# Object Trajectory Proposal via Hierarchical Volume Grouping

Xu Sun[1], Yuantian Wang[1], Tongwei Ren[1,*], Zhi Liu[2], Zheng-Jun Zha[3], and Gangshan Wu[1]
[1] State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China
[2] School of Communication and Information Engineering, Shanghai University, Shanghai, China
[3] University of Science and Technology of China, China
sunx@smail.nju.edu.cn,wangyt@smail.nju.edu.cn,rentw@nju.edu.cn,liuzhi@staff.shu.edu.cn,zhazj@ustc.edu.cn,
gswu@nju.edu.cn

## ABSTRACT

Object trajectory proposal aims to locate category-independent object candidates in videos with a limited number of trajectories, *i.e.*, bounding box sequences. Most existing methods, which derive from combining object proposal with tracking, cannot handle object trajectory proposal effectively due to the lack of comprehensive objectness measurement through analyzing spatio-temporal characteristics over a whole video. In this paper, we propose a novel object trajectory proposal method using hierarchical volume grouping. Specifically, we first represent a given video with hierarchical volumes by mapping hierarchical regions with optical flow. Then, we filter the short volumes and background volumes, and combinatorially group the retained volumes into object candidates. Finally, we rank the object candidates using a multi-modal fusion scoring mechanism, which incorporates both appearance objectness and motion objectness, and generate the bounding boxes of the object candidates with the highest scores as the trajectory proposals. We validated the proposed method on a dataset consisting of 200 videos from ILSVRC2016-VID. The experimental results show that our method is superior to the state-of-the-art object trajectory proposal methods.

## CCS CONCEPTS

• **Artificial intelligence** → **Computer vision**; *Hierarchical representations*;

## KEYWORDS

Object trajectory proposal, hierarchical volume representation, volume combinatorial grouping, multi-modal fusion scoring

**ACM Reference Format:**
Xu Sun[1], Yuantian Wang[1], Tongwei Ren[1,*], Zhi Liu[2], Zheng-Jun Zha[3], and Gangshan Wu[1]. 2018. Object Trajectory Proposal via Hierarchical Volume Grouping. In *ICMR '18: 2018 International Conference on Multimedia Retrieval, June 11–14, 2018, Yokohama, Japan.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3206025.3206059


*ICMR '18, June 11–14, 2018, Yokohama, Japan*

ACM ISBN 978-1-4503-5046-4/18/06…$15.00
https://doi.org/10.1145/3206025.3206059

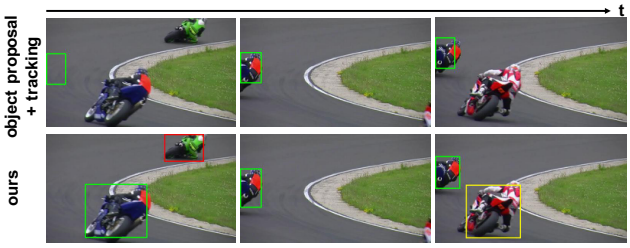


**Figure 1: An example of object trajectory proposal using object proposal and tracking combination and our grouping method.**

## 1 INTRODUCTION

Object trajectory proposal aims to locate category-independent object candidates in videos with a limited number of bounding box sequences named *trajectory* [29]. These generated trajectory proposals provide spatio-temporal characteristics of object candidates, which can be extracted and analyzed in numerous multimedia applications, such as object detection [17], action recognition [34], visual relation detection [28], object segmentation [35] and content-based video retrieval [7].

A primary strategy for object trajectory proposal is to take advantage of the advances in object proposal, which is studied to locate category-independent object candidates in images with a limited number of bounding boxes. Most existing object trajectory proposal methods apply object proposal on one or several selected video frames, and track the generated bounding boxes on other frames to generate trajectory proposals [11, 21, 30, 39]. Nevertheless, it is intractable to automatically select one or few video frames that contain all the objects appearing in a given video. As shown in the top row of Figure 1, two objects are omitted because the object proposal is only applied on the middle frame but these two objects do not appear in this frame. As the state-of-the-art object proposal methods like [9, 43] and object detection methods like [15, 26] can process images in real-time, some works apply object proposal densely or even on all the frames to avoid object omission. Different to specific object detection, such as pedestrian[2, 15], this kind of trajectory proposal methods can easily obtain excessive candidates, because the features of category-independent objects are much more general than those in particular categories. It will lead to high time consumption in object tracking and candidate merging. To make matters worse, these methods only measure objectness on video frames with appearance characteristics, but ignore the characteristics of object

candidates on other modalities, such as motion. Recently, Shang *et al.* [29] improve this strategy by traversing all the video frames with bounded computational cost and partially incorporating object motion in objectness measurement. However, they still conduct objectness measurement independently on each frame instead of generating object candidates through analyzing spatio-temporal characteristics over the whole video.

To overcome these drawbacks, we learn from the advances in object proposal on images. There are two strategies mainly used in object proposal: window scoring and grouping. The former samples sufficient bounding boxes and selects the ones with high objectness scores as proposals, while the latter segments images into regions and merges them into proposals. Considering there exist enormous possible trajectories in a video which cannot be sufficiently sampled, we choose grouping strategy for conducting object trajectory proposal in a whole video. Similar to grouping-based object proposal methods, there are three key issues which should be addressed in grouping-based object trajectory proposal methods: First, how to define a type of basic unit that facilitates grouping to represent various video content? Second, how to group the basic units into object candidates even they have complex compositions? Third, how to score the object candidates with the spatio-temporal characteristics of the video?

In this paper, we propose a novel object trajectory proposal method based on hierarchical volume grouping. Figure 2 shows an overview of the proposed method. Given a video, we first represent it with hierarchical volumes by mapping hierarchical regions with optical flow. Then, we filter the short volumes and the background volumes, and group the retained volumes into object candidates. Finally, we rank the object candidates by fusing appearance objectness and motion objectness, and generate the bounding boxes for the object candidates with the highest scores as the trajectory proposals. A grouping-based method with a similar framework was proposed in [12]. However, it has obvious limitations in addressing the aforementioned key issues of grouping-based object trajectory proposal (refer to Section 2.2) which leads to its unsatisfactory performance (refer to Section 4.3) as compared to that of our method. We validated the proposed method on a dataset consisting of 200 videos from ILSVRC2016-VID [27]. It shows that our method outperforms the state-of-the-art object trajectory proposal methods.

Our contributions mainly include:

- We propose a grouping-based object trajectory proposal method, which generates trajectory proposals by analyzing the spatio-temporal characteristics over a whole given video.
- We present multiple key techniques, namely hierarchical volume representation, volume combinatorial grouping, and multi-modal fusion scoring, which can effectively address the key issues in grouping-based object trajectory proposal.
- We construct a dataset with 200 videos from ILSVRC2016-VID to validate the performance of the proposed method. The experimental results show that our method is superior to the state-of-the-art methods.

# 2 RELATED WORK

## 2.1 Object proposal

The existing object proposal methods can be roughly classified into two categories: window scoring-based methods and grouping-based methods.

Window scoring-based methods measure the objectness of sufficiently sampled bounding boxes and select the bounding boxes with high scores as proposals on RGB images [1, 9, 40, 43] and RGB-D images [19]. Generally speaking, window scoring-based methods are efficient in object proposal, but they easily fail to generate the proposals with high Intersection of Union (IoU) to the groundtruths of objects.

Grouping-based methods segment images into regions and merge them into proposals on RGB images [20, 23, 33] and RGB-D images [36]. In contrast, grouping-based methods can generate accurate proposals against the groundtruths, but they usually have low efficiency because of the time-consuming bottom-up merging.

## 2.2 Object trajectory proposal

Most existing object trajectory methods derive from combining object proposal with tracking. Some methods mainly focuses on salient objects [30, 38] or moving objects [14, 22], while other methods extract objects in a unified objectness strategy [8, 11, 18, 21, 29, 39]. As mentioned above, these methods cannot handle object trajectory proposal effectively due to the lack of comprehensive objectness measurement through analyzing spatio-temporal characteristics over the whole video.

Dan *et al.* [12] first propose a grouping-based object trajectory proposal method using supervoxel clustering. Though Dan's method is similar in overview as compared to our method, it has some obvious limitations in addressing the key issues of grouping-based object trajectory proposal. First, Dan's method uses the supervoxels generated by hierarchical clustering as the basic units in grouping. These supervoxels on the coarsest level may not contain the detailed object portions because of the inaccuracy of clustering. In contrast, our method represents the video with hierarchical volume, which contains all the levels for the subsequent grouping. Second, Dan's method only merges the supervoxels with high similarities in grouping, which cannot group the object parts with different appearances. In contrast, our method uses combinatorial grouping, which can combine the neighboring volumes with significant differences. Third, Dan's method does not provide an effective scoring strategy, which directly treats the clustering result as the proposals. Our method presents a multi-modal fusing scoring mechanism to measure appearance objectness and motion objectness together.

## 2.3 Object tracking

Object tracking aims to estimate the existence and locations of target objects in the subsequent frames. Numerous object tracking methods have been proposed. For example, sparse representation based methods utilize local sparse representations and collaborative representations to determine the target locations [6, 41]; color histogram based methods track target objects via pixel level statistics [10, 25] and edge information [24, 32]; discriminative

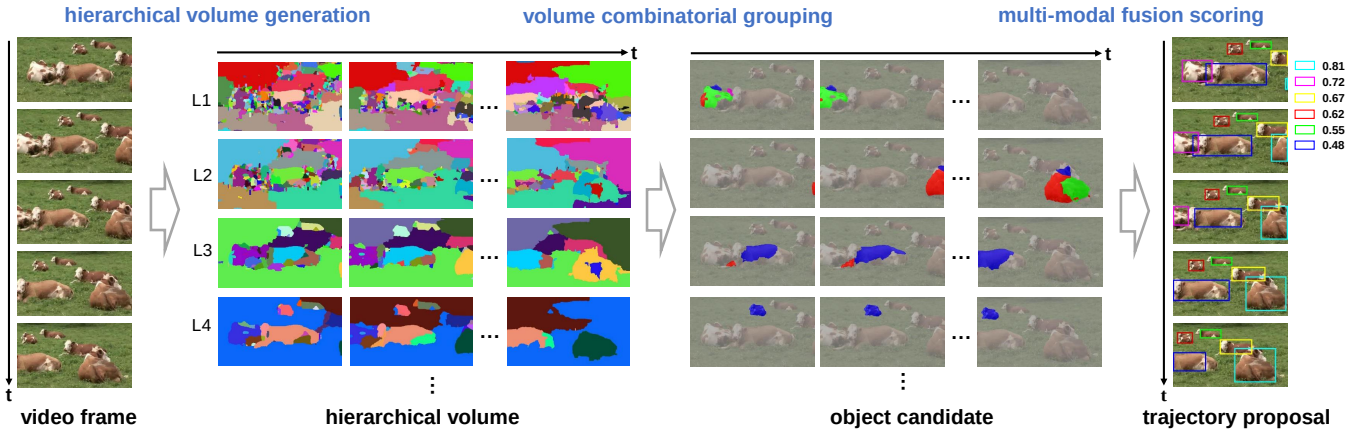


**Figure 2: An overview of the proposed method.**

model based methods utilize learned binary classifiers to segment the target objects from background [4, 5]. Although both tracking and trajectory proposal generate bounding box sequences as their outputs, they are significantly different. First, object tracking requires one or several manually labeled bounding boxes on the first frame for initialization while trajectory proposal searches for object candidates on all the frames automatically. Second, tracking only searches for similar regions through adjacent frames while trajectory proposal requires to measure objectness for trajectory candidates and eliminates those with low objectness. Third, the number of the trajectories extracted by trajectory proposal is much larger than the number of those generated by tracking, which means that trajectory proposal has stricter computational cost limitation in trajectory generation.

## 3 OUR METHOD

### 3.1 Hierarchical volume representation

Hierarchical volume is the basic unit of combinatorial grouping in our method. A volume is a spatio-temporal component within a given video, which is composed of a sequence of ordered regions. In effective hierarchical volume representation, each volume covers the regions with the same content on consecutive frames with coherent boundaries and accurate duration on different levels. Such hierarchical volume representation retains both the principal components and partial details of objects, and helps to generate object candidates.

**Hierarchical region representation of video frame.** To obtain the hierarchical volume representation of a video, we first represent each video frame with hierarchical regions referring to [23]. Specifically, we detect the contours in a given video frame and weight them in the value range of $[0, 1]$ using ultra-metric contour map [3], in which color, brightness and texture gradient are combined to provide indexed contours for hierarchical regions. Based on these contours, we obtain the finest region representation of the video frame. Then, we merge these regions iteratively by removing the contours whose strengths are lower than an incremental threshold from a set of trained thresholds. The regions surrounded with the rest of closed contours in each iteration are constructed as the hierarchical regions.

To each frame $f^t$, assume the number of leaf regions (*i.e.*, the regions on the finest level), is $N_L^t$ and the number of the regions on all the levels is $N_R^t$. We construct a binary matrix $\mathbf{C}^t$ in size of $N_R^t \times N_L^t$ to denote the coverage relationships between hierarchical regions and leaf regions. Here, $\mathbf{C}_{i,j}^t$ equals 1 if the $i$th hierarchical region covers the $j$th leaf region; otherwise, it equals 0.

**Volume generation by region mapping.** We map the hierarchical regions on adjacent video frames to generate hierarchical volumes. Rather than directly searching the most similar region for each region in the subsequent frame, which is very time consuming, we utilize optical flow to establish the mapping relationships between regions on adjacent frames, because optical flow estimation is only required to conduct once for a whole frame and it can be used for the mapping of all the regions in the frame.

We first estimate bidirectional dense optical flow using Deepflow [37], which effectively handles large displacements in videos via dense correspondences matching. To two adjacent video frames $f^t$ and $f^{t+1}$, we construct two matrices $\mathbf{O}^{t\to t+1}$ in size of $N_L^t \times N_L^{t+1}$ and $\mathbf{O}^{t+1\to t}$ in size of $N_L^{t+1} \times N_L^t$ to denote the mapping relationships between the leaf regions in $f^t$ and $f^{t+1}$. Specifically, $\mathbf{O}_{i,j}^{t\to t+1}$ denotes the number of the pixels in the $i$th leaf region in $f^t$ whose corresponding pixels mapped with the optical flow from $f^t$ to $f^{t+1}$ belong to the $j$th leaf region in $f^{t+1}$:

$$\mathbf{O}_{i,j}^{t\to t+1} = |P_i^{t\to t+1} \cap P_j^{t+1}|, \tag{1}$$

where $P_i^{t\to t+1}$ is the set of the pixels in $f^{t+1}$ which are mapped with optical flow from the pixels in the $i$th leaf region in $f^t$; $P_j^{t+1}$ denotes the set of the pixels belonging to the $j$th leaf region in $f^{t+1}$; $|\cdot|$ denotes the cardinality of a set, *i.e.*, the number of pixels within a set. The elements in $\mathbf{O}^{t+1\to t}$ are defined in a similar manner.

Based on the region coverage matrix and the leaf region mapping matrix, we map the hierarchical regions between the adjacent frames. To $f^t$ and $f^{t+1}$, we construct two matrices $\mathbf{M}^{t\to t+1}$ in size of $N_R^t \times N_R^{t+1}$ and $\mathbf{M}^{t+1\to t}$ in size of $N_R^{t+1} \times N_R^t$ to denote the mapping relationships between the hierarchical regions in $f^t$ and

$f^{t+1}$:

$$\mathbf{M}^{t\to t+1} = \mathbf{C}^t\mathbf{O}^{t\to t+1}(\mathbf{C}^{t+1})^{\mathrm{T}}, \quad (2)$$

$$\mathbf{M}^{t+1\to t} = \mathbf{C}^{t+1}\mathbf{O}^{t+1\to t}(\mathbf{C}^{t})^{\mathrm{T}}. \quad (3)$$

where each element $\mathbf{M}^{t\to t+1}_{i,j}$ in $\mathbf{M}^{t\to t+1}$ denotes the number of the pixels in the $i$th region in $f^t$ whose corresponding pixels mapped with the optical flow from $f^t$ to $f^{t+1}$ belong to the $j$th region in $f^{t+1}$. The elements in $\mathbf{M}^{t+1\to t}$ are defined in a similar manner.

We normalize the elements in $\mathbf{M}^{t\to t+1}$ to construct a matrix $\mathbf{\Psi}^{t\to t+1}$ as follows:

$$\mathbf{\Psi}^{t\to t+1}_{i,j} = \mathbf{M}^{t\to t+1}_{i,j}/|Q^t_i|, \quad (4)$$

where $Q^t_i$ denotes the set of the pixels belong to the $i$th hierarchical region in $f^t$; $|\cdot|$ denotes the cardinality of a set. We can construct a matrix $\mathbf{\Psi}^{t+1\to t}$ from $\mathbf{M}^{t+1\to t}$ in a similar manner.

To each hierarchical region in $f^t$, we search its mapping region in $f^{t+1}$ by checking the elements in the $i$th row in $\mathbf{\Psi}^{t\to t+1}$ and the $i$th column in $\mathbf{\Psi}^{t+1\to t}$. If both $\mathbf{\Psi}^{t\to t+1}_{i,j}$ and $\mathbf{\Psi}^{t+1\to t}_{j,i}$ are larger than a predefined threshold, which equals 0.5 in our experiments, we consider the $i$th region in $f^t$ is mapped to the $j$th region in $f^{t+1}$ successfully and merge them into a volume. If a region in $f^t$ is mapped to more than one regions in $f^{t+1}$, we select the one with the highest mapping score, *i.e.*, the sum of $\mathbf{\Psi}^{t\to t+1}_{i,j}$ and $\mathbf{\Psi}^{t+1\to t}_{j,i}$. We repeat the above procedure from the first frame to the last one, and generate hierarchical volumes by merging the mapped regions.

Note here, we map the hierarchical regions and retain the volumes on all the levels for the subsequent grouping, *i.e.*, one volume generated by our method may contain another volume or its portions. In contrast, supervoxels in [12] are generated by hierarchical clustering of single level superpixels, which only consist of the coarsest regions and have no overlap between each other.

There are two advantages of our hierarchical volume representation as compared to hierarchical supervoxel in [12]. On the one hand, it is difficult to find appropriate relationships when directly mapping the regions in the adjacent frames on superpixel level. In fact, we cannot generate the volumes with long lengths (the length of a volume denotes the number of the regions belonging to the volume) by mapping the leaf regions in the experiments unless we relax the mapping threshold to a small value. It leads to the generated supervoxels with low cohesion and prevents the effectiveness of the supervoxel representation in [12]. On the other hand, both the generation of hierarchical volume and supervoxel are only based on low-level features, which inevitably brings in inaccuracy. If only using the supervoxels composed of the coarsest regions [12], the detailed object portions lost in clustering cannot be obtained in grouping. It will further cause the inaccuracy of generating trajectory proposals. As comparison, our method generates the volumes in different levels and retains most of them, which will provide more comprehensive candidates in grouping.

**Broken volume connection.** Our hierarchical volume representation maps the regions based on optical flow, but optical flow only represents the low-level similarity between pixels in adjacent frames. It is easy to generate broken volumes because optical flow is sensitive to partial occlusion and illumination variation. These broken volumes increase the complexity of grouping them into the object candidates. Hence, we connect the volumes, which may represent the same content, based on their high-level appearance similarities.

Assume $v_i$ and $v_j$ are two volumes, and $v_i$ ends before $v_j$ starts. We track the bounding box of the region, which belongs to $v_i$ in its end frame, using kernelized correlation filter tracker [16]. If the tracked bounding box in the start frame of $v_j$ has large enough IoU to the bounding box of the region belonging to $v_j$ in the same frame, we consider that $v_i$ and $v_j$ can be connected.

In our implementation, we filter the volumes with extremely short lengths for efficiency, which are less than 10 frames in our experiments, and sort all the retained volumes into two queues $\boldsymbol{q_s}$ and $\boldsymbol{q_e}$ based on their start frames and end frames. To each volume $v$ in $\boldsymbol{q_e}$, we check whether exist volumes starting in a short temporal interval after $v$ ends, which equals 4 frames in our experiments. If so, we greedily connect $v$ to these volumes by tracking, *i.e.*, we only connect a volume to the first volume which can be connected to it. The IoU threshold for connection in our experiments is set to 0.6. Because the number of volumes that start in a short temporal interval after each volume can be treated as a constant and only once traversal is required for $\boldsymbol{q_e}$, the time complexity of volume connect is $O(N_V \log N_V)$, here $N_V$ is the number of volumes after filtering.

## 3.2 Volume combinatorial grouping

**Adaptive short volume filtering.** Considering most objects are visible for relatively long durations in videos, the volumes with short lengths are usually useless for generating object candidates. Preliminary filtering of such short volumes can decrease the number of volumes and effectively reduce the computational complexity of volume grouping.

Considering the variation of video lengths, we use an adaptive threshold in short volume filtering. On the one hand, we consider that the volumes contributive to object candidate generation should be visible for more than a certain duration, for example, one second (20 frames) in our experiments. On the other hand, to avoid too strict filtering for short videos, we require that the length of the filtered volumes should be no larger than a certain ratio of the length of the whole video, such as 20% in our experiments. Hence, the final threshold $\kappa$ for short volume filtering is calculated as:

$$\kappa = \min(20, 0.2 \cdot ||V||), \quad (5)$$

where $||V||$ denotes the length of the whole video.

**Background volume elimination.** The volumes containing background content usually have long lengths because the stability in background appearance helps to region mapping in hierarchical volume generation. These background volumes cannot be removed in short volume filtering, but they hamper trajectory proposal because they are easy to be grouped with other volumes in combinatorial grouping and generate many object candidates covering no objects.

To eliminate the background volumes, we calculate the boundary connectivity of each volume based on that of the regions belonging to this volume. Assume $v_i$ is a volume composed of a region sequence $\{r_i^{t_k}\}_{k=1,\cdots,||v_i||}$. Here, $v_i$ starts in frame $f^{t_1}$ and ends in frame $f^{t_{||v_i||}}$, and $||v_i||$ denotes the length of $v_i$. To each region

$r_i^{t_k}$, we calculate its boundary connectivity $B(r_i^{t_k})$ by measuring the extent of its connection to frame boundary [42] as follows:

$$B(r_i^{t_k}) = \frac{|C(r_i^{t_k})|}{\sqrt{|r_i^{t_k}|}}, \tag{6}$$

where $C(r_i^{t_k})$ denotes the set of pixels which are on both the boundaries of $r_i^{t_k}$ and frame $f^{t_k}$; $|\cdot|$ denotes the cardinality of a set, *i.e.*, the number of pixels within a set or a region.

Based on the boundary connectivity of all the regions belonging to $v_i$, we calculate the boundary connectivity of $v_i$ as follows:

$$B(v_i) = \frac{|\{r_i^{t_k} | B(r_i^{t_k}) > \theta, k = 1, \cdots, ||v_i||\}|}{||v_i||}, \tag{7}$$

where $\theta$ is a boundary connectivity threshold, which equals 1 in our experiments referring to [42].

To avoid eliminating the volumes covering the objects that appear in frame boundaries, we use a high threshold in background volume elimination. In our experiments, only the volumes whose boundary connectivity is larger than 0.9 will be eliminated as background.

**Candidate generation by volume grouping.** Based on the retained volumes, we generate the object candidates by grouping these volumes. Most exiting methods conduct grouping based on volume similarity, such as [12]. However, such grouping strategies usually fail in generating object candidates effectively, because the parts of an object usually have significant differences in appearance and/or motion. Inspired by [23], we adopt a combinatorial grouping strategy in volume grouping, which only measures the neighborhood relationships between volumes while taking no account of volume similarity.

Considering the neighborhood relationship between two volumes is more complex than that between two regions, *e.g.*, two volumes may be overlapping, adjacent and disjointed in different frames, we relax the combination condition in volume grouping from adjacent in [23] to overlapping or adjacent, in order to generate sufficient object candidates. Specifically, we first construct a binary matrix $\mathbf{A}^t$ in size of $N_L^t \times N_L^t$ to represent the neighborhood relationship between the leaf regions in frame $f^t$, in which $\mathbf{A}_{i,j}^t$ equals 1 if the $i$th leaf region and the $j$th leaf region in $f^t$ are adjacent and 0 otherwise. Then, we represent the neighborhood relationships between the hierarchical regions in $f^t$ by a matrix $\mathbf{\Lambda}^t$ in size of $N_R^t \times N_R^t$:

$$\mathbf{\Lambda}^t = \mathbf{C}^t \mathbf{A}^t (\mathbf{C}^t)^{\mathrm{T}}, \tag{8}$$

where $\mathbf{C}^t$ is the coverage relationship matrix defined in Section 3.1. Here, $\mathbf{\Lambda}_{i,j}^t$ is larger than 0 if the $i$th region and the $j$th region in $f^t$ are overlapping or adjacent, and $\mathbf{\Lambda}_{i,j}^t$ equals 0 if they are disjointed.

Assuming that two volumes $v_i$ and $v_j$ appear in a video with the common duration from $f^{t_1}$ to $f^{t_K}$, we measure the neighborhood relationship between these two volumes as follows:

$$\Theta(v_i, v_j) = \frac{\sum_{t=t_1}^{t_K} \eta(r_i^t, r_j^t)}{\min(||v_i||, ||v_j||)}, \tag{9}$$

where $r_i^t$ and $r_j^t$ are the regions belonging to $v_i$ and $v_j$ in $f^t$, respectively; $\eta(,)$ equals 1 if two regions are overlapping or adjacent in $f^t$ and 0 otherwise, which can refer to $\mathbf{\Lambda}^t$ in Eq. (8); $||\cdot||$ denotes the length of a volume. If $\Theta(v_i, v_j)$ is larger than a threshold, which equals 0.3 in our experiments, $v_i$ and $v_j$ are allowed to group.

Based on the constraint in Eq. (9), we group the volumes into the candidates containing from one to four volumes. Similar to [23], we constrain the number of object candidates using Pareto front optimization [13] to avoid the combinatorial explosion. In our experiments, volume grouping will terminate if all volume combinations are exhausted or the number of object candidates reaches 10,000.

## 3.3 Multi-modal fusion scoring

We measure the objectness of the object candidates, and select the ones with high scores to generate the final trajectory proposal. In objectness measurement, we analyze both appearance objectness and motion objectness of each candidate. The former has been widely used in image proposal [9, 43], and the latter has been validated its effectiveness in trajectory proposal [29].

As for appearance objectness, we adopt the scoring model provided in [23], which is a regressor trained on multi-modal features including shape, edges, size and location. To each candidate $c_k$, which appears from frame $f^{t_1}$ to $f^{t_K}$, we measure the appearance objectness of the grouped regions belonging to $c_k$ on each frame, and calculate the appearance objectness score of $c_k$ as follows:

$$s^A(c_k) = \frac{\sum_{t=t_1}^{t_K} s^A(g_k^t)}{||c_k||}, \tag{10}$$

where $s^A(\cdot)$ denotes appearance objectness score; $g_k^t$ denotes the grouped regions belonging to $c_k$ in frame $f^t$; $||\cdot||$ denotes the length of a candidate, *i.e.*, the number of the frames that the candidate appears in.

As for motion objectness, we measure the motion contrast of an object candidate to background. We calculate the average motion strength of all the regions belonging to background volumes (refer to Section 3.2) on each frame as the motion strength of background. If no background volume is detected in some frame, we use the average motion strength of the whole frame to instead. To each candidate $c_k$, which appears from frame $f^{t_1}$ to $f^{t_K}$, we calculate the distance between the average motion strength of the grouped regions belonging to $c_k$ and that of background, and calculate the motion objectness score of $c_k$ as follows:

$$s^M(c_k) = \frac{\sum_{t=t_1}^{t_K} |m_k^t - m_B^t|}{||c_k||}, \tag{11}$$

where $s^M(.)$ denotes motion objectness score; $m_k^t$ denotes the average motion strength of the grouped regions belonging to $c_k$ in frame $f^t$; $m_B^t$ denotes the average motion strength of background in frame $f^t$.

We normalize the appearance objectness score and the motion objectness score of each candidate $c_k$ to the value range of $[0, 1]$, and fuse them with linear combination as the total objectness score of $c_k$:

$$s(c_k) = \lambda \cdot s^A(c_k) + (1 - \lambda) \cdot s^M(c_k), \tag{12}$$

where $\lambda$ is a weight parameter, which equals to 0.7 in our experiments to handle the inaccuracy of motion contrast in motion objectness measurement.

We rank all the candidates according to their objectness scores, generate the bounding boxes of the top selected candidates as the trajectory proposals, and preferentially return the proposals composed with less volumes.

# 4 EXPERIMENTS

## 4.1 Dataset and experiment settings

We validated the performance of the proposed method on a dataset consisting of 200 videos, which are randomly selected from ILSVRC2016-VID [27]. These videos have various content and contain 3.26 objects on average. The average object number of our dataset is larger than the one of the whole ILSVRC2016-VID dataset, which is only 2.06. The manually labeled object trajectories in ILSVRC2016-VID were used as the groundtruths in our experiments. To the best of our knowledge, it is the largest dataset for object trajectory proposal.

We used the evaluation criteria proposed in [29], namely *mean Trajectory IoU* (mT-IoU) and *recall*, on the top 500 trajectory proposals for each video. These criteria are defined as follows:

$$mT\text{-}IoU = \frac{1}{N_V}\sum_{k=1}^{N_V}\frac{1}{N_G^k}\sum_{j=1}^{N_G^k}\max_i\{T\text{-}IoU_{i,j}\}, \tag{13}$$

$$recall = \frac{1}{N_V}\sum_{k=1}^{N_V}\frac{1}{N_G^k}\sum_{j=1}^{N_G^k}\delta\left(\max_i\{T\text{-}IoU_{i,j}\}-\tau\right), \tag{14}$$

where $T\text{-}IoU_{i,j}$ denotes the trajectory IoU between a trajectory proposal $\mathcal{T}_i$ and a trajectory groundtruth $\mathcal{G}_j$; $N_V$ is the number of videos; $N_G^k$ is the number of groundtruths in the $k$th video; $\delta$ is a function that outputs 1 if the input is positive, and 0 otherwise; $\tau$ is a threshold, which equals to 0.5 in our experiments.

All the experiments were conducted on a computer with i7 3.5GHz CPU and 32GB memory. For all the methods in comparison, we used their default settings suggested by the authors.

## 4.2 Component analysis

There are several key techniques incorporated in our method: region mapping, volume representation, volume filtering and candidate scoring. We validated their influences to the performance of our method.

**Region mapping.** We use bidirectional optical flow in region mapping to construct the mapping relationship between the regions in adjacent frames (refer to Section 3.1), which has high computational cost. To evaluate the necessity of bidirectional optical flow, we generate a baseline only using forward optical flow (FOF), which requires half computational cost in optical flow estimation as compared to our method.

Figure 3 shows the performance of our method using different optical flow in region mapping. We can see that the performance declines on both mT-IoU and recall when replacing bidirectional optical flow with forward optical flow. The reason is that optical flow estimation only focuses on pixel-level similarity measurement even using global consistency constraint, which is sensitive to variable video content. Once using bidirectional optical flow, we can diminish the influence of optical flow inaccuracy and map the regions more effectively.

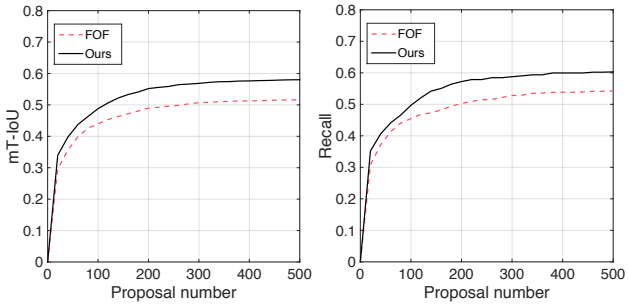


**Figure 3: Evaluation of our method using different optical flow in region mapping on mT-IoU and recall.**

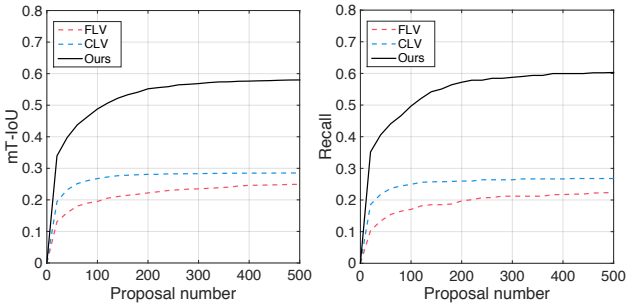


**Figure 4: Evaluation of our method using different volume representation on mT-IoU and recall.**

**Volume representation.** We represent video content with hierarchical volumes on all the levels, *i.e.*, one volume may contain another volume or its portions (refer to Section 3.1). Such a volume representation can represent both the principal components and the details of objects. To illustrate the superiority of our hierarchical volume representation, we generate two baselines: using the finest level volumes (FLV), which are generated by the finest level regions on each video frame, and using the coarsest level volumes (CLV), which merges the regions on each video frame until the number of regions is no more than 30% of the one of the finest regions and maps these regions to generate the volumes.

Figure 4 shows the performance of our method using different volume representations. We can see that both FLV and CLV have dramatic drops on mT-IoU and recall. For example, the mT-IoU and recall values obtained by CLV on 500 proposals are only half of the ones obtained by our method. It shows that only the coarsest level volumes, which represent the principal components of objects, cannot effectively support trajectory proposal because all the object details are lost. Moreover, FLV has worse performance than CLV, which is caused by two reasons: 1) The leaf regions are difficult to be mapped into long-duration finest level volumes due to the inaccuracy of optical flow, which increases the complexity of volume grouping. 2) Our method only groups at most four volumes into a candidate to avoid combinatorial explosion, which leads to the candidate grouped by the finest level volumes are usually seriously incomplete against objects. It validates the effectiveness of hierarchical volume for video content representation.

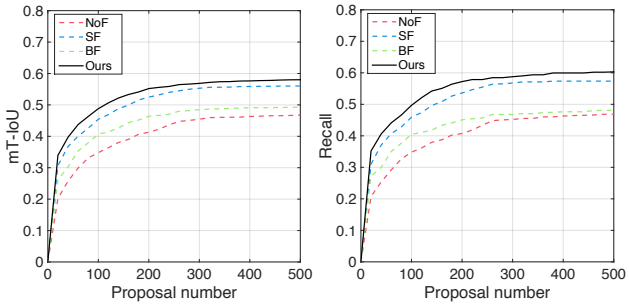


**Figure 5: Evaluation of our method using different volume filtering strategies on mT-IoU and recall.**

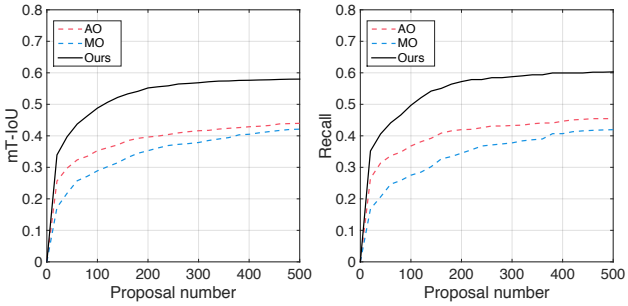


**Figure 6: Evaluation of our method using different candidate scoring mechanisms on mT-IoU and recall.**

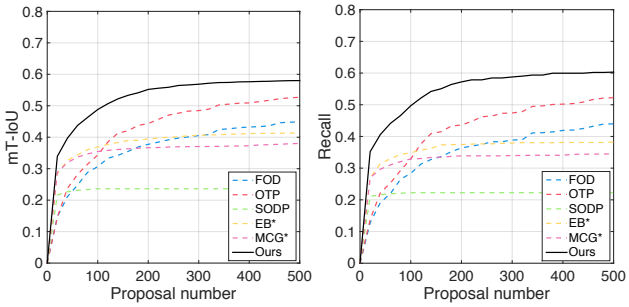


**Figure 7: Evaluation of different trajectory proposal methods on mT-IoU and recall.**

**Volume filtering.** We consider the short volumes and the background volumes are useless in object representation, and filter them to reduce the complexity of volume grouping (refer to Section 3.2). To illustrate the necessity of volume filtering, we generate three baselines: without short volume filtering or background volume filtering (NoF), only filtering short volumes (SF), and only filtering background volumes (BF).

Figure 5 shows the performance of our method using different volume filtering strategies. We can see that the performance of NoF is worse than that using other volume filtering strategies. Note here, in broken volume connection (refer to Section 3.1), we have filtered the volumes whose lengths are less than 10 frames for efficiency. It means that NoF actually uses the volume filtering strategy by removing the extremely short volumes (less than 10 frames) rather than filters nothing. Once increasing the filtering threshold on volume length from 10 frames to 20 frames, the performance of SF is improved significantly as compared to NoF. It shows the effectiveness of short volume filtering. Moreover, we can see that the performance will be improved when filtering background volumes, no matter on NoF (*i.e.*, BF) and on SF (*i.e.*, Ours). It shows that background volume elimination is beneficial to trajectory proposal.

**Candidate scoring mechanism.** We use a multi-modal fusion scoring mechanism in ranking object candidates, which fuses both appearance objectness and motion objectness (refer to Section 3.3). To validate the effectiveness of our multi-modal fusion scoring mechanism, we generate two baselines: scoring with appearance objectness (AO), and scoring with motion objectness (MO).

Figure 6 shows the performance of our method using different scoring strategies. We can see that the performance of both AO and MO is significantly worse than that of our method on mT-IoU and recall, which illustrates the effectiveness of fusing appearance objectness and motion objectness in scoring. Moreover, the performance of MO is only slightly worse than that of AO. Because objects usually have different motions against background, motion objectness contributes to trajectory proposal especially when object appearance is complex.

## 4.3 Comparison with the state-of-the-arts

To validate the performance of our method, we compared it with three state-of-the-art video object proposal methods, namely free object discovery (FOD) [11], object trajectory proposal (OTP) [29], and spatio-temporal object detection proposal (SODP) [12]. In the same way as [29], we generate two baselines EB* and MCG*, which generate object proposals on the middle frame of each video and track the proposals with KCF tracker [16] bidirectionally to generate trajectory proposals.

Figure 7 shows the performance of our method and five compared methods. We have: 1) Our method is superior to all the compared methods on both mT-IoU and recall, which illustrates the effectiveness of our method. 2) Though SODP has a similar framework to our method, its performance is dramatically worse than that of our method and other methods. It shows the effectiveness of the key techniques in our methods, namely hierarchical volume representation, volume combinatorial grouping, and multi-modal fusion scoring. 3) The performance of EB* and MCG* is worse than other methods except SODP, though EB and MCG are the best object proposal methods on images. It is because they cannot propose the object trajectories which do not appear on the middle frames of videos (or any other selected frames as the start of trajectory generation). It illustrates the drawbacks of object trajectory proposal by combining image object proposal and tracking. 4) Though FOD and OTP aim to address the start frame problem by traversing all the video frames, their performance is still worse than ours. It is because they cannot effectively measure objectness based on spatio-temporal characteristics, and propose objects independently on each frame.

Figure 8 illustrates several examples of our results. In each video example, the trajectory proposal with the highest mT-IoU to each trajectory groundtruth from the top 500 proposals is shown. It

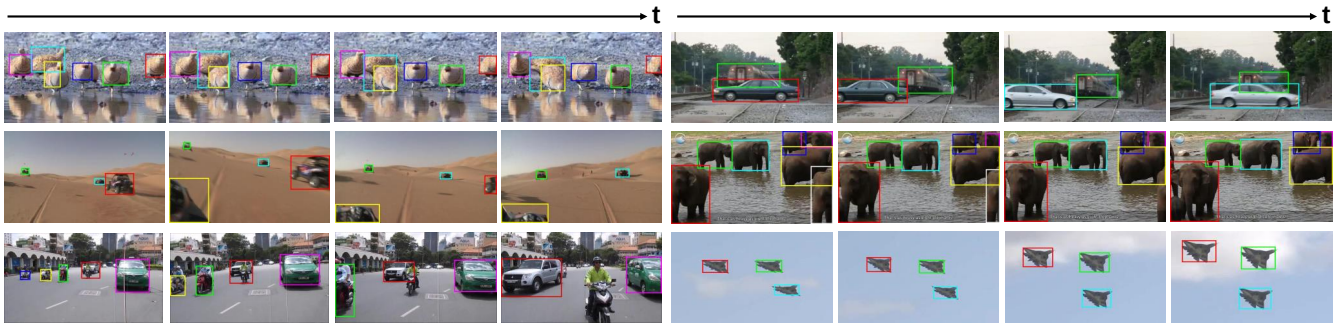


**Figure 8: Qualitative examples of object trajectory proposal using our method. In each video example, the bounding boxes with the same color indicate a trajectory proposal, and the trajectory proposal with the highest mT-IoU to each trajectory groundtruth from the top 500 proposals is shown.**

**Table 1: Comparison with the state-of-the-art methods and the baselines on time cost per frame.**

| Method | Language | Time (s) |
|---|---|---|
| FOD | C++ & Python | 3.4 |
| OTP | C++ & Python | 3.4 |
| SODP | C++ & Matlab & Python | 6.7 |
| EB* | C++ & Matlab | 5.5 |
| MCG* | C++ & Matlab | 3.5 |
| Ours | C++ & Matlab | 7.5 |

shows that our method can handle the videos with multiple objects in various content.

We also validated the efficiency of our method. Table 1 shows the time cost per frame of our method and the compared methods. It shows that our method is comparable to other grouping-based object trajectory proposal methods, such as SODP. The most time consuming component in our method is bidirectional optical flow estimation. It can be accelerated by GPU [31] to make our method more efficient.

### 4.4 Discussion

In the experiments, we found some limitations of our method. As shown in the top row of Figure 9, one goat is omitted because it is occluded by two other goats with similar appearances, which leads to serious inaccuracy in volume representation. Another failure example shown in the bottom row of Figure 9. There are many motorcycles and cars with small sizes are omitted. It is because these small objects are composed of multiple small volumes, which are easily filtered in volume grouping. Moreover, as shown in Table 1, the time costs of our method and other existing trajectory proposal methods are still high, which prevents their application in real-time conditions.

## 5 CONCLUSION

In this paper, we proposed an object trajectory proposal method based on hierarchical volume grouping. Three techniques, namely hierarchical volume representation, volume combinatorial grouping and multi-modal fusion scoring, were presented to address the key

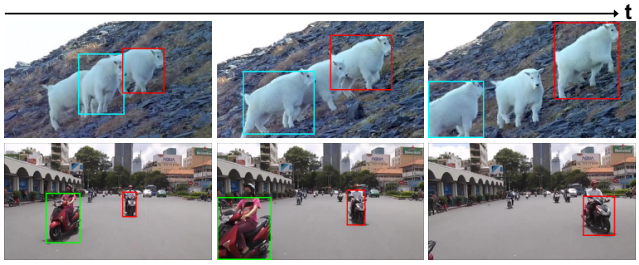


**Figure 9: Our failure examples influenced by object occlusion and small objects.**

issues in grouping-based object trajectory proposal. Benefiting from measuring objectness through analyzing spatio-temporal characteristics over a whole video, our method can effectively generate the trajectory proposals on the videos with multiple objects in various content. We constructed a dataset consisting of 200 videos from ILSVRC2016-VID dataset. The experimental results on the dataset show that our method outperforms the state-of-the-art object trajectory proposal methods.

## 6 ACKNOWLEDGEMENTS

This work is supported by National Science Foundation of China (61321491, 61771301, 61202320), and Collaborative Innovation Center of Novel Software Technology and Industrialization.

## REFERENCES

[1] Bogdan Alexe, Thomas Deselaers, and Vittorio Ferrari. 2012. Measuring the Objectness of Image Windows. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 34 (2012), 2189–2202.

[2] Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele. 2008. People-tracking-by-detection and people-detection-by-tracking. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. 1–8.

[3] P Arbelaez. 2006. Boundary Extraction in Natural Images Using Ultrametric Contour Maps. In *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW '06. Conference on*. 182–182.

[4] Shai Avidan. 2007. Ensemble Tracking. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 29, 2 (2007), 261–271.

[5] Boris Babenko, Ming Hsuan Yang, and Serge Belongie. 2011. Robust Object Tracking with Online Multiple Instance Learning. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 33, 8 (2011), 1619.

[6] Chenglong Bao, Yi Wu, Haibin Ling, and Hui Ji. 2012. Real time robust L1 tracker using accelerated proximal gradient approach. In *IEEE Conference on Computer Vision and Pattern Recognition*. 1830–1837.

[7] Rachid Benmokhtar and Benoit Huet. 2007. Multi-level Fusion for Semantic Video Content Indexing and Retrieval.. In *International Workshop on Adaptive Multimedia Retrieval*. 160–169.

[8] Michael Van Den Bergh, Gemma Roig, Xavier Boix, Santiago Manen, and Luc Van Gool. 2013. Online Video SEEDS for Temporal Window Objectness. In *IEEE International Conference on Computer Vision*. 377–384.

[9] Ming-Ming Cheng, Ziming Zhang, Wen-Yan Lin, and Philip H. S. Torr. 2014. BING: Binarized Normed Gradients for Objectness Estimation at 300fps. In *IEEE Conference on Computer Vision and Pattern Recognition*. 3286–3293.

[10] Dorin Comaniciu, Visvanathan Ramesh, and Peter Meer. 2003. Kernel-Based Object Tracking. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 25, 5 (2003), 564–575.

[11] Giovanni Cuffaro, Federico Becattini, Claudio Baecchi, Lorenzo Seidenari, and Alberto Del Bimbo. 2016. Segmentation Free Object Discovery in Video. In *European Conference on Computer Vision workshops*. 25–31.

[12] Oneata Dan, Jerome Revaud, Jakob Verbeek, and Cordelia Schmid. 2014. Spatio-temporal Object Detection Proposals. In *European Conference on Computer Vision*. 737–752.

[13] Matthias Ehrgott. 2007. *Multicriteria Optimization*. Springer-Verlag New York, Inc. 222–231 pages.

[14] Katerina Fragkiadaki, Pablo Arbelaez, Panna Felsen, and Jitendra Malik. 2015. Learning to segment moving objects in videos. In *IEEE Conference on Computer Vision and Pattern Recognition*. 4083–4090.

[15] L. Van Gool, M. Mathias, R. Timofte, and R. Benenson. 2012. Pedestrian detection at 100 frames per second. In *Computer Vision and Pattern Recognition*. 2903–2910.

[16] JoÃčo F. Henriques, Caseiro Rui, Pedro Martins, and Jorge Batista. 2015. High-Speed Tracking with Kernelized Correlation Filters. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 37, 3 (2015), 583–596.

[17] Kai Kang, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. 2016. Object Detection from Video Tubelets with Convolutional Neural Networks. In *IEEE Conference on Computer Vision and Pattern Recognition*. 817–825.

[18] Jianwu Li, Tianfei Zhou, and Yao Lu. 2017. Learning to generate video object segment proposals. In *IEEE International Conference on Multimedia and Expo*. 787–792.

[19] Jing Liu, Tongwei Ren, Yuantian Wang, Sheng Hua Zhong, Jia Bei, and Shengchao Chen. 2016. Object proposal on RGB-D images via elastic edge boxes. *Neurocomputing* 236 (2016).

[20] Santiago Manen, Matthieu Guillaumin, and Luc J. Van Gool. 2013. Prime Object Proposals with Randomized Prim's Algorithm. In *IEEE International Conference on Computer Vision*. 2536–2543.

[21] Liang Peng and Xiaojun Qi. 2016. Temporal objectness: Model-free learning of object proposals in video. In *IEEE International Conference on Image Processing*. 3663–3667.

[22] Federico Perazzi, Oliver Wang, Markus Gross, and Alexander Sorkine-Hornung. 2015. Fully Connected Object Proposals for Video Segmentation. In *IEEE International Conference on Computer Vision*. 3227–3234.

[23] Jordi Ponttuset, Pablo Arbelaez, Jonathan Barron, Ferran Marques, and Jitendra Malik. 2016. Multiscale Combinatorial Grouping for Image Segmentation and Object Proposal Generation. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 39, 1 (2016), 128–140.

[24] Fatih Porikli. 2005. Integral Histogram: A Fast Way To Extract Histograms in Cartesian Spaces. In *IEEE Conference on Computer Vision and Pattern Recognition*. 829–836.

[25] P. PÃĺrez, C. Hue, J. Vermaak, and M. Gangnet. 2002. Color-Based Probabilistic Tracking. In *European Conference on Computer Vision*. 661–675.

[26] Joseph Redmon and Ali Farhadi. 2016. YOLO9000: Better, Faster, Stronger. (2016), 6517–6525.

[27] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, and Michael Bernstein. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision* 115, 3 (2015), 211–252.

[28] Xindi Shang, Tongwei Ren, Jingfan Guo, Hanwang Zhang, and Tat-Seng Chua. 2017. Video Visual Relation Detection. In *ACM International Conference on Multimedia*. Mountain View, CA USA.

[29] Xindi Shang, Tongwei Ren, Hanwang Zhang, Gangshan Wu, and Tat Seng Chua. 2017. Object trajectory proposal. In *IEEE International Conference on Multimedia and Expo*. 331–336.

[30] Gilad Sharir and Tinne Tuytelaars. 2012. Video object proposals. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 9–14.

[31] Narayanan Sundaram, Thomas Brox, and Kurt Keutzer. 2010. Dense point trajectories by GPU-accelerated large displacement optical flow. In *European Conference on Computer Vision*. 438–451.

[32] Feng Tang, S Brennan, Qi Zhao, and Hai Tao. 2007. Co-Tracking Using Semi-Supervised Support Vector Machines. In *IEEE International Conference on Computer Vision*. 1–8.

[33] Jasper R. R. Uijlings, Koen E. A. van de Sande, Theo Gevers, and Arnold W. M. Smeulders. 2013. Selective Search for Object Recognition. *International Journal of Computer Vision* 104, 2 (2013), 154–171.

[34] Heng Wang, A. Klaser, C. Schmid, and Cheng Lin Liu. 2011. Action recognition by dense trajectories. In *IEEE Conference on Computer Vision and Pattern Recognition*. 3169–3176.

[35] Wenguan Wang, Jianbing Shen, and F Porikli. 2015. Saliency-aware geodesic video object segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*. 3395–3402.

[36] Yuantian Wang, Lei Huang, Tongwei Ren, Sheng-Hua Zhong, Yan Liu, and Gangshan Wu. 2017. Object Proposal via Depth Connectivity Constrained Grouping. In *Pacific-Rim Conference on Multimedia*. 1–10.

[37] Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui, and Cordelia Schmid. 2014. DeepFlow: Large Displacement Optical Flow with Deep Matching. In *IEEE International Conference on Computer Vision*. 1385–1392.

[38] Fanyi Xiao and Jae Lee Yong. 2016. Track and Segment: An Iterative Unsupervised Approach for Video Object Proposals. In *IEEE Conference on Computer Vision and Pattern Recognition*. 933–942.

[39] Jae Lee Yong, Jaechul Kim, and Kristen Grauman. 2011. Key-segments for video object segmentation. In *International Conference on Computer Vision*. 1995–2002.

[40] Ziming Zhang, Yun Liu, Xi Chen, Yanjun Zhu, Ming Ming Cheng, Venkatesh Saligrama, and Philip H. S. Torr. 2015. Sequential Optimization for Efficient High-Quality Object Proposal Generation. *IEEE Transactions on Pattern Analysis & Machine Intelligence* PP, 99 (2015), 1–1.

[41] W. Zhong, H. Lu, and M. H. Yang. 2014. Robust object tracking via sparse collaborative appearance model. *IEEE Transactions onImage Processing* 23, 5 (2014), 2356.

[42] Wangjiang Zhu, Shuang Liang, Yichen Wei, and Jian Sun. 2014. Saliency Optimization from Robust Background Detection. In *Computer Vision and Pattern Recognition*. 2814–2821.

[43] C. Lawrence Zitnick and Piotr Dollár. 2014. Edge Boxes: Locating Object Proposals from Edges. In *European Conference on Computer Vision*. 391–405.