mathbf{p}_i, i = 1, \cdots, N$
11: $k = 1, \quad \mathbf{P} = \emptyset$
12: repeat
13:     $\arg\max_i \sum_{b \in \mathbf{p}_i \bigcup \mathbf{p}^{(1)} \bigcup \cdots \bigcup \mathbf{p}^{(k-1)}} w(b)$
14:     if $\mathbf{p}_i$ satisfies Eq. 5 with $\mathbf{p}^{(j)}, j = 1, \cdots, k-1$ then
15:         $\mathbf{p}^{(k)} = \mathbf{p}_i, \quad \mathbf{P} = \{\mathbf{p}^{(1)}, \mathbf{p}^{(2)}, \cdots, \mathbf{p}^{(k)}\}$
16:         $k = k + 1$
17:     end if
18: until $k > K$
```

select the one $\mathbf{p}_i$ which can maximize

$$\arg\max_i \sum_{b \in \mathbf{p}_i \bigcup \mathbf{p}^{(1)} \bigcup \cdots \bigcup \mathbf{p}^{(k-1)}} w(b). \tag{9}$$

This objective function can successfully suppress the green paths in Fig. 1 but cannot eliminate the blue path which also largely overlaps with the selected path in **P**. To reduce the redundancy among the action proposals, the newly added path should also satisfy the constraint in Eq. 5.

An illustration of the proposed algorithm can be found in Algorithm 1. The input is the score $w(b_t^i)$ for each bounding box in the video, where b_t^i refers to the i-th bounding box in the t-th frame. The total number of bounding box in t-th frame is N_b^t. The first 9 lines illustrate the forward search process and a min-heap data structure is maintained to save the best N scores. Each time when the actionness score of the new bounding box is larger than the N-th best score, i.e., $f(b_t^i) > f_N$, the path with N-th best score is replaced with the new path ending at b_t^i. At the 10-th line, a back tracing step as in Eq. 8 is performed to locate the full paths for the best N scores. Finally, at the line from 11-18, the greedy-based search is performed to obtain the best K action proposals which satisfy the constraints in Eq. 3.

Another good property of our algorithm is that it can be applied for online processing. For example, as our algorithm naturally generates the action proposals in an online manner, our proposals can be further used for online applications like online action detection in video surveillance environment.

3.4. Actionness Score of Bounding Box

In this section we explain how to obtain the score of the bounding box $w(b)$ in the path. As the computational cost is an important concern for action proposals, we propose an efficient approach to compute the bounding box actionness score based on appearance and motion information. Other more advanced actionness measures like [34] can be employed in our framework as well but with more intensive computational cost.

We define the actionness of a bounding box b based on two parts: human detection score $H(b)$ and motion score $M(b)$:

$$w(b) = H(b) + \lambda M(b), \tag{10}$$

where λ is a parameter which balances the human detection score and motion score. Normally, an action should be performed by a human being and therefore human score is critical for the action proposal. Efficient human detector, e.g. [28, 29, 19] can be applied to obtain the human score $H(b)$, where $H(b) > 0$ means the bounding box is classified as a positive human region.

However, human detection score alone is not sufficient to determine an action instance since the human actor should perform meaningful motion to make it an action. Thus, besides the human score, we propose to add motion score that accounts for the motion pattern of generic actions.

Dense trajectory features [1] are extracted and map to each detected bounding box. For each trajectory, we can determine its motion based on the variation of the trajectory position. If the trajectory does not make any movement, it will be removed. One direct motion score is by counting the number of moving trajectories in the bounding box, which indicate high motion energy. On the other hand, compared with random motion, human action should have specific motion pattern. Thus, we propose to learn the motion score by matching the dense trajectory with a set of training actions. Suppose we have a set of positive actions $\mathcal{P} = \{d_{\mathcal{P}_1}, d_{\mathcal{P}_2}, \cdots, d_{\mathcal{P}_N}\}$ (can contain actions from multiple categories) and a set of negative actions $\mathcal{N} = \{d_{\mathcal{N}_1}, d_{\mathcal{N}_2}, \cdots, d_{\mathcal{N}_N}\}$ (optional), the matching score for each local interest point d_i can be computed as:

$$s(d_i) = D(d_i, \mathcal{N}) - D(d_i, \mathcal{P}), \tag{11}$$

where $D(d_i, \mathcal{N})$ is the average distance of d_i and the top-10 nearest points in $\mathcal{N}$ based on the descriptors of the trajectory. Similarly, $D(d_i, \mathcal{P})$ is the average distance of d_i against the top-10 nearest points in the positive action sequence set $\mathcal{P}$. The semantic score for the action candidate is defined as:

$$M(b) = \frac{\sum_{d_i \in b} s(d_i)}{A(b)}, \tag{12}$$

where $A(b)$ is the area of the bounding box b.

3.5. Real-time Implementation

Fast computational speed is a necessary requirement for the action proposal algorithms. In this subsection, we will provide the implementation details on how to speed up the algorithm. Let us begin with the cost from human detection. Fast human detection, e.g., [28, 29, 6, 19], is first utilized to obtain the human score for each bounding box and a non-maximum suppression is applied to eliminate the bounding boxes which are highly overlapped. Based on the remaining bounding boxes, a threshold (e.g., fixed at 0) is set to filter those detections below the threshold. This can significantly reduce the number of bounding boxes for each frame. As the human detector is applied on each frame individually, the temporal consistence may be ignored. To enable the smoothness of the human action, the detections on t-th frame will be mapped to the following frames with a decayed human score, until it reaches 0. For the motion score $M(b)$, instead of performing local interest point matching for all the dense trajectories, the motion score defined in Section 3.4 is computed only within those bounding boxes detected in each frame. This can significantly simplify the model structure for efficient action proposal. Then the top-N path candidate search discussed in Section 3.2 is applied. In order to avoid highly similar paths in the candidate set $\mathcal{S}$, when the score of a new path candidate is larger than the N-th path (Line 5 in Algorithm 1), an efficient linear comparison based on the last bounding box $b^i_{t_e-1}$ is evaluated to replace the highly overlapped path. Then a greedy search algorithm discussed in Section 3.3 will be performed to spatial-temporally locate the actions proposals.

On a workstation with Intel Xeon E5-2609 CPU and 64 GB memory, the computational time of generation of 2000 action proposals can be less than 20 seconds from the 30-min MSRII dataset, excluding the computing human detection and motion scores. With fast human detector [28, 29], the human score $H(b)$ can be efficiently computed at 30 fps. For the motion score $M(b)$, the cost for dense trajectory extracting as well as the trajectory matching can be operated at 15 fps.

4. Applications

The generated action proposals can be applied to two important applications on human action understanding: action detection and action search. Since the action proposals have already been spatio-temporally localized in the video sequences, it avoids the time-consuming sliding-window evaluation in the video space as in [4, 7].

4.1. Action Detection

For action detection on specific action category, a set of labeled training videos are required to train the supervised model. Dense trajectory features are extracted from each video sequence and the descriptors (e.g., HoG, HoF, trajectory, MBH) are utilized for each local point [1]. Fisher vector representation [18] is applied for these local descriptors respectively. Then a linear SVM is trained on the concatenated fisher vector representation. During the testing stage, for each action proposal in $\mathbf{p}_i \in \mathbf{P}$, dense trajectory feature will be encoded with fisher vector and the action recognition response is computed based on the trained linear SVM. Power normalization and L_2 normalization are employed for fisher vector as in [2].

4.2. Action Search

Different from action detection, action search [26] tries to locate the action instances in large-scale video database which are similar to a query action. During the offline stage, action proposal algorithm is applied to the video database and a set of action proposals can be located. For each action proposal, dense trajectory feature as well as the descriptors are extracted and bag-of-visual-word (BoW) is applied to cluster the local trajectories. We denote $\mathbf{f}_i$ as the feature representation (normalized histogram of BoW) for the action instance $\mathbf{p}_i$. During the online search stage, a query video $\mathcal{V}_q$ is provided and BoW action representation based on dense trajectory is extracted as $\mathbf{f}_q$. The similarity between the query video $\mathbf{f}_q$ and database action proposal $\mathbf{f}_i$ can be evaluated based on histogram intersection:

$$\mathbf{s}(\mathbf{f}_q, \mathbf{f}_i) = \sum_k \min(\mathbf{f}_q^{(k)}, \mathbf{f}_i^{(k)}), \tag{13}$$

where $\mathbf{f}^{(k)}$ refers to the k_{th} dimension of $\mathbf{f}$. As the action proposal algorithm is performed offline, our algorithm can significantly reduce the online cost of action search compared with [26].

5. Experiments

To evaluate the performance of our action proposals and its application on action detection and search, MSRII [4] and UCF 101 [5] datasets are employed.

5.1. Experimental Results on MSRII

For MSRII dataset [4], there are 54 long video sequences where each video consists of several actions performed by different people in a crowded environment. The videos contain three categories of actions: handwaving, handclapping and boxing. Following the same experimental setting in [4], cross-dataset evaluation is employed, where KTH dataset [24] is used for training while the testing step is performed on the MSRII dataset.

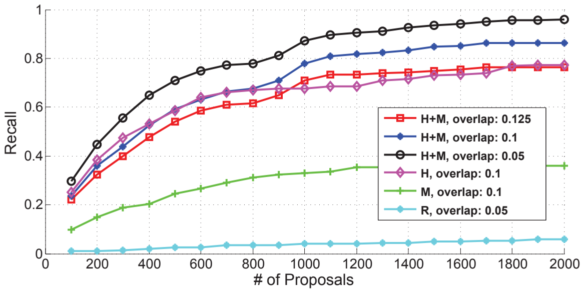

Figure 2. Recall for action proposal in MSRII dataset.

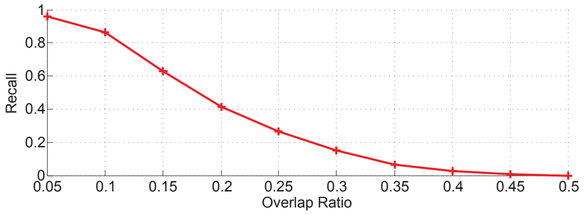

Figure 3. Evaluation of the relationship between the recall and overlap ratio θ.

Method	H+M	H	M	R @ 2K	R @ 1M
Recall	0.862	0.773	0.360	0.000	0.015
AP	0.450	0.409	0.150	0.008	0.061

Table 1. Evaluation of the recall (IOU with $\theta = 0.1$) and average precision (AP) on MSRII dataset. In total 2000 proposals are generated for our algorithm.

We first evaluate the performance of our action proposals based on the recall. We consider a hit of ground-truth action $\mathbf{G}$ if the intersection-over-union (IOU) $\mathbf{O}(\mathbf{p}^*, \mathbf{G}) > \theta$, where the overlap function $\mathbf{O}$ is defined in Eq. 6 and $\mathbf{p}^*$ is one action proposal, and θ is the overlap threshold. To collect the training set for the motion score discussed in Section 3.4, video clips with the categories of handwaving, handclapping and boxing from KTH dataset are combined as the positive videos $\mathcal{P}$ while walking clips are used as negative videos $\mathcal{N}$. Fig. 2 illustrates the recall of 203 ground-truth human actions from MSRII dataset with different overlap ratio θ. In the legend of Fig. 2, "H" is the human score, "M" is the motion score and "R" is the random generation algorithm. We can find that our discovered small set of action proposals can cover most of the ground-truth actions. In addition, Fig. 3 shows the relationship between the recall and overlap ratio of our proposed algorithm ("H+M").

Also, based on the 2000 action proposals generated for all the videos (except for random approach), we can rank them based on actionness score in Eq. 8. Average precision is used to evaluate our actionness score by using the 203 actions in MSRII as the positive actions. Following [4], for the computation of the precision we consider a true detection if : $\frac{\text{Volume}(\mathbf{p}^* \cap \mathbf{G})}{\text{Volume}(\mathbf{p}^*)} > \frac{1}{8}$, where $\mathbf{G}$ is the annotated ground truth subvolume, and $\mathbf{p}^*$ is one detected action candidate. On the other side, for the computation of the recall we consider a hit if: $\frac{\text{Volume}(\mathbf{p}^* \cap \mathbf{G})}{\text{Volume}(\mathbf{G})} > \frac{1}{8}$. Notice that the recall calculation for average precision is different from the one used for Fig. 2 and Fig. 3. Table 1 shows that our action proposal algorithm can successfully reduce the number of action candidates and provide an effective ranking of the candidates based on our score function. According to our evaluation results in Table 1, the recall and AP of random sampling (average over 5 rounds) are significantly lower than that of our action proposal algorithm.

Action detection on MSRII dataset

Based on the discovered 2000 action proposals, action detection is evaluated according to average precision. Following the cross-dataset detection setting [4], our algorithm based on the action proposals achieves state-of-the-art performance as shown in Table 2. As far as we know, this is the best action detection result on MSRII dataset. Although [11] reports action detection results for MSRI-I dataset, the setting is quite different. The sequences in MSRII dataset are split into two sets (one for training and the other for testing) for [11] while we perform the standard cross-dataset evaluation (training on the KTH dataset but testing on the MSRII sequences).

Method	Handwaving	Handclapping	Boxing	mAP
Ours	0.699	0.466	0.674	**0.613**
Tubelet [27]	0.858	0.314	0.460	0.543
NBMIM [4]	0.649	0.431	0.580	0.553
CAD [7]	0.367	0.132	0.175	0.225
SDPM [21]	0.447	0.239	0.389	0.358
DynamicPoselets [43]	0.809	0.502	0.417	0.576

Table 2. Cross-dataset action detection results on MSRII dataset based on average precision.

Fig. 4 compares precision-recall curves for the three categories of actions. Our algorithm significantly outperforms Cross Action Detection (CAD) [7] and Spatio-temporal Deformable Part Models (SDPM) [21] on all the three actions. Compared with spatio-temporal branch-and-bound search (NBMIM) [4], the better performance of our algorithm is mainly due to the high recall of our action proposals. For example, our algorithm can successfully locate 95% of handclapping actions while NBMIM [4] can only find 57% of handclapping actions. Moreover, compared with these sub-volume based action detection methods [4, 21, 7], our approach is capable to handle the moving actions. In Section 5.2, the challenging UCF 101 dataset will be tested to show that our approach is also useful to localize the moving actions.

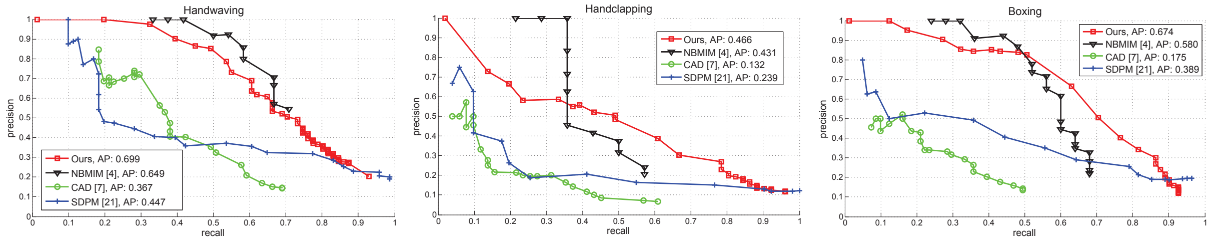

Figure 4. Precision-recall curve for action detection on MSRII dataset.

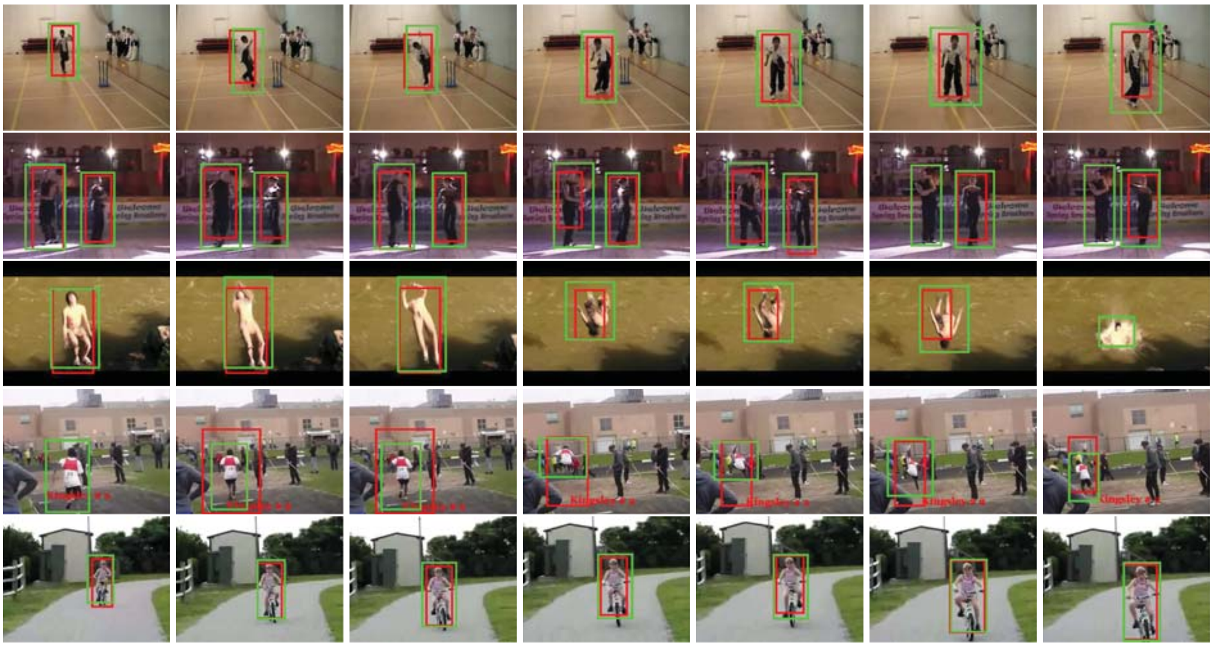

Figure 5. Illustrative examples for the discovered action proposals (with red bounding boxes) on UCF 101 dataset. The ground-truth action is marked by green bounding box. The IOU between the red action proposal with the green ground-truth action for each row is $0.55, (0.37, 0.44), 0.39, 0.38, 0.80$.

Action search on MSRII dataset

Based on the discussion in Section 4.2, action search is evaluated on the MSRII dataset. Following the same evaluation setup in [26], the video sequences from MSRII dataset are used as database while the query video is from KTH dataset. Table 3 provides the action retrieval results based on average precision. Our algorithm provides superior performance compared with state-of-the-art action search algorithm [26] as well as action detection algorithm [7]. The poor perform for the boxing action is mainly due to the indiscriminative of the motion pattern from the boxing action (e.g., similar to walking action).

Method	Handwaving	Handclapping	Boxing	average
Ours	0.601	0.373	0.274	**0.416**
CAD [7]	0.367	0.132	0.175	0.225
RF [26]	0.492	0.312	0.302	0.369

Table 3. Action search on MSRII dataset based on mAP.

5.2. Experimental Results on UCF 101

method	$\theta = 0.1$			$\theta = 0.05$		
	S1	S2	S3	S1	S2	S3
H+M@10K	0.545	0.572	0.564	0.834	0.824	0.828
R@10K	0.001	0.000	0.001	0.105	0.090	0.108
R@100K	0.002	0.002	0.001	0.381	0.383	0.382
R@500K	0.011	0.006	0.014	0.597	0.579	0.581

Table 4. Evaluation of the recall on UCF101 dataset. “H+M” is our algorithm with both human and motion scores. “R” is the random algorithm. θ is the intersection-over-union ratio defined in Section 5.1.

UCF 101 [5] is a challenging dataset with unconstrained environments. For action localization task, there are 3207 video clips with 24 categories of human actions which have bounding box annotations. We follow the three splits defined in [5] to evaluate our algorithm. For the motion score in Section 3.4, the positive data are from the temporal windows based on the ground-truth in the training split while

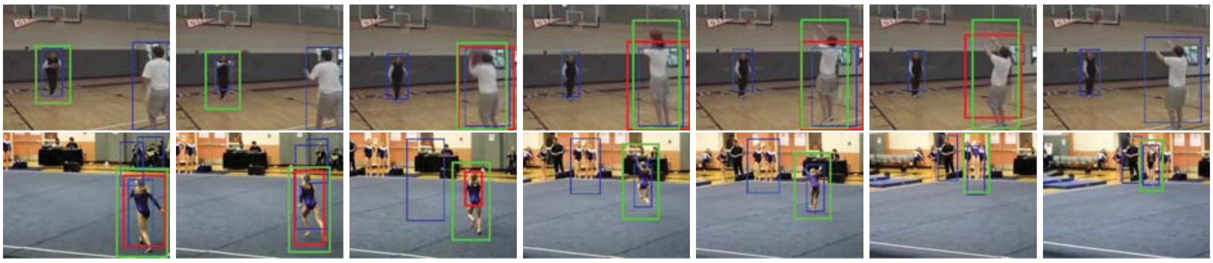

Figure 6. Action proposals (marked with blue rectangle) on two videos from UCF 101 dataset. The action proposal with maximum actionness path score $f(b^*)$ is marked with red rectangle and the ground-truth action is marked by green bounding box.

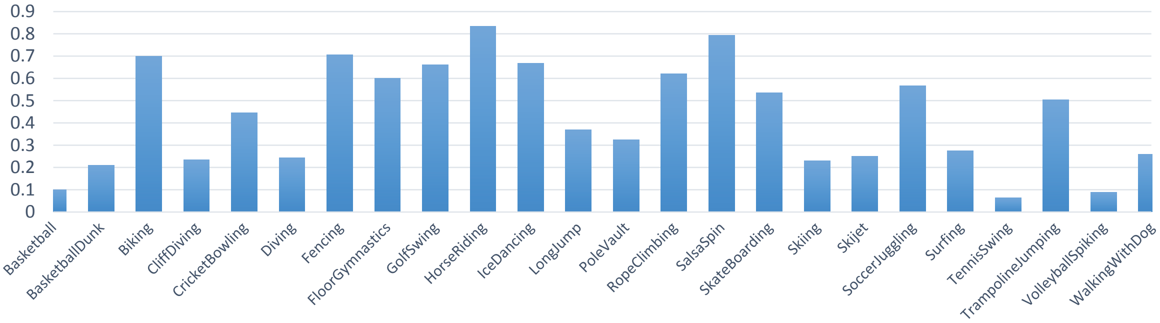

Figure 7. Our action detection results for UCF101 dataset based on average precision.

the negative data are sampled from the video data which does not overlap with the ground-truth.

Table 4 summarizes the recall of our action proposal algorithm. In the first column, “H+M” is our algorithm with both human and motion scores. “R” is the random algorithm. We can see that, based on 10K action proposals for all the video sequences, our action proposal algorithm is significantly better than random sampling (average over 5 rounds), especially when the intersection-over-union threshold $\theta = 0.1$. The results on the three splits (S1, S2, and S3) are consistent.

In Fig. 5, a few action proposals are illustrated with red rectangles. We can see that even with serious pose and scale changes, our action proposal is still able to recover the path of human action. In addition, Fig. 6 provides all the proposed action candidates with a blue rectangle for two videos. The red rectangle describes the action proposal with the maximum actionness score $f(b^*)$ in Eq. 8. It can be found that the proposed action candidates usually cover the potential positions with human actions.

Method	H+M@10K	R@10K
AP	0.428	0.086

Table 5. Action detection results on UCF 101 dataset based on mean average precision.

Action detection on UCF 101 dataset

Based on the discovered action proposals, action detection is further performed by recognizing the specific action category. To be consistent, we follow the same evaluation measure (average precision) in Section 5.1. Table 5 lists the results of our action detection algorithm based on split 1. Our algorithm significantly outperforms the baseline with random sampling. In Fig. 7, we list our results of action detection on the 24 categories of human actions based on average precision.

6. Conclusions

We present a novel method to generate action proposals in unconstrained videos, which can effectively capture spatial-temporal video tube of high potential to include a human action with specific motion pattern. The problem is formulated by a maximum set coverage problem and a greedy-based solution is presented which can efficiently locate the action candidates. Action detection and search can then be applied on the discovered action proposals. As a data-drive approach, our action proposal algorithm can work well with the moving cameras, and can also track the action despite the dynamic and cluttered backgrounds. Promising results have been obtained on two challenging datasets. In the future work, we will try to evaluate our action proposal on more challenging and diverse datasets like TRECVID.

Acknowledgement

This work is supported in part by Singapore Ministry of Education Tier-1 Grant M4011272.040.

References

[1] H. Wang, C. Schmid, "Action Recognition with Improved Trajectories," *ICCV*, 2013. 4, 5

[2] F. Perronnin, J.Sanchez, T. Mensink, "Improving the Fisher kernel for large-scale image classification," *ECCV*, 2010. 5

[3] D. Oneata, J. Revaud, J. Verbeek, C. Schmid, "Spatio-Temporal Object Detection Proposals," *ECCV*, 2014. 2

[4] J. Yuan, Z. Liu, Y. Wu, "Discriminative Video Pattern Search for Efficient Action Detection," in *TPAMI*, pp. 1728 - 1743, Vol. 33, 2011. 2, 5, 6

[5] K. Soomro, A. Zamir, M. Shah, "UCF101: A Dataset of 101 Human Action Classes From Videos in The Wild," in *CRCV-TR*, 2012. 2, 5, 7

[6] T. Dean, J. Yagnik, M. Ruzon, M. Segal, J. Shlens, S. Vijayanarasimhan, "Fast, Accurate Detection of 100,000 Object Classes on a Single Machine," *CVPR*, 2013. 5

[7] L. Cao, Z. Liu, and T.S. Huang, "Cross-dataset action detection," *CVPR*, 2010. 5, 6, 7

[8] B. Alexe, T. Deselaers, V. Ferrari, "What is an object?," *CVPR*, 2010. 1, 2

[9] N. cinbis, S. Sclaroff, "Object , Scene and Actions : Combining Multiple Features for Human Action Recognition," *ECCV*, 2010. 2

[10] Y. Xie, H. Chang, Z. Li, L. Liang, X. Chen, D. Zhao, "A unified framework for locating and recognizing human actions," *CVPR*, 2011. 2

[11] P. Siva, T. Xiang, "Weakly Supervised Action Detection," *BMVC*, 2011. 2, 6

[12] T. Lan, Y. Wang, G. Mori, "Discriminative figure-centric models for joint action localization and recognition," *ICCV*, 2011. 2

[13] D. Tran, J. Yuan, "Max-Margin Structured Output Regression for Spatio-Temporal Action Localization," *NIPS*, 2012. 2

[14] G.L. Nemhauser, L.A. Wolsey, M.L. Fisher, "An analysis of approximations for maximizing submodular set functions I," *Mathematical Programming*, Vol. 14, pp. 265-294, 1978. 2, 3

[15] J. Gall, A. Yao, N. Razavi, L. Van Gool, V. Lempitsky, "Hough forests for object detection, tracking, and action recognition," *PAMI*, Vol. 33(11), pp. 2188 - 202, 2011. 2

[16] J. Uijlings, K. van de Sande, T. Gevers, A. Smeulders, "Selective search for object recognition," in *IJCV*, 2013. 2

[17] S. Ma, J. Zhang, N. Cinbis, S. Sclaroff, "Action Recognition and Localization by Hierarchical Space-Time Segments," *ICCV*, 2013. 1

[18] D. Oneata, J. Verbeek, C. Schmid, "Action and Event Recognition with Fisher Vectors on a Compact Feature Set," *ICCV,* 2013 5

[19] L. Bourdev, J. Malik, "Poselets: Body part detectors trained using 3d human pose annotations," *ICCV*, 2009. 4, 5

[20] M. Van Den, G. Roig, X. Boix, S. Manen, L.V. Gool, "Online video seeds for temporal window objectness," *ICCV*, 2013. 2

[21] Y. Tian, R. Sukthankar, M. Shah, "Spatiotemporal Deformable Part Models for Action Detection," *CVPR,* 2013 1, 6

[22] Y. Ke, R. Sukthankar, M. Hebert, "Event Detection in Crowded Videos," *ICCV,* 2007 2

[23] P. Siva, T. Xiang, "Action Detection in Crowd," *BMVC,* 2010 2

[24] C. Schuldt, I. Laptev, B. Caputo, "Recognizing Human Actions : A Local SVM Approach," *ICPR,* 2004 5

[25] N. Shapovalova, M. Raptis, L. Sigal, G. Mori, "Action is in the Eye of the Beholder : Eye-gaze Driven Model for Spatio-Temporal Action Localization," *NIPS,* 2013 1

[26] G. Yu, J. Yuan, Z. Liu, "Unsupervised Random Forest Indexing for Fast Action Search," in *CVPR*, 2011. 1, 5, 7

[27] M. Jain, J. Gemert, H. Jegou, P. Bouthemy, C. Snoek, "Action localization by tubelets from motion," in *CVPR*, 2014. 2, 6

[28] R. Benenson, M. Mathias, R. Timofte, L.V. Gool, "Pedestrian detection at 100 frames per second," in *CVPR*, 2012. 4, 5

[29] P. Dollr, R. Appel, S. Belongie, P. Perona, "Fast Feature Pyramids for Object Detection," in *PAMI*, 2014. 4, 5

[30] D. Tran, J. Yuan, D. Forsyth, "Video Event Detection: from Subvolume Localization to Spatio-Temporal Path Search," *PAMI*, 2014. 1, 2, 3

[31] M. Cheng, Z. Zhang, W.Y. Lin, P. Torr, "BING: Binarized normed gradients for objectness estimation at 300fps," *CVPR*, 2014. 1, 2

[32] J. Hosang, R. Benenson, B. Schiele, "How good are detection proposals, really?," *BMVC*, 2014. 2

[33] R. Girshick, J. Donahue, T. Darrell, J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," *CVPR*, 2014. 1, 2

[34] W. Chen, C. Xiong, R. Xu, J. Corso, "Actionness ranking with lattice conditional ordinal random fields," *CVPR*, 2014. 2, 4

[35] H. Pirsiavash, D. Ramanan, C. Fowlkes, "Globally-optimal greedy algorithms for tracking a variable number of objects," *CVPR*, 2011. 2

[36] G. Yu, J. Yuan, Z. Liu, "Propagative Hough Voting for Human Activity Recognition," *ECCV*, 2012. 2

[37] C. Xu, J.J. Corso, "Evaluation of super-voxel methods for early video processing," *CVPR*, 2012. 2

[38] F. Galasso, N.S. Nagaraja, T.J. Crdenas, T. Brox, B. Schiele, "A unified video segmentation benchmark: Annotation, metrics and analysis," *ICCV*, 2013. 2

[39] D. Banica, A. Agape, A. Ion, C. Sminchisescu, "Video object segmentation by salient segment chain composition," *ICCVW*, 2013. 2

[40] G. Yu, J. Yuan, Z. Liu, "Real-time Human Action Search using Random Forest based Hough Voting," *ACM Multimedia Conference*, 2011. 2

[41] W. Brendel, M. Amer, S. Todorovic, "Multiobject tracking as maximum weight independent set," *CVPR*, 2011. 2

[42] W.S. Chu, F. Zhou, F. Torre, "Unsupervised Temporal Commonality Discovery," *ECCV*, 2012. 2

[43] L. Wang, Y. Qiao, X. Tang, "Video Action Detection with Relational Dynamic-Poselets," *ECCV*, 2014. 6