Latent SVM for Object Localization in Weakly Labeled Videos

Mrigank Rochan and Yang Wang
Department of Computer Science
University of Manitoba
Winnipeg, Canada
{*mrochan, ywang*}*@cs.umanitoba.ca*

***Abstract*—We consider the problem of object localization in Internet videos. An Internet video (e.g. YouTube videos) is often associated with a semantic label (also known as a tag) describing the main object present within it. However, the tag does not provide any spatial or temporal information about the main object in the video. Such videos are weakly labeled. Given weakly labeled video with video-level object class tags, our goal is to learn a model that can be used to localize the objects in other videos with such tags. We define a latent SVM based learning framework to tackle this problem. We demonstrate the effectiveness of our method on a dataset composed of videos collected from YouTube.**

***Keywords*-video understanding; weakly supervised; object localization**

I. Introduction

There is a massive amount of video content available on Internet through various video sharing websites (e.g. YouTube). These Internet videos are frequently associated with semantic tags describing the main object (or object of interest) present in them. However, these video-level tags do not provide any spatial or temporal information about the object of interest. For example, if a YouTube video is tagged with "cat", it tells us that the object "cat" is present somewhere in the video. However, the tag does not provide any information regarding the location of the object in each frame of the video. In computer vision, these videos are commonly referred to as weakly labeled videos. In this paper, we tackle the problem of localizing the objects in weakly labeled videos. We believe that this line of research can significantly improve the performance of existing video retrieval algorithms by reducing the number of false positives from search results. Moreover, it may also help in addressing several problems in the domain of video understanding.

Despite its significance, the problem is not well addressed in the computer vision literature. This is due to the fact that many standard techniques to tackle this problem are supervised (e.g. [1], [2]) and require access to large amount of labeled training data. As we know that collecting labeled training data is very expensive and time-consuming. To avoid the need of labeled training data, weakly supervised techniques are proposed. Our work is inspired by previous work on learning localized concepts [3]–[10] in videos. Given a video with video-level tag, say "cat", we try to localize and segment the region corresponding to object "cat" in each frame of the video.

In this paper, we use the latent SVM [1], [11] to learn the discriminative object models for object localization in videos. The main advantage of LSVM is that it does not require the exact object annotation in videos for learning. Given an input video with a video-level object class label, we would like to automatically determine where the object is in each frame of the video. We treat the spatial location of the object in each frame of the video as latent variables in our model.

The main technical contribution of this paper is the development of a latent SVM formalism to localize and segment objects in weakly labeled videos. This formalism treats the location of object as a latent variable and learn the object concepts using weakly labeled training data. Therefore, our method is scalable and can be used to exploit the huge amount of weakly labeled video data available on Internet to address various challenges in video understanding. In nutshell, our framework can be easily used to learn object concepts from Internet data.

II. Previous Work

In this paper we tackle the problem of localizing the object of interest (i.e. object corresponding to the video level tag) in weakly labeled videos. Our proposed method is inspired by the Multiple Instance Learning (MIL) framework. MIL is used to handle the problems with incomplete knowledge about labels of training data. In MIL, we are given a set of postively and negatively labeled bags of instances, where a positive bag contains at least one positive instance, and a negative bag contains no positive instances. Maron et al. [14] used MIL for scene classification. Galleguillos et. al [15] proposed MIL-based framework to recognize and localize objects in images. MIL has also been used in image annotation [10], object detection [1], etc.

Our work is also related to spatio-temporal segmentation in videos [9], [16]–[20] and some of the recent work that uses object annotation for various tasks in video understanding, including event detection [21], object segmentation [5], [22], and human activity recognition [23]. Our latent SVM based formulation is similar to that of Shapovalova et.

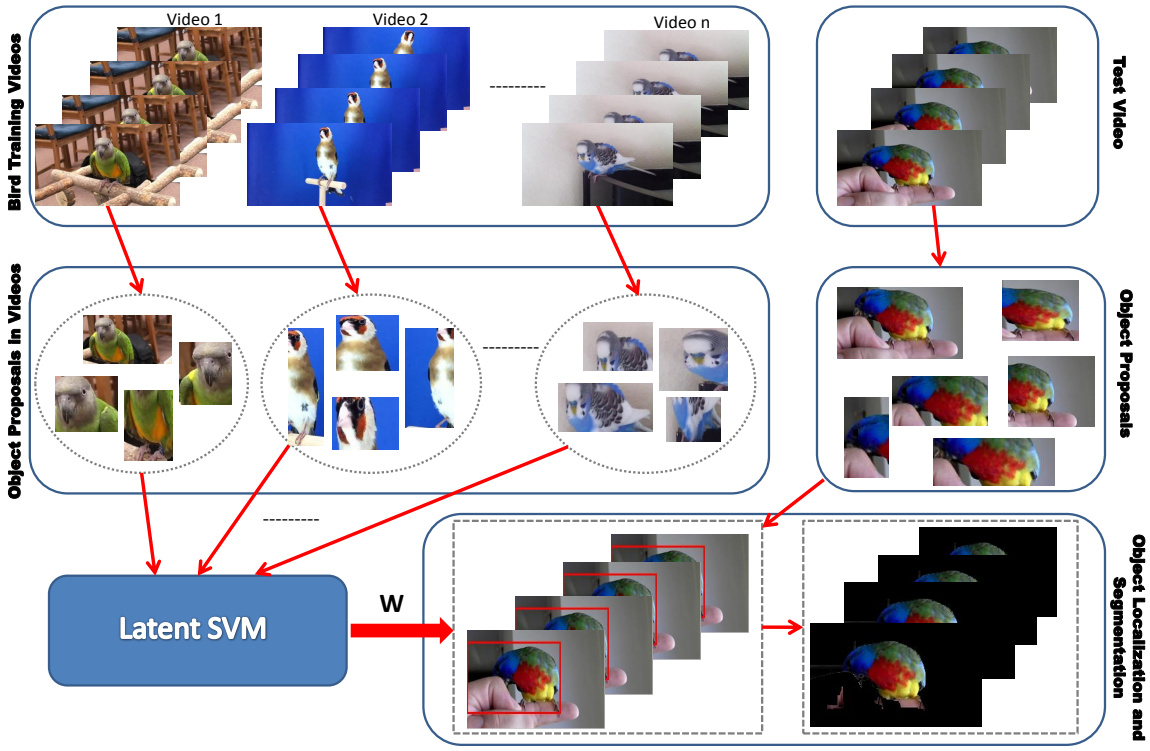

Figure 1. An overview of our approach. (1st row) We are given a collection of videos tagged with an object (say, "bird"). Our goal is to use them to learn a model to localize bird in a new testing video labeled as "bird". (2nd row) For each frame in a video, we extract object proposals by applying the Edge Boxes algorithm [12]. (3rd row) We use the latent SVM framework to learn object model from the training data. We then apply the learned model to re-score object proposals in each frame of the test video and select the top scored object proposal as the object location in each frame. Finally, we apply GrabCut [13] to segment the object in each frame.

al [24], where a latent SVM is used for weakly supervised action recognition and localization in videos.

Our work is closely related to a line of research on image and video understanding using weakly labeled data. Tang et. al [9] used video-level tags to annotate spatio-temporal segments in videos. Wang and Mori [10] proposed a latent model to capture the relationship between object tags and image regions. Rochan et al. [3] proposed a method for learning video specific appearance models to localize objects in weakly labeled videos.

III. Our Approach

The input to our method is a video with an object class label (e.g. bird). We follow the assumptions made in [3], [9] about the video. 1) There is only one instance of the object corresponding to the object class label of the video. 2) The label associated to the video represents the main object present within it and it appears in each frame of the video. For example, if a video is tagged with "bird", we assume that there is a bird in each frame of the video.

Our proposed approach (see Fig. 1) consists of following steps.

1) Generating object proposals: Given a video with an object class label, the first step of our approach is to generate a set of object proposals on each frame of the video. Each object proposal is a bounding box which is likely to contain any object. Note that we are interested in building an approach which can be applied to a video of any object category, so we use a generic algorithm that is not tuned to any specific object categories to generate object proposals.

2) Latent SVM learning: We use the latent SVM to learn a model for localizing an object in each frame of a video for a given object class. Our training data consist of frames extracted from videos with video-level object tags. We represent each frame in the form (x, h, y), where x is the frame itself, h is the latent variable that capture the unobserved information about the data, and y is the object class label. For example, suppose we want to learn a "bird" model from a set of videos labeled as "bird" (positive) or not "bird" (negative). We know that the object "bird" is present in each frame of a positive video but we do not know its exact location in the frame. In this scenario, h is used to represent the unobserved location of "bird" in each frame of the video. We train LSVM using an iterative algorithm that alternates between inferring variable h on frames of

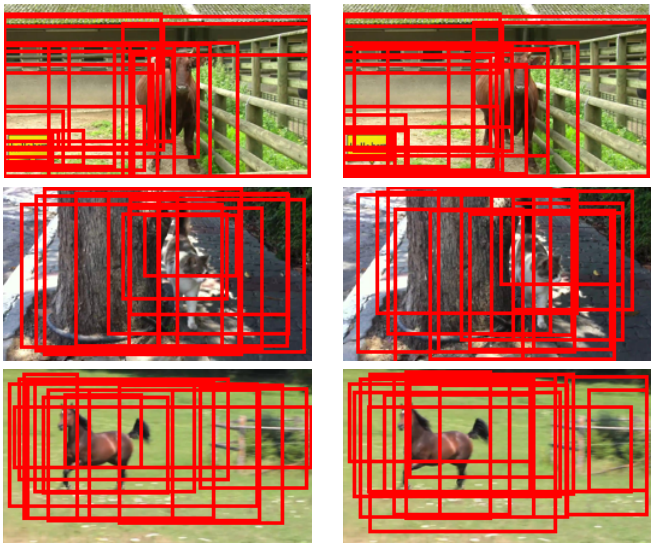

Figure 2. An example of generating object proposals. Given any frame of a video, we apply the Edge Boxes algorithm [12] on it. The algorithm returns a collection of object proposals that are likely to contain *any* object. The algorithm also assigns an objectness score to each object proposal indicating how likely it contains an object.

videos where object of interest (e.g. "bird") is present and optimizing the model parameters.

3) Applying the learned model on test videos: We now apply our learned models on test videos. When a test video with object class label (e.g. "bird") is given as an input, we take the learned "bird" model from the previous step and apply it to every frame of the video. Our learned model re-scores object proposals obtained from the first step. Note that in the first step, an object proposal will have a high score if it is likely to contain *any* object. However, after re-scoring, a high scored object proposal indicates the presence of object of interest (e.g. "bird") within it. We select the object proposal with the maximum score from each frame of the video as the final object location.

4) Segmenting the object of interest: In the last step, we consider the returned object proposals from the previous step as the location of the object of interest and perform segmentation. We employ the GrabCut [13] algorithm to segment the object from its background on each frame.

We describe the details of each step in the following subsections.

A. Generating Object Proposals

The first step in our approach is to generate a set of object proposals on each frame of a given video. We use the Edge Boxes algorithm [12] for this purpose. Note that we choose not to use existing object detectors (e.g. [1]) to generate the proposals. The reason is that we are interested in a framework to be used to localize objects of *any* object category. Current object detectors can only generate proposals for a limited number of object categories.

The Edge Boxes algorithm [12] relies on one simple observation: the number of edges that are totally enclosed within a bounding box is indicative of the presence/absence of an object in the box. The algorithm assigns an objectness score to a bounding box based on the edge strengths within it minus those that are part of a contour which straddles the boundary of that box. This algorithm is not restricted to any particular set of object categories and therefore we decide to use it for generating object proposals within each frame of a video.

Figure 2 shows examples of object proposals generated on several frames of videos.

B. Latent SVM

We now define the latent SVM formulation for object localization in weakly labeled videos. We assume that we are given videos with object class labels. For each frame in a video, we assume a latent variable indicating the location of the object in that frame. We aim to learn a model that can predict a latent region corresponding to object of interest in each frame of a test video.

1) Scoring Function: We consider all the frames from the training videos with the class label (e.g. "bird") as positive examples and all the other frames from the training videos as negative examples. We represent a frame as (x, h, y), where x is the frame itself and y is the object class label of this frame. Each frame is associated with a latent variable h indicating the location of the object in the frame. This is a latent variable because this information is unobserved on training data. We formulate a model for scoring a frame x with object class label y as follows:

$$F_w(x, y) = \max_h f_w(x, h, y) \tag{1}$$

$$f_w(x, h, y) = w^T \phi(x, h, y) \tag{2}$$

where $\phi(x, h, y)$ is a vector of image features extracted at the location h in the frame, and w is a vector of model parameters to be learned. In this paper, we use the Caffe-based CNN feature [25] which has been shown to be effective on a wide range of recognition problems in computer vision.

2) Learning Formulation: Give a set of N training examples $\{(x_i, y_i)\}_{i=1}^N$, we use the following latent SVM [1], [11] formulation to learn the model parameters w:

$$\min_{w,\xi>0} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N} \xi_i \tag{3}$$

$$\text{s.t.} \quad F_w(x_i, y_i) - F_w(x_i, y) \geq \Delta(y, y_i) - \xi_i \quad \forall i, \forall y$$

Eq. 3 is a constrained optimization equation. The constraints in Eq. 3 will make sure that the learned model w correctly classify the training examples. The slack variables ξ_i ensure the optimization is solvable. The parameter C is a hyperparameter that controls overfitting. $\Delta(y, y_i)$ is a function that

measures the loss of predicting y when the ground-truth label is y_i. We use the standard 0/1 loss:

$$\Delta(y, y_i) = \begin{cases} 1 & \text{if } y \neq y_i \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

3) Learning Procedure: We use a non-convex bundle method [26] to solve the optimization in Eq. 3. It is an extension of the cutting plane algorithm used in standard SVM solvers. This algorithm performs piecewise quadratic approximation of the objective function. In each iteration, it computes a linear approximation of the objective function using a subgradient and add it to the piecewise quadratic approximation.

First, we rewrite Eq. 3 as a equivalent non-constraint optimization as follows:

$$\min_w \frac{1}{2}\|w\|^2 + \sum_{i=1}^{N} R(w; x_i, y_i), \text{where} \quad (5)$$

$$R(w; x_i, y_i) = \max_y \Big[\Delta(y, y_i) + F_w(x_i, y)\Big] - F_w(x_i, y_i)$$

The NRBM [26] method requires the subgradient $\frac{\partial R(w;x_i,y_i)}{\partial w}$ at each iteration. It can be shown that the subgradient can be calculated as:

$$\frac{\partial R(w; x_i, y_i)}{\partial w} = \phi(x_i, h^*, y^*) - \phi(x_i, h, y_i) \quad (6)$$

where h^*,y^*, h' are defined as:

$$(h^*, y^*) = \arg\max_{y,h}(\Delta(y, y_i) + f_w(x_i, h, y))$$
$$h' = \arg\max_h \ f_w(x_i, h, y_i) \quad (7)$$

Given the subgradient, the NRBM method will find a local optimum of Eq. 5.

C. Applying the Learned Model on Test Videos

Once we have learned the model parameters w for different object classes, they can be used to perform inference on test videos. For a given test video v of the object class label y, our inference task is to select the top scored object proposal h^* (i.e., a latent region corresponding to the object of interest) in each frame of the video.

The latent region h^* for a frame x is computed as follows:

$$h^* = \arg\max_h \ f_w(x, h, y) \quad (8)$$

Figure 3 shows example of applying the learned "cow", "cat" and "horse" model on several frames of test videos.

D. Segmenting the Object of Interest

Now we have a bounding box in each frame indicating the location of the object in that frame. Our next step is to segment the object from its background. As in [3], we use the GrabCut [13] algorithm to segment out the object of interest in each frame of the video. The standard GrabCut algorithm requires the user input in the form of a bounding box around the object. In our case, we do not require any user input since we can use the selected object proposal from each frame of the video as the input to the GrabCut algorithm.

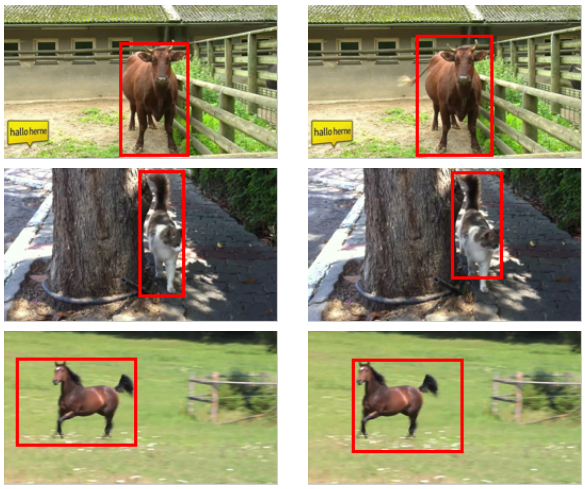

Figure 3. Visualization of applying a learned "cow", "cat", "horse" model on several frames of test videos labeled as "cow", "cat" and "horse" respectively. We apply the learned model on each frame of test video to re-score the object proposals. Finally, we select the top scored object proposal in each frame as the location of object of interest.

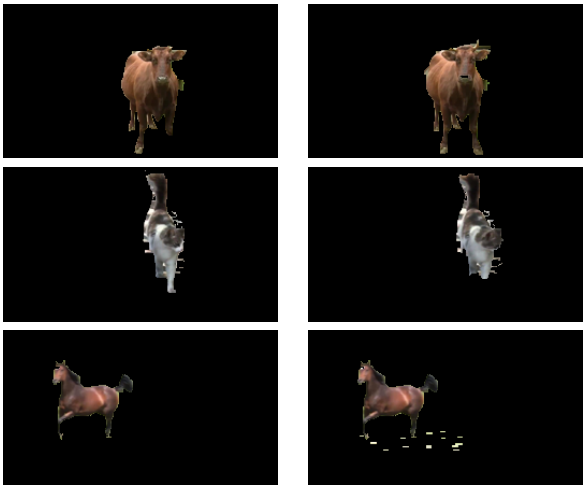

Figure 4. The GrabCut algorithm [13] is applied to segment out the object of interest from its background in each frame of a video. We use the selected object proposal in each frame (see Sec. III-C, also see Fig. 3) as the input to the GrabCut algorithm in order to make the entire method fully automatic.

Figure 4 shows example of applying the GrabCut algorithm to segment the object of interest within the frames of test videos.

IV. Experiments

We evaluate our proposed approach on a dataset consisting of videos collected from YouTube and compare with several

Table I
COMPARISON OF OUR APPROACH WITH PREVIOUS WORK [3]. FOR EACH OBJECT CLASS, WE SHOW THE PERCENTAGE OF THE FRAMES WHERE THE OBJECT OF INTEREST IS CORRECTLY LOCALIZED.

method	aeroplane	bird	car	cow	motorbike	boat	cat	dog	horse	train	average
[3]	22.61	26.04	37.66	17.81	25.11	6.62	**14.6**	29.09	19.29	8.86	20.77
[3] with CNN feature	**73.03**	**45.11**	18.18	25.11	1.75	23.47	0.5	33.31	30.74	16.44	26.76
Ours	58.12	19.07	**67.53**	**64.46**	**37.77**	**45.46**	9.7	**61.19**	**33.17**	**16.71**	**41.32**

baseline approaches.

A. Dataset and Setup

We use the dataset from Tang et. al [9] which contains 144 video shots from 10 different object classes, including aeroplane, bird, car, cow, etc. This dataset is originally built from YouTube-Objects dataset [7]. Every video shot in the dataset is annotated with the segmentation of main object (i.e. object of interest) in it. Table II summarizes the number of video shots and the total number of frames for each object class in the dataset.

We divide this dataset into training and testing sets. We randomly choose nearly 50 percent of video shots of each class for training and use the rest for testing. In the end, our test data contain 65 video shots with a total of 11169 frames from 10 different object classes.

Table II
SUMMARY OF THE DATASET USED IN THE EXPERIMENTS.

Class	Number of Shots	Number of Frames
Aeroplane	9	1423
Bird	6	1206
Car	7	577
Cow	20	2978
Motorbike	10	827
Boat	17	2779
Cat	13	3870
Dog	27	3803
Horse	17	3990
Train	18	3270
Total	144	24723

We define our evaluation metrics in terms of the percentage of frames in which the object of interest is correctly localized. We follow the PASCAL-criterion [27] to define whether a localization is correct. For a frame in a test video, let P_r be the set of foreground pixels returned by our algorithm and P_{gt} be the set of ground-truth foreground pixels provided by the annotation of the dataset. We define a ratio r as:

$$r = |P_r \cap P_{gt}| \,/\, |P_r \cup P_{gt}| \tag{9}$$

We consider the object of interest to be correctly localized in a test video frame if the ratio r is greater than 0.5. We evaluate the performance of our algorithm in terms of percentage of frames in which the object of interest is correctly localized.

B. Results

We compare the performance of our method with previous work in [3]. For a given video, [3] first generates a set of object proposals on each frame of the video. Then it builds a video specific appearance model of the object of interest by selecting top K (equal to number of frames in the video [3]) object proposals across all the frames of the video based on the objectness score. In the end, it applies this appearance model to select the best object proposal in each frame and performs segmentation.

The appearance model in [3] uses color histogram, which is not a very strong feature. As a second baseline, we replace the color histogram feature with the state-of-the-art CNN feature [25] to make their method more robust.

Table I shows the comparison of our method and the baseline methods ([3] and [3] with CNN feature) for 10 object classes. It is clear from the results that our method significantly outperforms the baseline methods on 7 out of 10 object classes. Moreover, we also achieve a significant improvement in overall average performance compared to the baseline methods. Figure 5 and Figure 6 show some sample results of our approach on these 10 object classes.

V. CONCLUSION

We have introduced a latent SVM framework for efficient object localization in weakly labeled videos (e.g. YouTube videos). The algorithm learns the object appearance models from training videos with only video-level tags. Then these appearance models can be applied on test videos to localize the object of interest within them. Experimental results show the effectiveness of our proposed approach.

There are many possible directions for future work. First, we would like to extend our work to handle multiple instance of object of interest in a video. Secondly, it would be interesting to incorporate a temporal consistency model to our framework as objects do not tend to move far apart between consecutive frames. Lastly, we would also like to consider the complex online videos where objects from different classes appear at the same time.

ACKNOWLEDGEMENT

This work was supported by NSERC and the University of Manitoba Research Grants Program (URGP). We gratefully acknowledge the support of NVIDIA Corporation with the GPU donation used in this research.

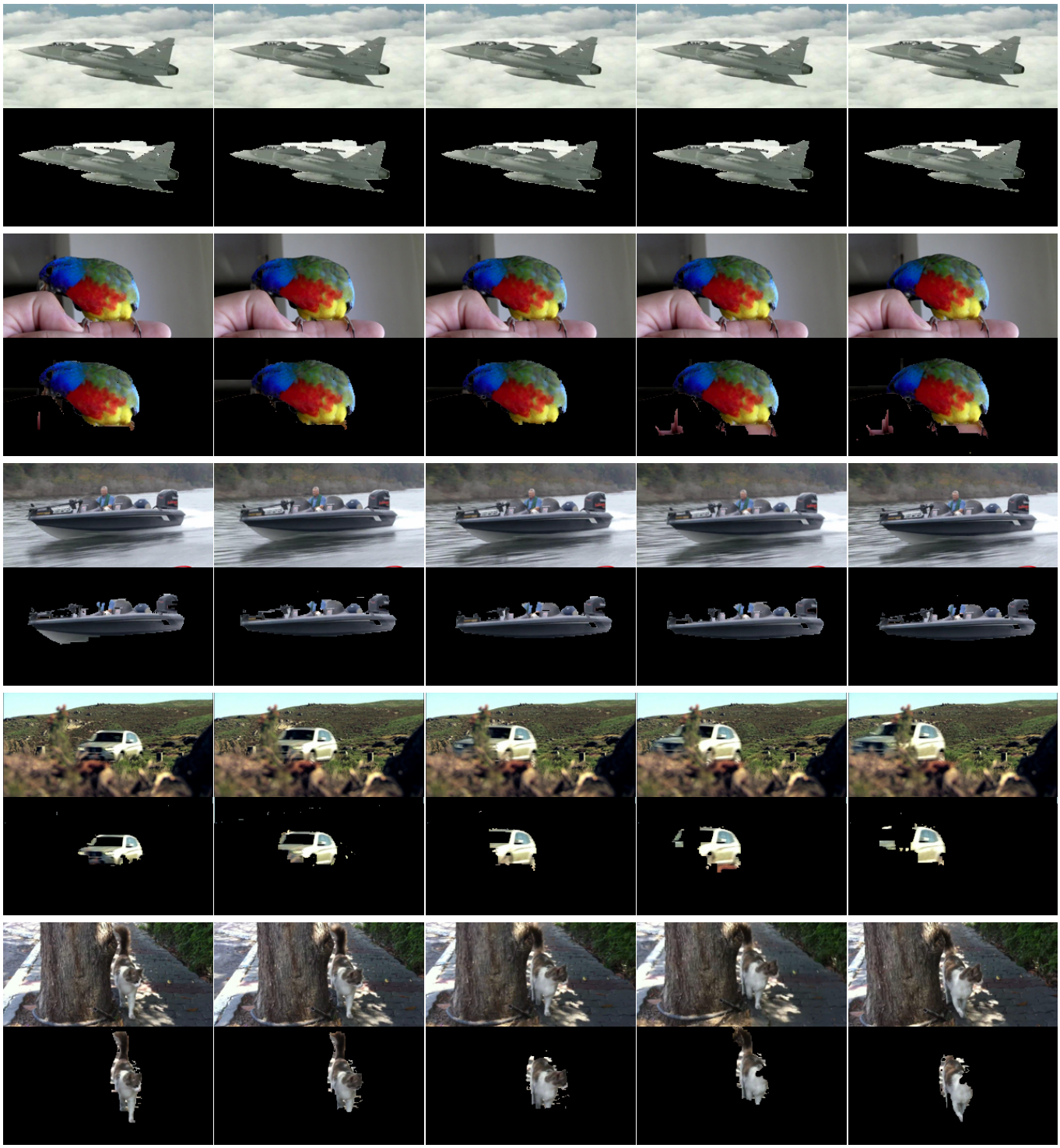

Figure 5. Visualization of results on videos tagged as (from top to bottom) “aeroplane”, “bird”, “boat”, “car”, and “cat”, respectively. For each video, we show the original frames (1st row) and the segmentation results (2nd row).

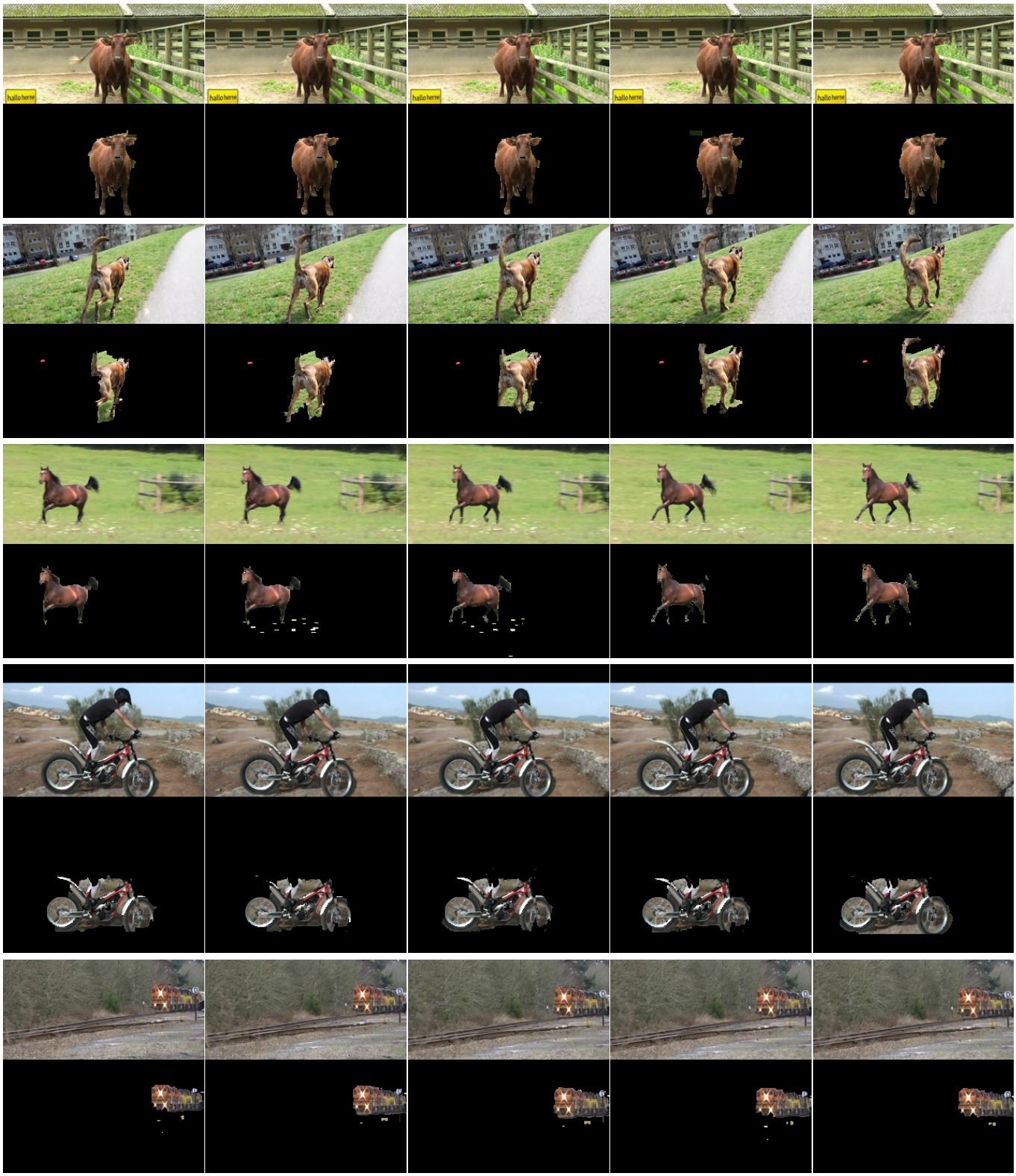

Figure 6. Visualization of results on videos tagged as (from top to bottom) “cow”, “dog”, “horse”, “motorbike”, and “train”, respectively. For each video, we show the original frames (1st row) and the segmentation results (2nd row).

REFERENCES

[1] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[2] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "TextonBoost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation," in *European Conference on Computer Vision*, 2006.

[3] M. Rochan, S. Rahman, N. D. Bruce, and Y. Wang, "Segmenting objects in weakly labeled videos," in *Canadian Conference on Computer and Robot Vision*. IEEE, 2014, pp. 119–126.

[4] M. Rochan and Y. Wang, "Efficient object localization and segmentation in weakly labeled videos," in *Advances in Visual Computing*. Springer, 2014, pp. 172–181.

[5] G. Hartmann, M. Grundmann, J. Hoffman, D. Tsai, V. Kwatra, O. Madani, S. Vijayanarasimhan, I. Essa, J. Rehg, and R. Sukthankar, "Weakly supervised learning of object segmentations from web-scale video," in *ECCV Workshop on Web-scale Vision and Social Media*, 2012, pp. 198–208.

[6] J. C. Niebles, C.-W. Chen, and L. Fei-Fei, "Modeling temporal structure of decomposable motion segments for activity classification," in *European Conference on Computer Vision*, 2010, pp. 392–405.

[7] A. Prest, C. Leistner, J. Civera, C. Schmid, and V. Ferrari, "Learning object class detectors from weakly annotated video," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3282–3289.

[8] K. Tang, L. Fei-Fei, and D. Koller, "Learning latent temporal structure for complex event detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1250–1257.

[9] K. D. Tang, R. Sukthankar, J. Yagnik, and F.-F. Li, "Discriminative segment annotation in weakly labeled video," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2483–2490.

[10] Y. Wang and G. Mori, "A discriminative latent model of image region and object tag correspondence," in *Advances in Neural Information Processing Systems*, 2010, pp. 2397–2405.

[11] C.-N. J. Yu and T. Joachims, "Learning structural svms with latent variables," in *International Conference on Machine Learning*. ACM, 2009, pp. 1169–1176.

[12] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *European Conference on Computer Vision*, 2014.

[13] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," in *ACM Transactions on Graphics*, vol. 23, no. 3. ACM, 2004, pp. 309–314.

[14] O. Maron and A. L. Ratan, "Multiple-instance learning for natural scene classification," in *International Conference on Machine Learning*, 1998.

[15] C. Galleguillos, B. Babenko, A. Rabinovich, and S. Belongie, "Weakly supervised object localization with stable segmentations," in *European Conference on Computer Vision*. Springer, 2008, pp. 193–207.

[16] T. Brox and J. Malik, "Object segmentation by long term analysis of point trajectories," in *European Conference on Computer Vision*, 2010, pp. 282–295.

[17] M. Grundmann, V. Kwatra, M. Han, and I. Essa, "Efficient hierarchical graph-based video segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2141–2148.

[18] J. Lezama, K. Alahari, J. Sivic, and I. Laptev, "Track to the future: Spatio-temporal video segmentation with long-range motion cues," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 3369–3376.

[19] J. Xiao and M. Shah, "Motion layer extraction in the presence of occlusion using graph cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1644–1659, 2005.

[20] C. Xu, C. Xiong, and J. J. Corso, "Streaming hierarchical video segmentation," in *European Conference on Computer Vision*, 2012, pp. 626–639.

[21] Y. Ke, R. Sukthankar, and M. Hebert, "Event detection in crowded videos," in *IEEE 11th International Conference on Computer Vision*, 2007, pp. 1–8.

[22] Y. J. Lee, J. Kim, and K. Grauman, "Key-segments for video object segmentation," in *IEEE International Conference on Computer Vision*, 2011, pp. 1995–2002.

[23] W. Brendel and S. Todorovic, "Learning spatiotemporal graphs of human activities," in *IEEE 11th International Conference on Computer Vision*, 2011, pp. 778–785.

[24] N. Shapovalova, A. Vahdat, K. Cannons, T. Lan, and G. Mori, "Similarity constrained latent support vector machine: An application to weakly supervised action classification," in *European Conference on Computer Vision*. Springer, 2012, pp. 55–68.

[25] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv:1408.5093*, 2014.

[26] T.-M.-T. Do and T. Artières, "Large margin training for hidden markov models with partially observed states," in *International Conference on Machine Learning*. ACM, 2009, pp. 265–272.

[27] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.