Efficient Object Localization and Segmentation in Weakly Labeled Videos

Mrigank Rochan and Yang Wang

Department of Computer Science, University of Manitoba, Canada
{mrochan, ywang}@cs.umanitoba.ca

Abstract. In this paper, we tackle the problem of efficiently segmenting objects in weakly labeled videos. Internet videos (e.g., YouTube) are often associated with a semantic tag describing the main object within the video. However, this tag does not provide any spatial or temporal information about the object within the video. So these videos are weakly labeled. We propose a novel and efficient approach to localize the object of interest within the video and perform pixel-level segmentation. Given a video with an object tag, our proposed method automatically localizes the object and segments it from the background in each frame of the video. Our method combines object appearance modeling and temporal consistency among frames in a principled framework. Our method does not require user inputs or object detectors, so it can be potentially applied to videos of any object categories. We evaluate our method on a dataset consisting of more than 100 video shots of 10 different object categories. Our experimental results show that our method outperforms other baseline approaches.

1 Introduction

Today we have access to an enormous amount of video content through video sharing websites like YouTube. These videos are often associated with textual descriptions, such as tags. These tags are created by users to provide some information about the visual content (e.g., main object) present in the video. The object tag tells us whether an object is present in the video, but it does not provide any spatial or temporal information to localize the object within the video. Thus these videos are weakly labeled. In this paper, we tackle the problem of segmenting the object of interest in weakly labeled videos. This line of research will play an important role in many tasks related to video understanding. For example, it can enhance the browsing experience of users on video-sharing websites (e.g., YouTube). It can also improve video retrieval algorithms by removing the noisy videos or false positives from the search results.

Video segmentation is a fundamental problem in computer vision. Supervised learning of segmentation models requires all pixels in the training videos to be fully labeled. This is very time consuming and expensive. To address this drawback, weakly-supervised methods are proposed to alleviate the burden of labeling training video data. Weakly labeled videos have video-level labeling

instead of pixel-level labeling. For example, a video may have a video-level tag assigned to it, say "dog". From this tag we can interpret that an object "dog" may be present in the video. However, we do not have any spatial or temporal information of the object "dog" within the video.

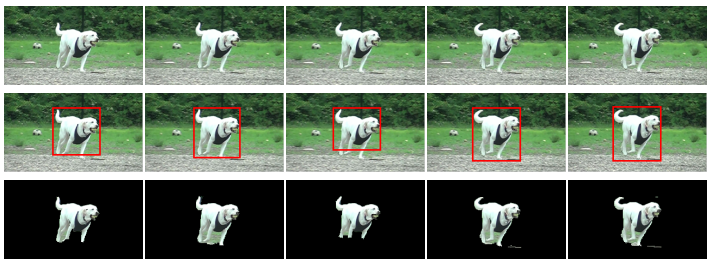

Fig. 1. A demonstration of our proposed approach. Given an input video with an object tag, e.g. "dog" (1st row), our proposed method can localize (2nd row) and segment (3rd row) the object in each frame.

Figure 1 demonstrates the pipeline of our proposed method. Given an input video with a tag, say "dog", we want to localize the dog in each frame, and segment the pixels corresponding to "dog" from the background. Our method is generic and can be applied to videos of any object category.

2 Related Work

Video segmentation is an active area of research in computer vision. Some of the proposed approaches are unsupervised, e.g. region tracking [1], hierarchical graph model [2, 3], multiple hypothesis tracking [4] and spatio-temporal based segmentation [5, 6]. Unsupervised methods can only perform low-level video segmentation and can not provide semantic labels for the segments.

Supervised methods have also been studied for semantic video segmentation, e.g. [7]. The major drawback of supervised video segmentation is that it requires lots of labeled video data for training. To address this issue, semi-supervised video segmentation [8, 9] methods are proposed. These methods address the limitation of supervised methods to some extent, but getting sufficient pixel-level labeled data is still nontrivial.

Weakly supervised video segmentation methods [10–12] are proposed to curtail the need of pixel-level labeled training video data. Our proposed method is inspired by this line of research. These methods use the semantic tags associated with the videos and do not require pixel-level labels. Since it is much easier to tag a video than labeling every pixels in the video, the labeling effort required is greatly reduced for weakly supervised methods. Rochan et al. [12] used video specific appearance model to localize object of interest in the video. Tang et al. [13] incorporated a temporal consistency model to their framework in order

to make their localization algorithm robust. These two recent works are the most relevant to our work.

3 Our Approach

In this paper, we tackle the problem of segmenting objects in weakly labeled videos. An input video is labeled with an object tag. Our goal is to segment the corresponding object in each frame of the video. Similar to [12], we make two assumptions about the input video. First, the object tag corresponds to the main object in the video. Second, there is a single instance of the tagged object within a video.

Our approach consists of four stages. 1) We generate a set of candidate object proposals for each frame within a video. Each object proposal is a bounding box that is likely to contain an object. 2) We build the appearance model of object of interest based on the object proposals. 3) We localize the object by selecting one bounding box for each frame. We model the object localization problem as performing the maximum a posterior (MAP) inference in an undirected chain graphical model. Each node in the graphical model corresponds to a frame and object proposals within a frame are the possible states of the node. An edge in the model enforces temporal consistency between two consecutive frames. The object in the video is localized by finding the optimal labeling of nodes in the graphical model. 4) After getting object localized in each frame, we segment the object from the background using the GrabCut [14] algorithm.

3.1 Generating Object Proposals and Building Appearance Model

For a given video, we generate a set of candidate object bounding boxes for every frame. Although we could use the state-of-the-art object detectors (e.g. [15]) to generate the object proposals, we will be limited to only a handful of object classes (e.g., 20 object classes in PASCAL datasets) for which reliable detectors are built. In this paper, we are interested in developing a method which can be applied to *any* object class, so we choose not to use object detectors in generating object proposals.

Instead, we use the Edge Boxes algorithm [16] to generate object bounding box proposals. This algorithm relies on one simple observation: the number of contours that are wholly enclosed by a bounding box is indicative of the presence of the object within the bounding box. The object proposals are detected using the edge maps. For a given bounding box, the algorithm also defines an objectness scoring function which measures the likelihood of this bounding box containing an object. Since this algorithm is not restricted to any particular object classes and can be potentially used for any object categories, we choose to use this algorithm to generate the object proposals.

Given an input video, we apply the Edge Boxes algorithm [16] to generate 10 object proposals (i.e. bounding boxes) for every frame within the video. This gives us a collection of candidate bounding boxes which are likely to contain

an object. Figure 2 shows an example of applying the Edge Boxes algorithm on frames in a video. The next step of our approach is to select a few bounding boxes which actually correspond to the object of interest in the video. We build our bounding box selection strategy based on two observations. First, it is observed that the Edge Boxes algorithm tends to give high objectness scores to the bounding boxes which enclose the object of interest within a video. Second, the appearance of object of interest remains consistent across all the frames within a video. In other words, if a "dog" is black in one frame, it will be black in all frames of the video. Using these two observations, we can build an appearance model of the object of interest for a specific video.

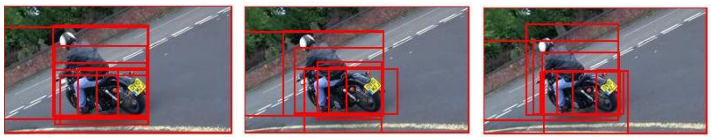

Fig. 2. Examples of applying the Edge Boxes algorithm on frames of a video.

We use an approach similar to [12] to build the object appearance model. We sort the candidate object proposals in a video according to their objectness scores returned by the Edge Boxes algorithm. Then we select T bounding boxes with the highest objectness scores. Following [12], we set T to be the number of frames within a video. We then build a color-based appearance model for the object of interest within the video. We compute the normalized color histogram of the selected T bounding boxes. The appearance model of the object of interest within a video is obtained by simply taking the mean of the color histograms of the selected bounding boxes[12].

3.2 Object Localization

We have a set of object proposals for every frame within a video. In this section, our goal is to localize the object in the video by selecting one bounding box for each frame. We model the localization problem using an undirected chain graph. Each node in the graph represents a frame within a video. The value assigned to a node indicates which object proposal is chosen for this frame. Since we have 10 object proposals for each frame, each node can take its value from $\{1, 2, ..., 10\}$. The nodes of two adjacent frames are connected by an edge indicating the temporal consistency constraint between these two frames. Let $X_1, X_2, ..., X_k$ be the frames in a video with k frames, and $P_1, P_2, ..., P_k$ be the corresponding object proposals selected for each frame. We use the following optimization problem to solve the object localization:

$$\max_{P_1,P_2,...,P_k} \sum_i \phi(P_i, X_i) + \sum_{i,i+1} \psi(P_i, P_{i+1}) \tag{1}$$

This optimization problem in Eq. 1 involves unary potential functions $\phi(\cdot)$ defined on nodes and pairwise potential functions $\psi(\cdot)$ defined on edges in the graph. In the following, we describe these potential functions in detail.

Unary Potentials: The unary potential $\phi(\cdot)$ measures the likelihood that an object proposal belongs to the object class, i.e. it captures the compatibility between an object proposal and the appearance model of the object. For each frame, we define the unary potential as follows:

$$\phi(P_i, X_i) = \exp\Big(-\|A - f_h(P_i, X_i)\|^2\Big) \tag{2}$$

where A is the appearance model of the object of interest within the video (see Sec. 3.1) and $f_h(P_i, X_i)$ is the normalized color histogram of the image patch corresponding to the bounding box P_i in the frame X_i. The unary potential will encourage each frame to choose a bounding box whose appearance (i.e. color histogram) is consistent with the appearance model of the object.

Pairwise Potentials: The pairwise potential is a term which encourages the consistency between the bounding boxes selected in two adjacent frames. It is very unlikely that objects will undergo drastic changes in their properties such as size, position and appearance between two consecutive frames of a video. Following [13], we define the pairwise potential as:

$$\psi_v(P_i, P_j) = \alpha \exp\left(-\|f_c(P_i) - f_c(P_j)\|^2 - \|f_a(P_i) - f_a(P_j)\|^2\right) \tag{3}$$

where $f_c(P_i)$ denotes the coordinates of the center of the bounding box P_i, and $f_a(P_i)$ denotes the area of this bounding box. We normalize $f_c(P_i)$ by the height and width of the object proposal, and $f_a(P_i)$ by the maximum area between the two object proposals. The parameter α control the relative influence of the pairwise potential in the model.

The pairwise potential is intuitive because if two object bounding boxes of adjacent frames contain the same object, they should not be far apart and their area should not vary much. It will help us in eliminating object proposals which are far apart and vary greatly in their area between two consecutive frames.

Decoding: Given the model defined above, the inference problem we need to solve is to jointly choose the values of $P_1, P_2, ..., P_k$ to maximize Eq. 1. Figure 3 illustrates this inference problem. Each column in Fig. 3 corresponds to a frame. In each column, the rows indicate the object proposals in that frame. The inference problem can be interpreted as finding the optimal path from the start to end in Fig. 3. It can be efficiently solved by dynamic programming.

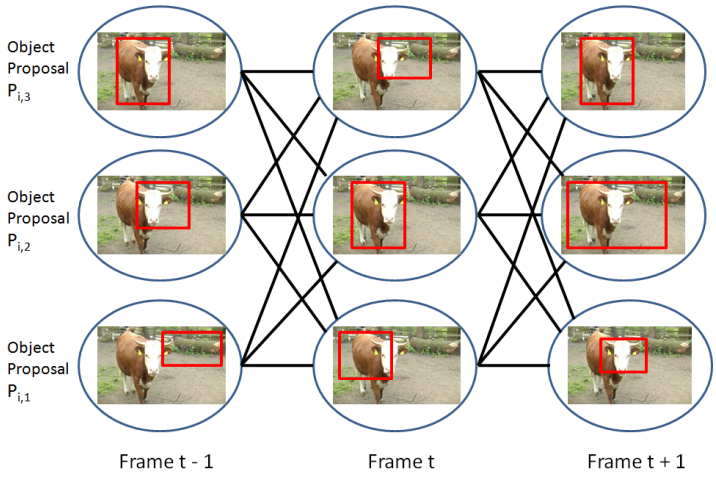

Fig. 3. For the given consecutive frames of a video, the inference problem for object localization can be represented as finding the optimal path in a graph. Each frame in the graph represents the node and their object proposals (blue circle) represent the possible state that node can take. The edges between the object proposals of two frames indicate the pairwise consistency constraint between the bounding boxes of two adjacent frames . Our goal is to find the best configuration of object bounding boxes among the frames of the video. This is equivalent to finding the optimal path in the graph.

3.3 Object Segmentation

After obtaining one bounding box in each frame, we apply GrabCut [14] to segment the object of interest from its background. GrabCut is an efficient segmentation algorithm, but it requires the user input in the form of a bounding box around the object to be segmented. Following [12], we eliminate the need for user inputs and make the GrabCut algorithm fully automatic. We simply use the bounding boxes returned from Sec. 3.2 as the input to the GraphCut algorithm.

4 Experiments

In this section, we describe the dataset and parameter settings used in our experiments. We then present the experimental results of our approach and perform the comparison with other state-of-the-art methods.

4.1 Dataset and Setup

We use the subset of the dataset described in Tang et al. [10]. The dataset is built using YouTube-Objects dataset [17] which consists of videos collected for

Table 1. Summary of the dataset used in our experiments.

	aeroplane	bird	boat	car	cat	cow	dog	horse	bike	train	total
# of shots	9	6	17	7	13	20	27	17	10	18	144
# of frames	1423	1206	2779	577	3870	2978	3803	3990	827	3270	24723

10 different object classes. We use this dataset because all the frames of a video have object of interest segmented [10]. Therefore, these videos can be used as ground-truth for evaluation. We use 144 video shots with a total of 24,723 frames in our experiments. Table 1 summarizes the number of shots and frames for each object classes in the dataset.

We randomly choose one video from every object class for setting the parameter α (see section 3.1). We empirically found $\alpha = 1.5$ to be a good choice and use this value throughout our experiments.

4.2 Results

Following [13, 12], we define our evaluation metric in terms of the percentage of frames for which we correctly localize the object of interest. We use the PASCAL-criterion [18] to evaluate the performance of our approach for every frame within a video shot. For a given frame, let P_b be the set of foreground pixels returned by the algorithm and P_{gt} be the set of ground-truth foreground pixels in this frame. We define a ratio r as $r = |P_b \cap P_{gt}| \,/\, |P_b \cup P_{gt}|$. We consider the object to be correctly localized in this frame if the ratio r is greater than 0.5.

Table 2. Comparison of our approach with previous work [12, 13]. For each object class, we show the percentage of the frames where the object is correctly localized.

method	aeroplane	bird	boat	car	cat	cow	dog	horse	bike	train	average
[12]	54.79	37.91	27.15	49.73	16.06	42.23	34.79	21.64	9.27	12.28	30.59
[13]	25.12	31.18	27.78	38.46	**41.18**	28.38	33.91	**35.62**	**23.08**	**25.00**	30.97
Ours	**57.53**	**39.8**	**29.4**	**52.04**	17.32	**45.19**	**38.36**	22.93	10.54	14.63	**32.77**

We compare the performance of our approach with previous work in [13, 12]. The comparisions are shown in Table 2. We can see that our proposed approach outperforms [12] in every object class. This demonstrates the benefit of incorporating pairwise consistency between adjacent frames. We also outperforms the state-of-the-art method in [13] in 6 out of 10 object classes. Sample results of our method on these 10 object categories are shown in Fig. 6.

Figure 4 shows two examples demonstrating the benefit of having the pairwise potential in the model. Without the pairwise potential (1st row in Fig. 4), the selected bounding boxes can vary dramatically in terms of size and position. The pairwise potential can alleviate this problem and enforce the selected bounding boxes to be consistent (2nd row in Fig. 4).

Fig. 4. Examples illustrating the benefit of enforcing consistency between adjacent frames. (Top row) Without the pairwise potential, the selected bounding boxes can be dramatically different. (Bottom row) With the pairwise potential, the bounding boxes are more consistent across all frames.

In Fig. 5, we show some representative failure cases of our approach. The failures are often caused by occlusion, multiple instances of the object of interest, object of interest being too small in the scene, etc.

(a) (b) (c)

Fig. 5. Some typical failure cases of our approach: (a) occlusion; (b) multiple instances of the object of interest; (c) object of interest is too small in the scene.

5 Conclusions

In this paper, we have proposed a novel approach to segment the object efficiently based on video-level tags. We have introduced a formulation based on chain structured graphical models. Using dynamic programming, we can efficiently localize the objects in all frames in a video. Our experimental evaluation demonstrates the effectiveness of our approach compared with other methods. In the future, we would extend our work to handle videos with multiple object tags.

Acknowledgement: This work was supported by NSERC and the University of Manitoba Research Grants Program (URGP).

References

1. Brendel, W., Todorovic, S.: Video object segmentation by tracking regions. In: IEEE International Conference on Computer Vision. (2009) 833–840
2. Grundmann, M., Kwatra, V., Han, M., Essa, I.: Efficient hierarchical graph-based video segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition. (2010) 2141–2148

3. Xu, C., Xiong, C., Corso, J.J.: Streaming hierarchical video segmentation. In: European Conference on Computer Vision. (2012) 626–639
4. Vazquez-Reina, A., Avidan, S., Pfister, H., Miller, E.: Multiple hypothesis video segmentation from superpixel flows. In: European Conference on Computer Vision. Springer (2010) 268–281
5. Lezama, J., Alahari, K., Sivic, J., Laptev, I.: Track to the future: Spatio-temporal video segmentation with long-range motion cues. In: IEEE Conference on Computer Vision and Pattern Recognition. (2011) 3369–3376
6. Brox, T., Malik, J.: Object segmentation by long term analysis of point trajectories. In: European Conference on Computer Vision. (2010) 282–295
7. Raza, S.H., Grundmann, M., Essa, I.: Geometric context from videos. In: IEEE Conference on Computer Vision and Pattern Recognition. (2013) 3081–3088
8. Badrinarayanan, V., Galasso, F., Cipolla, R.: Label propagation in video sequences. In: IEEE Conference on Computer Vision and Pattern Recognition. (2010) 3265–3272
9. Badrinarayanan, V., Budvytis, I., Cipolla, R.: Semi-supervised video segmentation using tree structured graphical models. IEEE Transactions on Pattern Analysis and Machine Intelligence **35** (2013) 2751–2764
10. Tang, K.D., Sukthankar, R., Yagnik, J., Li, F.F.: Discriminative segment annotation in weakly labeled video. In: IEEE Conference on Computer Vision and Pattern Recognition. (2013) 2483–2490
11. Hartmann, G., Grundmann, M., Hoffman, J., Tsai, D., Kwatra, V., Madani, O., Vijayanarasimhan, S., Essa, I., Rehg, J., Sukthankar, R.: Weakly supervised learning of object segmentations from web-scale video. In: ECCV Workshop on Web-scale Vision and Social Media. (2012) 198–208
12. Rochan, M., Rahman, S., Bruce, N.D., Wang, Y.: Segmenting objects in weakly labeled videos. In: Canadian Conference on Computer and Robot Vision, IEEE (2014) 119–126
13. Joulin, A., Tang, K., Fei-Fei, L.: Efficient image and video co-localization with frank-wolfe algorithm. In: European Conference on Computer Vision. (2014)
14. Rother, C., Kolmogorov, V., Blake, A.: Grabcut: Interactive foreground extraction using iterated graph cuts. In: ACM Transactions on Graphics (TOG). Volume 23., ACM (2004) 309–314
15. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part based models. IEEE Transactions on Pattern Analysis and Machine Intelligence **32** (2010) 1672–1645
16. Zitnick, C.L., Dollár, P.: Edge boxes: Locating object proposals from edges. In: European Conference on Computer Vision. (2014)
17. Prest, A., Leistner, C., Civera, J., Schmid, C., Ferrari, V.: Learning object class detectors from weakly annotated video. In: IEEE Conference on Computer Vision and Pattern Recognition, IEEE (2012) 3282–3289
18. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International Journal of Computer Vision **88** (2010) 303–338

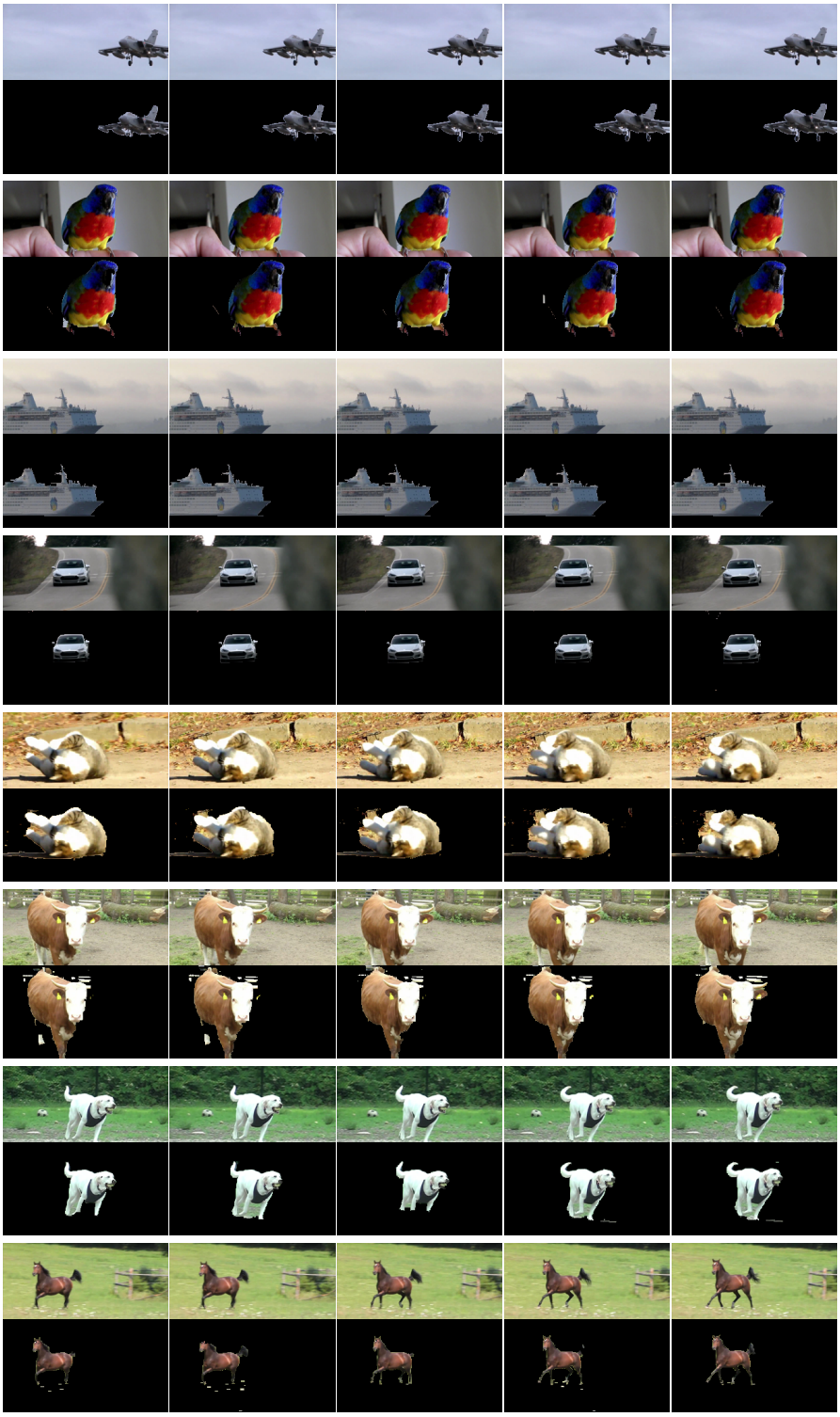

Fig. 6. For each video, we show the original input frames (1st row) and the segmented tagged object produced by our method (2nd row).