This ICCV2013 paper is the Open Access version, provided by the Computer Vision Foundation.
The authoritative version of this paper is available in IEEE Xplore.

Video Co-segmentation for Meaningful Action Extraction

Jiaming Guo[1], Zhuwen Li[1], Loong-Fah Cheong[1] and Steven Zhiying Zhou[1,2]
[1]Department of ECE, National University of Singapore, Singapore
[2]National University of Singapore Research Institute, Suzhou, China
{guo.jiaming, lizhuwen, eleclf, elezzy}@nus.edu.sg

Abstract

Given a pair of videos having a common action, our goal is to simultaneously segment this pair of videos to extract this common action. As a preprocessing step, we first remove background trajectories by a motion-based figure-ground segmentation. To remove the remaining background and those extraneous actions, we propose the trajectory co-saliency measure, which captures the notion that trajectories recurring in all the videos should have their mutual saliency boosted. This requires a trajectory matching process which can compare trajectories with different lengths and not necessarily spatiotemporally aligned, and yet be discriminative enough despite significant intra-class variation in the common action. We further leverage the graph matching to enforce geometric coherence between regions so as to reduce feature ambiguity and matching errors. Finally, to classify the trajectories into common action and action outliers, we formulate the problem as a binary labeling of a Markov Random Field, in which the data term is measured by the trajectory co-saliency and the smoothness term is measured by the spatiotemporal consistency between trajectories. To evaluate the performance of our framework, we introduce a dataset containing clips that have animal actions as well as human actions. Experimental results show that the proposed method performs well in common action extraction.

1. Introduction

Consider Fig. 1, which shows two frames from a video example: Two penguins are tobogganing and one penguin is walking. One basic task of a vision system is to extract the interesting foreground of this video. This begs the question: What is the interesting foreground? A straightforward approach would be to extract those objects that move in the scene [3, 12]. In that case, all the penguins in Fig. 1 would be extracted as foreground. Clearly, this simple criteria is not fine-grained enough for many applications where specific kinds of actions may be of interest. The labeling task for these latter kinds of applications can be collectively termed as video tag information supplementation [24]. For instance, most of the videos on Youtube are currently tagged with a set of keywords by the owners. However, the manual tagging process through which this is usually done is quite unwieldy and would require far too much labor to provide additional information such as when and where the contents depicted by the tags actually appear in the tagged video. Therefore, most of the videos are only provided with a simple tag. It will be desirable for a video tag information supplementation system to augment the tag with more fine-grained information. For example, if the video in Fig. 1 is tagged as "Penguin Tobogganing", only the tobogganing penguins should be extracted as foreground. Another example is that if the content referred to by the tag only appears in some frames of a long video, only those frames should be retrieved, while the others can be discarded.

Figure 1. Two frames in a video example. If the desired action is penguin tobogganing, motion cues alone would fail to identify the correct foreground.

Such informative content retrieval also has important benefits for action recognition or detection. In the training of an action classifier or detector, the collection of positive examples includes not only gathering videos that contain useful information, but also retrieving those pertinent parts from these videos. Most of the existing action recognition or detection methods simply rely on the labor intensive process of manually drawing boxes to define the action [13, 17]. For the case of human actions, while one may make use of human pose estimation [1] for automatic retrieval of the relevant bounding boxes [22], the method may still fail when there exist extraneous actions.

A similar problem exists in object-oriented image segmentation where there might exist extraneous objects in the

foreground. To handle this problem, the technique of image co-segmentation has been used [15, 21, 9]. It simultaneously segments common regions occurring in multiple images. In this paper, we develop an analogous video co-segmentation framework for common action extraction, which allows us to extract the desired action without having to use higher level cues or any learning process.

In image co-segmentation [21, 9], a pair of regions are defined to be co-salient if they exhibit strong internal coherence and strong local saliency w.r.t the background, and the correspondences between the regions are supported by high appearance similarity of features. Our work is based on the similar concept of trajectory co-saliency. Compared to the case of image, the time dimension in video presents significant challenges that must be overcome before the trajectory co-saliency idea can be realized effectively. First we must have a set of effective spatiotemporal descriptors that can discriminate various animate and inanimate motion patterns. The second challenge is the additional variation brought about by the time dimension. Not only the common action across the multiple videos can be misaligned in both time and space, the action may also exhibit significantly more complex intra-class variation.

We address these challenges at various levels. At the most basic level, we adopt dense trajectories as the measurement unit, as they capture the long-term dynamics of an action better [23]. Compared to other representation such as tubes [16] or cuboid [6], trajectory representation allows us to track action details explicitly without having to deal with the extraneous region that inevitably comes with a space-time volume approach. We adopt the motion boundary histogram (MBH) [5] to describe the spatiotemporal variation of motion along the trajectory, as well as to help suppress the uninformative constant motion induced by camera motions. We then build upon the MBH so as to accommodate similarity measurement between trajectories with different lengths and probably spatiotemporally misaligned.

Relying solely on similarity measurement at the level of a single trajectory would result in many ambiguous matches as it is not unlikely that two trajectories from different actions share some similarities. Instead, we carry out matching at the level of trajectory clusters. We first associate each trajectory with a trajectory cluster by a spatiotemporal over-segmentation within each video; then, a trajectory is co-salient if 1) the trajectory cluster it belongs to succeeds in finding a large number of trajectory matches in another trajectory cluster of the other video, and 2) these trajectory matches exhibit high geometric coherence.

The final step is formulated as a binary labeling of a Markov random field (MRF) which classifies the trajectories into common action and action outliers. The data term penalizes any foreground trajectories with low co-saliency and vice versa, and the smoothness term penalizes the assignment of different labels to two trajectories near in some spatiotemporal sense in the same video.

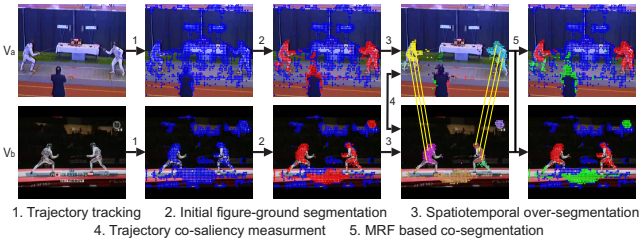

Figure 2. Overview of the system. Best viewed in color.

2. Overview

Fig. 2 shows the overview of our system. Given two videos that contain a similar action, we first use the tracker of [19] to generate dense trajectories in each video.

Next, we perform a "background subtraction" in each video to remove the background trajectories as much as possible. We eschew the 3D motion segmentation approaches [18, 26], as they are in general numerically suspect for large number of motions. Instead we propose a figure-ground segmentation step which is based on 2D motion cues. While it contains several improvements over [7, 12] so as to better extract motion with small contrast (Sect. 3), it is not the main focus of this paper and we do not assume that good background subtraction in either video is a must.

After the initial background subtraction, the remaining trajectories in the videos might still contain action outliers, namely, the remaining background trajectories and those extraneous actions. To remove these action outliers, we simultaneously perform the segmentation of the remaining trajectories from both videos. This co-segmentation problem is finally cast as a binary labeling of a MRF (Sect. 5, step 5 of Fig. 2), with its customary data and smoothness terms. The preceding steps (3 and 4) basically compute the data term, i.e., the trajectory co-saliency that rewards common observations among multiple videos. We first associate each trajectory with a trajectory cluster by a spatiotemporal over-segmentation within each video (Sect. 4.2, step 3 in Fig. 2). Trajectory correspondence candidates are initialized using the proposed extended MBH (Sect. 4.1). Then, the trajectory co-saliency is computed by taking into account both the feature similarity of the trajectories and the geometric coherence of the associated regions via a graph matching framework (Sect. 4.3, step 4 of Fig. 2).

3. 2D Motion based Figure-Ground Segmentation

Let $\mathcal{T}$ denote the trajectory set of a video clip V. Our objective in this step is to separate $\mathcal{T}$ into the foreground $\mathcal{F}$ and the background $\mathcal{B}$. The foreground trajectories are those with high motion saliency w.r.t. the background.

Algorithm 1 GMM based figure-ground segmentation

Input: Trajectory set $\mathcal{T}$ of a video clip having L frames
  $\mathcal{B} \leftarrow \mathcal{T}, \mathcal{F} \leftarrow \emptyset$;
  for $t = 1 \rightarrow L - T$ **do**
    while true **do**
      Compute s_t^i of all $\mathrm{tr}^i \in \mathcal{B} \cap \mathcal{T}_t$ using (1);
      Fit a GMM based on s_t^i and compute ψ using (2);
      $\mathcal{F}_t \leftarrow \{\mathrm{tr}^i | s_t^i > \psi\}, \mathcal{F} \leftarrow \mathcal{F} \cup \mathcal{F}_t, \mathcal{B} \leftarrow \mathcal{B} - \mathcal{F}_t$;
      if $F_t = \emptyset$ **then**
        break;
      end if
    end while
  end for
  return $\mathcal{F}$ and $\mathcal{B}$

Denote the i-th trajectory in $\mathcal{T}$ as tr^i. The Euclidean distance between two trajectories tr^i and tr^j at a particular instant t is $d_t(\mathrm{tr}^i, \mathrm{tr}^j) = \frac{1}{T}\{(u_t^i - u_t^j)^2 + (v_t^i - v_t^j)^2\}$, where $u_t^i = x_{t+T}^i - x_t^i$ and $v_t^i = y_{t+T}^i - y_t^i$ denote the motion of tr^i aggregated over T frames. We set T as 5 in our implementation. Use s_t^i to represent the saliency of tr^i at time t. We measure s_t^i using the median value of the distances between tr^i and all the others, i.e.,

$$s_t^i = \mathrm{median}\{d_t(\mathrm{tr}^i, \mathrm{tr}^k), \mathrm{tr}^k \in \mathcal{T}_t, k \neq i\}, \qquad (1)$$

where $\mathcal{T}_t$ is the set of trajectories present from t to $t + T$.

After calculating s_t^i of all trajectories present at t, we can use a threshold ψ to extract those trajectories of high s_t^i. Our intuition is that the background is usually the largest object in the scene, and thus, for any particular trajectory in the background, there usually exist a large amount of trajectories elsewhere in the scene that move in a similar manner and hence its median value s_t^i will be small. To set a proper ψ, we can fit a 1D Gaussian Mixture Model (GMM) with two components $f(s) = \sum_{c=1,2} \pi_c \mathcal{N}(s|\mu_c, \sigma_c)$, where $\mathcal{N}$ is a Gaussian with mean μ_c ($\mu_1 < \mu_2$) and standard deviation σ_c, and π_c is the mixing coefficient. A straightforward way to set ψ is to use the mean value of μ_1 and μ_2, i.e., $\psi = \frac{\mu_1+\mu_2}{2}$. However, this is not reasonable when μ_1 is very close to μ_2, indicating that the GMM process fails to isolate the foreground component so that both of the two fitted components mainly contain the background trajectories. We thus compare the difference between μ_1 and μ_2 against a threshold ρ to determine whether it falls into this situation, and if so, ψ should be set relying only on μ_1, i.e.,

$$\psi = \begin{cases} \frac{\mu_1+\mu_2}{2}, & \text{if } \mu_2 - \mu_1 > \rho, \\ \mu_1 + \frac{\rho}{2}, & \text{otherwise.} \end{cases} \qquad (2)$$

The ρ in (2) controls the sensitivity of motion detection: The lower ρ is, the more trajectories will be detected as moving.

For every time instant, we perform the GMM based segmentation iteratively. This is because if one foreground object possesses larger motion contrast than another one, it would likely happen that one component generated by the GMM fitting contains the object with larger motion contrast, while the other contains the background and the object with smaller motion contrast. Thus, a further GMM fitting process excluding the trajectories that are already classified as foreground is needed to extract the object with smaller motion contrast. In our algorithm, the iteration is stopped when all remaining trajectories are classified as background.

In a video shot, some objects may be stationary in the beginning but move later. Therefore, we carry out the proposed GMM based segmentation in a sliding window manner with the window size set as T. The complete algorithm is summarized in Algorithm 1.

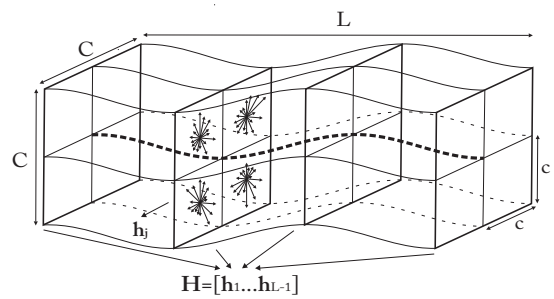

Figure 3. The extraction of the MBH set. $\mathbf{h}_j$ contains two components from I_u and I_v respectively. Here we only show one.

4. Trajectory Co-Saliency

Given two videos V_a and V_b, we denote the trajectories remaining in V_a and V_b after the initial background subtraction as $\mathcal{F}_a = \{\mathrm{tr}_a^1, ..., \mathrm{tr}_a^m\}$ and $\mathcal{F}_b = \{\mathrm{tr}_b^1, ..., \mathrm{tr}_b^n\}$ respectively.

4.1. Trajectory Feature Distance Measurement

Given a trajectory tr^i of length L_i with its local neighborhood of $C \times C$ pixels in each frame, a 3D volume of size $C \times C \times L_i$ can be obtained. From each pair of successive frames, we extract the MBH $\mathbf{h}$ as follows: 1) Compute dense optical flow u and v (which we already obtained during the dense trajectory generation step), 2) Treating the two "flow images" I_u, I_v independently, compute the corresponding gradient magnitudes and orientations, and 3) Use them to weigh the votes into local orientation histograms (refer to [5, 23] for details). We set the block size $C = 32$, the cell size $c = 16$ and the bin number $b = 16$ in each cell for full orientation quantization (See Fig. 3). Based on these, two histograms with $2 \times 2 \times b = 64$ bins are obtained from I_u and I_v respectively; we simply concatenate them to generate a 128-bin histogram. We next normalize the final 128-bin histogram using its ℓ_2-norm. The MBH feature helps to suppress the non-informative constant motion induced by camera motions.

After extracting all the MBH features of tr^i, we can represent tr^i using $\mathbf{H}^i = [\mathbf{h}_1^i, \ldots, \mathbf{h}_{L_i-1}^i]$. To measure the fea-

ture distance between two inter-video trajectories, all $\mathbf{h}_k^i$ in $\mathbf{H}_i$ should be treated as elements of a set, in view of the lack of temporal alignment. Thus, a set-to-set similarity definition is required. The straightforward "min-dist" measure [25] could have been used. However, this measure discards most of the information from $\mathbf{H}^i$, which is not desirable since even two very different types of actions may share the same local feature at some time instant. Here we propose to take advantage of the accumulated energy of each bin of the MBHs across frames, and the temporal correlation between these bins. We first compute

$$\mathbf{P}^i = \frac{1}{L_i - 1}\mathbf{H}^i{\mathbf{H}^i}^{\mathrm{T}}. \tag{3}$$

It is evident that the diagonal elements of $\mathbf{P}_i$ are the average energy of each bin and the non-diagonal ones represent the temporal correlation between different bins. We produce the final feature by taking the upper triangle elements of $\mathbf{P}^i$ and rearranging them as $\mathbf{p}^i \triangleq [P_{11}, P_{12}, \ldots, P_{1d}, P_{22}, P_{23}, \ldots, P_{dd}]$. Then, we measure the distance between two trajectories from different videos as follows:

$$d_{\text{inter}}(\mathrm{tr}_a^i, \mathrm{tr}_b^j) = \|\mathbf{p}_a^i - \mathbf{p}_b^j\|_2. \tag{4}$$

4.2. Trajectory Grouping

We now associate each trajectory in a video with a trajectory cluster, so that geometric coherence can be brought to bear on the measurement of trajectory co-saliency. To form the clusters, we adapt the trajectory grouping method proposed in [13]. Given two trajectories tr^i and tr^j that co-exist in a time interval $[\tau_1, \tau_2]$, their distance is defined as follows:

$$d_{\text{intra}}^{ij} = \max_{t\in[\tau_1,\tau_2]} d_{\text{spatial}}^{ij}[t] \cdot \frac{1}{\tau_2 - \tau_1}\sum_{k=\tau_1}^{\tau_2} d_{\text{velocity}}^{ij}[k], \tag{5}$$

where $d_{\text{spatial}}^{ij}[t]$ is the Euclidean distance of the trajectory points of tr^i and tr^j at the time instant t, and $d_{\text{velocity}}^{ij}[t]$ is that of the velocity estimate. Then, the affinity between trajectories i and j is computed as follows and stored in the (i, j) entry of the affinity matrix W:

$$\mathbf{W}(i,j) = \begin{cases} 0, & \text{if } \max_{t\in[\tau_1,\tau_2]} d_{\text{spatial}}^{ij}[t] \geq 30, \\ \exp(-d_{\text{intra}}^{ij}), & \text{otherwise.} \end{cases} \tag{6}$$

It enforces spatial compactness by setting the affinity to be zero for trajectory pairs not spatially close. If two trajectories never co-exist at any time interval, the affinity between them is also set to zero.

Assuming there are n trajectories, an $n \times n$ affinity matrix $\mathbf{W}$ is constructed. Spectral clustering [10] is then used to segment these n trajectories into K clusters. As for the number of clusters K, we do not need to set it to be exactly the number of objects or motions. We only need to ensure the cluster size is large enough so that the cluster-to-cluster matching procedure has enough number of trajectories to make good decision. Thus, in our experiment, we simple set $K = \mathrm{ceil}(n/200)$.

4.3. Graph Matching

Denote the trajectory clusters obtained from $\mathcal{F}_a$ and $\mathcal{F}_b$ as $\mathcal{C}_a = \{\mathcal{C}_a^1, ..., \mathcal{C}_a^{K_a}\}$ and $\mathcal{C}_b = \{\mathcal{C}_b^1, ..., \mathcal{C}_b^{K_b}\}$ respectively. Following the graph matching formulation in [8], the matching score of two trajectory clusters $\mathcal{C}_a^h$ and $\mathcal{C}_b^k$ (from $\mathcal{C}_a$ and $\mathcal{C}_b$ respectively) can be computed as

$$\begin{aligned} S(\mathcal{C}_a^h, \mathcal{C}_b^k) &= \frac{1}{|\mathcal{C}_a^h| + |\mathcal{C}_b^k|}\left\{\max_{\mathbf{x}} \mathbf{x}^{\mathrm{T}}\mathbf{M}_{hk}\mathbf{x}\right\} \\ \text{s.t.} &\begin{cases} \mathbf{x} \in \{0,1\}^{pq} \\ \mathbf{X}\mathbf{1}_{q\times 1} \preceq \mathbf{1}_{p\times 1}, \mathbf{X}^{\mathrm{T}}\mathbf{1}_{p\times 1} \preceq \mathbf{1}_{q\times 1}, \end{cases} \end{aligned} \tag{7}$$

where $|\cdot|$ denotes the cardinality of a set; p and q denote the number of trajectory candidates for matching in $\mathcal{C}_a^h$ and $\mathcal{C}_b^k$ respectively; $\mathbf{X} \in \{0,1\}_{p\times q}$ is a binary assignment that represents possible trajectory correspondences; $\mathbf{x} \in \{0,1\}^{pq}$ denotes a column-wise vectorized replica of $\mathbf{X}$; the two-way constraints of (7) ensure that there are only one-to-one trajectory matches; $\mathbf{M}_{hk}$ is a $pq \times pq$ symmetric matrix encoding geometric coherence, with its diagonal element $\mathbf{M}_{hk}(il, il)$ representing the self-coherence of a trajectory correspondence candidate (i, l), and the non-diagonal element $\mathbf{M}_{hk}(il, jr)$ representing the pair-wise coherence of two correspondence candidates (i, l) and (j, r). In other words, $\mathbf{M}_{hk}(il, jr)$ is set to be small if the deformation between (i, j) and (l, r) is large.

In our implementation, we first initialize those inter-video trajectory pairs with the top 0.01% smallest inter-video feature distances (calculated in (4)) as the correspondence candidates. Then, the graph matching is performed only between those trajectory clusters $\mathcal{C}_a^h$ and $\mathcal{C}_b^k$ containing at least 2 correspondence candidates, while the matching scores S between those containing less than 2 correspondence candidates are set to zero. To construct $\mathbf{M}_{hk}$, the unary terms $\mathbf{M}_{hk}(il, il)$ are set to 0 since all selected correspondence candidates tend to have high and similar unary affinity, rendering it unnecessary to differentiate them. As for the pair-wise terms $\mathbf{M}_{hk}(il, jr)$, we first compute the relative polar coordinates of the trajectory pair (i, j), i.e., $\mathbf{c}_{ij} = \{(d_{ij}^{\tau_1}, \theta_{ij}^{\tau_1}), \ldots, (d_{ij}^{\tau_2}, \theta_{ij}^{\tau_2})\}$, where $[\tau_1, \tau_2]$ is the time interval over which the trajectories i and j co-exist. $\mathbf{c}_{lr}$ is similarly defined. Imposing strong inter-region geometric coherence means that we demand $\mathbf{c}_{ij}$ and $\mathbf{c}_{lr}$ to be similarly distributed. Assuming both d and θ are Gaussian-distributed, $\mathbf{M}_{hk}(il, jr)$ is then computed as $\mathbf{M}_{hk}(il, jr) =$

Figure 4. The results of graph matching between trajectory clusters. The matching scores by (7) are overlaid at the top.

$\exp\{-\frac{1}{2}(\mathrm{Bh}(d_{ij}, d_{lr}) + \mathrm{Bh}(\theta_{ij}, \theta_{lr}))\}$, where $\mathrm{Bh}(\cdot,\cdot)$ represents the Bhattacharyya distance between two Gaussian distributions. To solve the optimization problem in (7), we use the spectral matching proposed in [8].

Fig. 4 shows some graph matching results. It can be seen in the second and third rows that there may be many correspondence candidates not belonging to the common action. However, the association of trajectory clusters and the incorporation of graph matching help to suppress the matching scores of the erroneous matches and significantly boost those of the correct ones (the first row in Fig. 4).

4.4. Co-Saliency Measurement

With the matching scores of all inter-video cluster pairs at our disposal, we can now compute the co-saliency of a trajectory in $\mathcal{F}_a$ w.r.t $\mathcal{F}_b$ as follows:

$$\mathrm{Mt}(\mathrm{tr}_a^i, \mathcal{F}_b) = \max_{\mathrm{C}_b^k \in \mathcal{C}_b} S(\mathcal{C}_a^h, \mathcal{C}_b^k), \mathrm{tr}_a^i \in \mathcal{C}_a^h. \quad (8)$$

which assigns the best cluster-to-cluster matching score of $\mathcal{C}_a^h$ as the co-saliency of all trajectories within this cluster.

5. MRF Based Co-Segmentation

The final classification of the trajectories into common action and action outliers is cast as a binary labeling of a MRF. This is achieved by minimizing an energy function incorporating the trajectory co-saliency measure as the data term, subject to suitable smoothness measure. Formally, denoting the union set of $\mathcal{F}_a$ and $\mathcal{F}_b$ as $\mathcal{U} = \{\mathrm{tr}_a^1, ..., \mathrm{tr}_a^m, \mathrm{tr}_b^1, ..., \mathrm{tr}_b^n\}$, our task is to find $\Sigma = \{\sigma_a^1, ..., \sigma_a^m, \sigma_b^1, ..., \sigma_b^n\}$ so that $\sigma_v^i \in \{0, 1\}$ indicates whether tr_v^i belongs to the action outliers or the common action.

The optimal binary labeling is computed by minimizing the following energy function over the labels σ_a and σ_b:

$$E_T(\Sigma, \mathcal{U}) = E(\sigma_a, \mathcal{F}_a, \mathcal{F}_b) + E(\sigma_b, \mathcal{F}_b, \mathcal{F}_a), \quad (9)$$

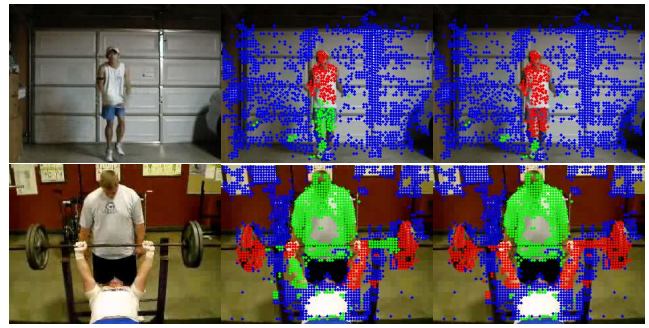

Figure 5. Better results can be obtained by MRF labeling rather than simply thresholding the trajectory co-saliency. From left to right: Original frames, the segmentation by thresholding the co-saliency using $\gamma = 0.2$, and the segmentation using MRF labeling.

where

$$E(\sigma_{v_1}, \mathcal{F}_{v_1}, \mathcal{F}_{v_2}) = \sum_{\mathrm{tr}_{v_1}^i \in \mathcal{F}_{v_1}} D\left(\sigma_{v_1}^i, \mathrm{tr}_{v_1}^i, \mathcal{F}_{v_2}\right) + \lambda_V \sum_{\{i,j\} \in \mathcal{N}} V(\sigma_{v_1}^i, \sigma_{v_1}^j, \mathrm{tr}_{v_1}^i, \mathrm{tr}_{v_1}^j),$$

which consists of a data term D and a smoothness term V, with λ_V as the weighing factor. The purpose of D is to penalize the assignment of trajectories with low co-saliency to the common action and vice versa. It is defined as:

$$D(\sigma, \mathrm{tr}, \mathcal{F}) = \delta_{\sigma,1}(1 - f(\mathrm{tr}, \mathcal{F})) + \delta_{\sigma,0} f(\mathrm{tr}, \mathcal{F}), \quad (10)$$

where $\delta_{\cdot,\cdot}$ is the Kronecker delta, i.e., $\delta_{p,q} = 1$ if $p = q$ and otherwise $\delta_{p,q} = 0$; f is in turn defined as

$$f(\mathrm{tr}, \mathcal{F}) = max\left(0, \mathrm{sign}(\hat{\mathrm{M}}\mathrm{t}(\mathrm{tr}, \mathcal{F}) - \gamma)\right), \quad (11)$$

in which $\hat{\mathrm{M}}\mathrm{t}(\cdot,\cdot)$ linearly normalizes the trajectory co-saliency $\mathrm{Mt}(\cdot,\cdot)$ in (8) to $[0, 1]$ and γ is a threshold.

The smoothness term V encourages the labeling to be spatiotemporally consistent and is defined as

$$V(\sigma^i, \sigma^j, \mathrm{tr}^i, \mathrm{tr}^j) = (1 - \delta_{\sigma^i,\sigma^j}) \exp(-d_{\text{intra}}^{ij}), \quad (12)$$

where d_{intra}^{ij} is calculated as in (5). To build the neighbor pair set $\mathcal{N}$, we use Delaunay triangulation to connect the tracked points for each frame of a video. Any pair of trajectories that are ever connected by one of the Delaunay triangulations is considered to be a neighbor pair.

Since it is a binary labeling with the smoothness term being a metric, the global minimum can be computed via graph cuts [2]. Note that the labeling is processed at the trajectory level rather than at the cluster level, since it is easier to impose the spatiotemporal smoothness constraint (12). This smoothness constraint helps to restore parts of the common action that are not initially detected as co-salient back to their correct group. The superiority of the MRF labeling results is illustrated by Fig. 5.

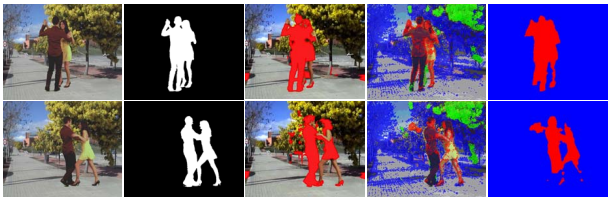

Figure 6. Co-segmentation of the *Cha-cha-cha* videos. From left to right: original frames, ground truth, segmentation of [16], our method with $\gamma = 0.1$ and that post-processed by [11].

video	chacha1	chacha2	chacha3	chacha4
Labeling accuracy on dense trajectories (%)				
ours ($\gamma = 0.1$)	98	98	97	98
Labeling accuracy on pixels (%)				
ours ($\gamma = 0.1$) plus [11]	96	96	95	96
[16]	61	81	56	74

Table 1. Co-segmentation results of the *Cha-cha-cha* videos.

6. Experiment

6.1. Comparison with [16]

In this subsection, we apply our method on the *Cha-cha-cha* videos from the Chroma database [20] and compare the results with those reported in [16]. Since [16] presented their results in terms of pixels rather than dense trajectories like ours, we use the method of [11] to turn our trajectory labels into pixel labels for comparison. The ratio of correct labels (labeling accuracy) is summarized in Table 1. It can be seen that our method obtains at least 97% labeling accuracy at the level of trajectories; for pixels, our method achieves at least 95%, significantly better than the results of [16] (74%). As reported in [16] and can be seen from the third column of Fig. 6, its algorithm is sensitive to wrong initial segmentation caused by those background contents that confuse the objectness and saliency measures.

6.2. Experiment on a 80-pair Dataset

Dataset and Evaluation Method: We build a dataset comprising 80 pairs of sequences containing a significant amount of action outliers in the sense defined by this paper. Among them, 50 are selected from the UCF50 [14] depicting human actions. We should remark that these 50 human action video pairs are temporally segmented, i.e., the tagged actions appear throughout the clips. Another 30 pairs are excerpted mainly from various BBC animal documentaries depicting animal actions. Different from the collected human action videos, the animal action videos are relatively longer (most of them have more than 300 frames) and the tagged actions need not stretch over the entire videos. Table 2 lists all the included action tags and the corresponding number of pairs. Taken together, these 80 video pairs allow us to evaluate our algorithm's performance on both the spatial and temporal extraction of the tagged contents.

We have annotated all the common actions with bounding boxes in order to quantify our common action extraction performance (Examples of the bounding boxes can be seen in Fig. 8 and Fig. 9). For the 30 animal action video pairs, indices of all frames where the tagged actions appear are also given. To evaluate the performance on action outlier detection, we measure the action outlier detection error (AODE) as the number of bad labels of the action outliers over the total number of action outliers, which is estimated by counting the moving trajectories outside the bounding boxes. To evaluate the performance on spatial localization, we define the localization score (following [13]) as $\mathrm{LOC} = \frac{1}{T}\sum_{t=1}^{T}[\frac{|\mathcal{A}_t \cap \mathcal{L}_t|}{|\mathcal{A}_t|} \geq \alpha]$, where $[\cdot]$ is the zero-one indicator function, $\mathcal{L}_t$ and $\mathcal{A}_t$ are respectively the set of points inside the annotated bounding box and those belonging to the detected common action at the time instant t, and α is a threshold that defines the minimum acceptable overlap. Moreover, we evaluate the coverage of our localization using $\mathrm{COV} = \frac{1}{T}\sum_{t=1}^{T}\frac{\mathrm{Ar}(\mathcal{A}_t \cap \mathcal{L}_t)}{\mathrm{Ar}_g}$, where $\mathrm{Ar}(\cdot)$ is the area of the minimum bounding box formed by a set of given points, and Ar_g is the area of the given bounding box.

Human Action	Num.	Animal Action	Num.
Basketball Shooting	5	Big Cat Running	1
Bench Press	6	Big Cat Walking	3
Clean and Jerk	3	Bird Swallowing Prey	4
Fencing	6	Dragonfly's Ovipositing Flight	2
Horse Riding	3	Frog Jumping off	1
Jumping Rope	5	Frog Calling	1
Lunges	6	Inchworm Moving	2
Pommel Horse	2	Kangaroo Jumping	3
Rope Climbing	1	Penguin Tobogganing	3
Skate Boarding	2	Penguin Walking	4
Skiing	6	Snake Slithering	3
Swing	5	Dolphin Breaching	3

Table 2. Action tags included in the dataset and the corresponding number of pairs of sequences.

To evaluate the temporal localization performance, we compute another two measures: missing rate (MR) and false alarm rate (FAR). Denoting those frames where the common action appears as active frame, MR then represents the rate of error of mistaking active frames for non-active frames, whereas FAR represents that of mistaking non-active frames for active frames. To determine whether a frame in a clip is to be regarded as active or not, we first find the frame in this clip that contains the maximum number of detected common action points and denote this maximum number as N_{max}. Then, all frames in this clip that contain more than ηN_{max} ($\eta < 1$) detected common action points are regarded as active frames. As a baseline comparison for temporal localization, we select a recently proposed method for unsupervised temporal commonality discovery (TCD) [4]. It uses a Bag of Temporal Words (BoTW) model to represent video segments and then finds two subsequences from two different videos having the minimum distance measure, with the constraint that both of the subsequences are longer than a preset window length L. In the experiment, we use the MBH as the basic feature for input to TCD (as this method is feature-neutral). In particular, only those MBHs along the trajectories detected as

Criteria	LOC ($\alpha = 0.9$)	LOC ($\alpha = 0.7$)	LOC ($\alpha = 0.5$)	COV (%)	AODE (%)
Human Action (50 video pairs)					
Mean	0.76	0.89	0.91	61.78	13.58
Median	1.00	1.00	1.00	70.58	11.04
Animal Action (30 video pairs)					
Mean	0.82	0.89	0.93	38.98	12.00
Median	0.93	1.00	1.00	39.46	3.20

Table 3. Common action extraction results of our method.

moving by the initial figure-ground segmentation (Sect. 3) are fed to the BoTW model.

Experimental setting: We set the sampling density of the trajectory tracker [19] as 4 (only every 4th pixel in the x and y direction is taken into account) for the human action clips and 8 for the animal action clips. We discard all trajectories whose lengths are less than 8 frames. For the motion based figure-ground segmentation, ρ in (2) is set as $\rho = 3$. The threshold γ in (11) is set as $\gamma = 0.2$ and the scalar parameter λ_V in (10) is set as $\lambda_V = 50$.

Experiment results: Table 3 presents the quantitative results of the spatial localization performance of our method. It can be seen that our method has an average localization score more than 0.75 even when the threshold α is as high as 0.9. Moreover, the trajectories that are extracted as common action cover more than 60% and 35% of the annotated bounding boxes for the human action dataset and the animal action dataset respectively. The figure is significantly lower for animal actions due to the substantial postural variation in some categories (e.g. the Black Skimmer and Shoebill in example (1) of Fig. 9), compounded by the much more irregular outlines of the animals that are not well fitted by a rectangular box. As for the subtraction of action outliers, our method is able to detect more than 85% and 90% of the action outliers for the human action dataset and the animal action dataset respectively.

In Fig 7, we depict the MR and FAR of our method and the TCD [4] with various settings of η and window lengths. It can be seen that our method can achieve a low MR as well as a low FAR (especially for η between 0.1 and 0.3), significantly outperforming TCD [4]. One reason for the unsatisfactory performance of TCD is that it heavily predicates on the assumption that the common action from two different videos shares the same entries of the built BoTW whereas the temporal action outliers fall into other entries. This is, however, usually not the case when the BoTW is based on low-level features such as the MBH and no learning process is used to discard the uninformative ones. Another important shortcoming of TCD is that it cannot deal with spatial action outliers, i.e., those extraneous actions coexisting with the common action.

Some qualitative results of our method are also depicted in Fig. 8 and Fig. 9. It can be seen that the proposed figure-ground segmentation based on 2D motion cues is able to subtract most of the still background across a wide variety of motion-scene configurations. However, it has poor

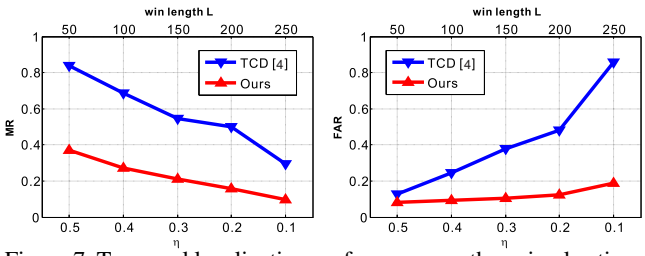

Figure 7. Temporal localization performance on the animal action.

performance when the camera undergoes complex motions; nevertheless, after further co-segmentation, most of the background is finally removed as action outliers (see examples (3), (5), (9) and (10) in Fig. 8). The results shown in Fig. 9 demonstrate that the proposed method is able to spatially and temporally locate the tagged common action. It can be seen from examples (3) and (4) of Fig. 9 that our method succeeds in distinguishing different actions of the same species having nearly the same appearance. Furthermore, some of the common actions can be identified across rather different bird species (example (1) of Fig. 9), ignoring the peculiarities of appearance.

7. Conclusions

We have presented a video co-segmentation framework for common action extraction using dense trajectories. Given a pair of videos that contain a common action, we first perform motion based figure-ground segmentation within each video as a preprocessing step to remove most of the background trajectories. Then, to measure the co-saliency of the trajectories, we design a novel feature descriptor to encode all MBH features along the trajectories and adapt the graph matching technique to impose geometric coherence between the associated cluster matches. Finally, a MRF model is used for segmenting the trajectories into the common action and the action outliers; the data terms are defined by the measured co-saliency and the smoothness terms are defined by the spatiotemporal distance between trajectories. Experiments on our dataset shows that the proposed video co-segmentation framework is effective for common action extraction and opens up new opportunity for video tag information supplementation.

Acknowledgements. This work was partially supported by the Singapore PSF grant 1321202075 and the grant from the National University of Singapore (Suzhou) Research Institute (R-2012-N-002).

References

[1] M. Andriluka, S. Roth, and B. Schiele. Pictorial structures revisited: people detection and articulated pose estimation. In *CVPR*, 2009. 1

[2] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *TPAMI*, 23(11):1222–1239, 2001. 5

[3] T. Brox and J. Malik. Object segmentation by long term analysis of point trajectories. In *ECCV*, 2010. 1

[4] W. Chu, F. Zhou, and F. D. Torre. Unsupervised temporal commonality discovery. In *ECCV*, 2012. 6, 7

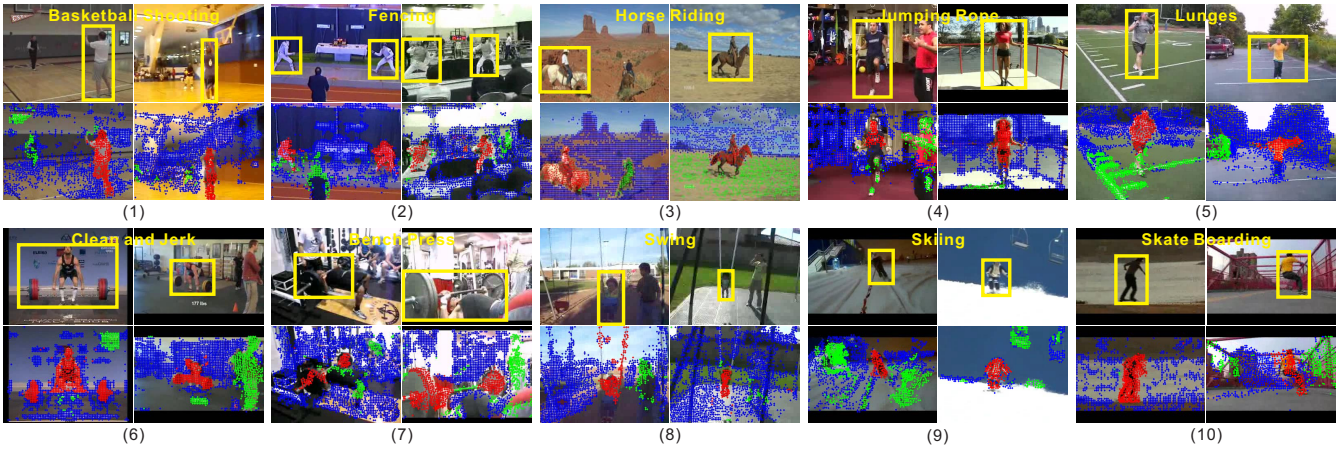

Figure 8. Results of ten video pair examples from the human action dataset. In each example, from top to bottom: two image frames from the pair, and the co-segmentation results. Blue denotes the background trajectories detected in the initial background subtraction step; green denotes the detected action outliers; red denotes the detected common action. The yellow bounding boxes are the given annotations that indicate the interesting regions. The corresponding tags of the videos are overlaid on the top of each example.

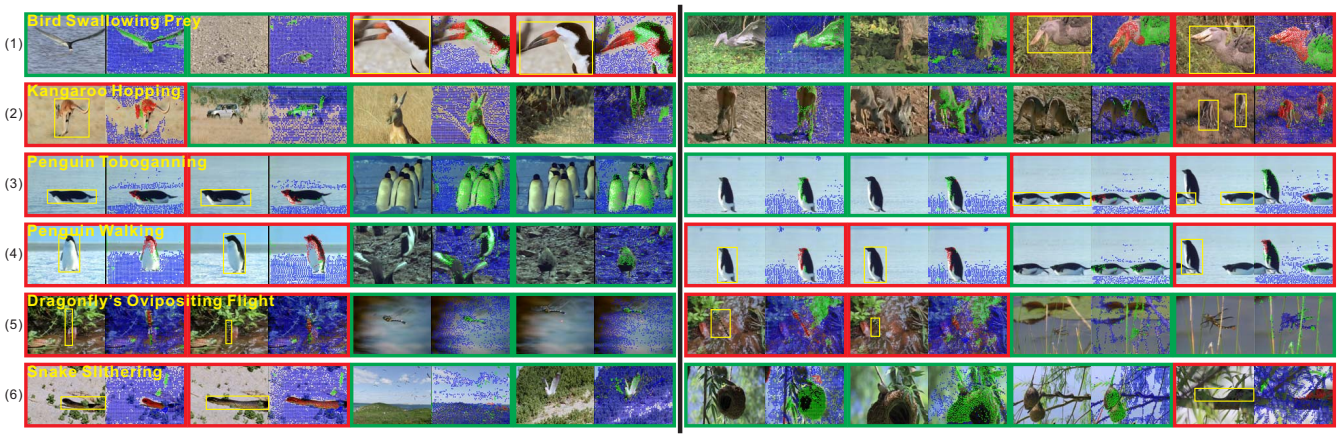

Figure 9. Results of six examples from the animal action dataset, with the same color notation scheme as in Fig. 8. In each example, multiple frames of the two input videos (separated by the bold black line in the middle) are arranged in time order. The active and the non-active frames are bordered in red and green respectively. The corresponding tags are overlaid on the top-left of each example.

[5] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *ECCV*, 2006. 2, 3

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

[7] K. Fragkiadaki and J. Shi. Detection free tracking: Exploiting motion and topology for segmenting and tracking under entanglement. In *CVPR*, 2011. 2

[8] M. Leordeanu and M. Hebert. A spectral technique for correspondence problems using pairwise constraints. In *ICCV*, 2005. 4, 5

[9] H. Li and K. N. Ngan. A co-saliency model of image pairs. *TIP*, 20(12):3365–3375, 2011. 2

[10] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *NIPS*, 2001. 4

[11] P. Ochs and T. Brox. Object segmentation in video: A hierarchical variational approach for turning point trajectories into dense regions. In *ICCV*, 2011. 6

[12] E. Rahtu, J. Kannala, M. Salo, and J. Heikkilä. Segmenting salient objects from images and videos. In *ECCV*, 2010. 1, 2

[13] M. Raptis, I. Kokkinos, and S. Soatto. Discovering discriminative action parts from mid-level video representations. In *CVPR*, 2012. 1, 4, 6

[14] K. K. Reddy and M. Shah. Recognizing 50 human action ctegories of web videos. *Machine Vision and Applications Journal*, 2012. 6

[15] C. Rother, V. Kolmogorov, T. Minka, and A. Blake. Cosegmentation of image pairs by histogram matching - incorporating a global constraint into mrfs. In *CVPR*, 2006. 2

[16] J. C. Rubio, J. Serrat, and A. López. Video co-segmentation. In *ACCV*, 2012. 2, 6

[17] S. Sadanand and J. J. Corso. Action bank: A high-level representation of activity in video. In *ICCV*, 2009. 1

[18] Y. Sheikh, O. Javed, and T. Kanade. Background subtraction for freely moving camera. In *ICCV*, 2009. 2

[19] N. Sundaram, T. Brox, and K. Keutzer. Dense point trajectories by gpu-accelerated large displacement optical flow. In *ECCV*, 2008. 2, 7

[20] F. Tiburzi, M. Escudero, J. Bescós, and J. M. M. Sanchez. A ground truth for motionbased video-object segmentation. In *ICIP*, 2008. 6

[21] A. Toshev, J. Shi, and K. Daniilidis. Image matching via saliency region correspondences. In *CVPR*, 2007. 2

[22] K. N. Tran, I. A. Kakadiaris, and S. H. Shah. Modeling motion of body parts for action recognition. In *BMVC*, 2011. 1

[23] H. Wang, A. Kläser, C. Schmid, and C. Liu. Action recognition by dense trajectories. In *ICCV*, 2009. 2, 3

[24] M. Wang, B. Ni, H. X., and T. Chua. Assistive tagging: a survey of multimedia tagging with human-computer joint exploration. *ACM Computing Surveys*, 44(4), 2012. 1

[25] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *CVPR*, 2011. 4

[26] J. Yan and M. Pollefeys. A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate. In *ECCV*, 2006. 2