Object Localization in Weakly Labeled Images and Videos

by

Mrigank Rochan

A Thesis submitted to the Faculty of Graduate Studies of
The University of Manitoba
in partial fulfilment of the requirements of the degree of

MASTER OF SCIENCE

Department of Computer Science
University of Manitoba
Winnipeg

Thesis advisor

Yang Wang

Author

Mrigank Rochan

Object Localization in Weakly Labeled Images and Videos

Abstract

We consider the problem of localizing objects in weakly labeled images/videos. An image/video (e.g., Flickr image and YouTube video) is weakly labeled if it is associated with a tag describing the main object present in the image/video. It is weakly labeled because the tag only indicates the presence/absence of the object, but does not provide the detailed spatial location of the object. Given an image/video with an object tag, our goal is to localize the object in it. In this thesis, we propose two novel techniques to handle this challenging problem. First, we build a video-specific object appearance model and then incorporate temporal consistency information to localize the object. Second, we make use of existing detectors of some other object classes (which we call "familiar objects") to build the appearance model of the unseen object class (i.e., the object of interest). Experimental results show the effectiveness of the proposed methods.

Contents

List of Figures

List of Tables

Acknowledgments

First and foremost, I would like to begin by thanking my advisor Dr. Yang Wang for his continuous support and guidance. It has been a great inspiration to work under his mentorship. I am also honored to have Dr. Neil Bruce, Dr. Carson Leung and Dr. Wayne Xu on my thesis committee.

I wish to express my gratitude for the financial support provided by Dr. Wang and the Department of Computer Science, which was essential for the completion of this degree. I would also like to thank all the support staff in the department for their assistance.

I am extremely thankful to all my labmates for their help and useful suggestions on my research. Thank you for collaborating with me on various problems.

Last but not least, I am deeply thankful to my family for their encouragement and support, without which I wouldn't have reached this far.

This thesis is dedicated to my parents.

Publications

Some of the ideas, materials and figures in this thesis have appeared previously in the following publications:

1. **M. Rochan**, S. Rahman, N. Bruce, and Y. Wang. Segmenting Objects in Weakly Labeled Videos. *Conference on Computer and Robot Vision* (**CRV**), Montreal, Canada, May 2014.

2. **M. Rochan** and Y. Wang. Efficient Object Localization and Segmentation in Weakly Labeled Videos. *International Symposium on Visual Computing* (**ISVC**), Las Vegas, USA, December 2014.

3. **M. Rochan** and Y. Wang. Weakly Supervised Localization of Novel Objects using Appearance Transfer. *IEEE Conference on Computer Vision and Pattern Recognition* (**CVPR**), Boston, USA, June 2015.

4. **M. Rochan**, S. Rahman, N. Bruce, and Y. Wang. Weakly Supervised Object Localization and Segmentation in Videos. *Image and Vision Computing Journal* (**IVC**) (to appear).

Chapter 1

Introduction

Due to the popularity of online image and video sharing websites (e.g., Flickr and YouTube), an ever-increasing amount of image and video contents are becoming available nowadays. These online images/videos prove to be both a valuable resource and a grand challenge for computer vision. Internet images/videos are often weakly labeled. An image/video is weakly labeled if it is associated with a tag (e.g., YouTube videos with tags) describing the main object present within it. It is weakly labeled because the tag only indicates the presence/absence of the object, but does not give the detailed spatial location of the object in the image/video. For example, many Flickr images and YouTube videos have some tags associated with them. These tags are generated by users and provide some information about the contents (e.g., objects) within them. However, tags do not provide detailed spatial information about where the objects are. For instance, if a YouTube video is tagged with "dog", we know there is probably a dog somewhere in the video. But we cannot localize the dog in the video. In this thesis, our goal is to localize (i.e., output the bounding box) the

object in every frame of such weakly labeled videos. Similarly, given a collection of images labeled with an object category (e.g., "dog"), our aim is to localize this object in each image.

We propose simple and effective methods to localize object of interest in weakly labeled images/videos. Figure 1.1 illustrates the goal of our work. Given a collection of images or a video with a tag, say "car", we would like to localize the "car" in the images or the video. In other words, we try to answer the question "where is the object" in the weakly labeled data? A reliable solution to this problem will provide better image/video retrieval and browsing experience for users. It will also help us to solve a wide range of tasks related to to image/video understanding.

How would one detect an object class, say "car", in images? The de facto answer in computer vision is to collect a set of labeled training data (e.g., images with object bounding box annotations) for this object class and apply standard supervised machine learning to learn the appearance model for this object category. Then this appearance model can be used to detect cars in any image. The key of this standard pipeline is that we need to have access to a large amount of manually labeled training data. In the past few years, the availability of large-scale annotated datasets (e.g., PASCAL VOC [6] and ImageNet [30]) has been one of the driving forces of much progress in visual recognition. The PASCAL dataset [6] has focused only on 20 common objects. ImageNet [30] covers more object classes, but is still limited to the objects defined in the WordNet hierarchy, and most of the images in ImageNet are not annotated with object bounding boxes. Collecting labeled training data in the form of object bounding box annotation is an expensive task. Therefore, it is not

a collection of images labeled as "car"

a video labeled as "car"

Figure 1.1: Our goal is to localize objects in weakly labeled data. (Top) Given a collection of images labeled as "car", our algorithm will localize the car in each image of the collection. (Bottom) By applying our algorithm on a single video labeled as "car", we can localize the specific instance of "car" in this video. The red bounding boxes in this figure are outputs of our algorithm.

clear how this straightforward approach would scale up when we need to deal with a large number of concepts emerging over time, which is common for images/videos on the Internet.

We propose object localization techniques that do not require labeled (in the form of bounding boxes) training data. We use cheaply available labels (e.g., user-generated tags) associated with images/videos as cue to learn the appearance model of the object of interest. Our work is motivated by previous work on learning localized concepts [12; 20; 24; 35; 37; 39] in videos. Our work is also inspired by previous work on transfer learning for object detection [16]. The main advantage of our methods is

that they can be applied for *any* object category.

There are three major contributions in this thesis. First, we build a video-specific object appearance model to localize the object of interest in weakly labeled videos. Second, we introduce a temporal consistency constraint to our video-specific object appearance model to improve its efficiency. Third, we incorporate knowledge transfer into weakly supervised learning (WSL) of object classes.

The remainder of the thesis is organized as follows. In Chapter 2, we briefly discuss the related work. In Chapter 3, we describe a video-specific object appearance model to localize objects in weakly labeled videos. We also show that incorporating a temporal consistency constraint between consecutive frames is useful for localizing objects in videos. In Chapter 4, we introduce a transfer learning based framework to learn the appearance model of the object of interest. In Chapter 5, we conclude this thesis and discuss possible future directions.

Chapter 2

Related Work

In this chapter, we briefly discuss the works most related to our proposed methods.

2.1 Object Localization

We propose the following two ways to localize the objects in weakly labeled images/videos: i) building video-specific object appearance model and then incorporate temporal consistency, and ii) leveraging detectors of other related objects to learn the appearance of the object of interest. In the following, we review previous work related to each of these two strategies.

2.1.1 Object Localization with Temporal Consistency

For localizing object in a weakly labeled video, we build a video-specific appearance model of the object. This approach is inspired by some works in the domain of animal [26] and human tracking [25; 27]. The key idea used in these works is

that they learn the video-specific appearance models for the object. For example, Ramanan et al. [27] proposed a human kinematic tracking system which first detects stylized human poses in a given video. Then, they build an appearance model of human limbs specifically tuned for the person present in that particular video. It then applies this appearance model to every frame within the video to localize human. We use the similar logic to localize object in a video by constructing the video-specific appearance model of the object.

Ideally, in a video, objects in two consecutive frames does not undergo much change in their position, size and appearance. Tang et al. [15] proposed a temporal consistency model in their recent work on image and video co-localization. Co-localization is essentially an unsupervised problem in which common object in a set of images or videos are localized using bounding boxes [15]. This model incorporates temporal consistency metric to measure how well the bounding boxes between two consecutive frames agree in size and position. We use a similar way to incorporate temporal consistency to improve the performance of video-specific object appearance model. This consistency constraint ensures that the object localization (i.e., bounding box selected by our model in every frame of a video) is consistent.

2.1.2 Object Localization with Appearance Transfer

In computer vision, we currently have reasonable good detectors for a handful of object categories, e.g., the 20 object classes in the PASCAL challenge [8; 10]. We propose a transfer learning framework to leverage the object detectors from a small set of known object classes to build detectors for new object classes.

Lampert et al. [16] introduced an attribute-based classification technique which enabled information transfer between object classes. The transfer is done using an intermediate representation consisting of high-level attribute (e.g., shape, color, etc.) description provided by human. They learn these attributes with training examples. The learned model is then used to detect object classes for new test images based on their attribute description. These test images can even contain object classes for which there are no training example.

Our proposed method is related to a line of research on using linguistic knowledge to provide the link between the source object classes and target classes for knowledge transfer [28]. In natural language processing, people have developed techniques for learning the semantic relatedness of words from large collections of text documents [13]. For example, linguistic knowledge will tell us that "tiger" and "leopard" are two similar object classes. We propose to exploit this linguistic knowledge in our proposed approach.

2.2 Multiple Instance Learning

Our work is also related to a line of research on weakly-supervised learning (in particular, multiple-instance learning (MIL)) in computer vision. MIL is generally applied to tackle the problems where there is uncertainty about the labels of training data. In MIL framework, we have training data as pairs of bags labeled as positive or negative. A bag is labeled as positive if there is at least one instance of the label present within it, whereas a bag is labeled as negative if it contains no positive instance of the label. This setup is equivalent to our object localization problem setup where

we consider that the object of interest is present in a sub-region of positive examples (images or video frames), whereas the object of interest is not present in any sub-regions of negative examples.

MIL has been adopted in many computer vision applications, e.g., scene classification [18], object detection [8], object localization [9], image annotation [39], etc.

Maron et al. [18] applied multiple-instance learning for scene classification. Each image is treated as a bag where sub-regions of the image are the instances. They learn a classifier of the concept using a collection of positive and negative examples. This classifier is then used to select images from dataset that contains the concept.

Felzenszwalb et al. [8] proposed a object detection system that relies on partially labeled data. They reformulate MIL and SVM in terms of latent variables and called the formalism latent SVM. This method achieved state-of-the-art performance on detection problem.

MIL-based framework has been used to recognize and localize objects in images. For example, Galleguillos et al. [9] proposed a object recognition system that combines image descriptors and segmentations with a MIL algorithm to recognize and localize objects in images.

Wang and Mori [39] handle the problem of annotating image with unaligned textual object annotations. They develop a latent SVM framework [8] that captures the mapping between the annotations and image regions. This learned mapping relates the image sub-regions to their corresponding textual annotations.

Chapter 3

Object Localization with Temporal Consistency

In this chapter, we propose a video-specific appearance model for localizing objects in weakly labeled videos. We also introduce a temporal consistency constraint between consecutive frames to improve the performance of the model.

Video-specific appearance model construction has proven to be an efficient way for animal [26] and human tracking [25; 27]. The main idea used in these works is that they learn the video-specific object appearance models to localize the object. Ramanan et al. [27] proposed a human kinematic tracking system which first detects stylized human poses in a given video. Following this, they build an appearance model of human limbs specifically tuned for the person present in that video. It then uses this appearance model to localize human in each frame of the video. Motivated by this approach, we also build a video-specific appearance model to localize the object in a video tagged with an object name.

The major drawback of the above appearance-based method is that it ignores the temporal information between consecutive frames of a video. In general, an object appearance, size and position do not undergo drastic change between two consecutive frames. Based on this knowledge, we incorporate a temporal consistency constraint (similar to [15]) to our appearance-based framework to further improve its performance.

3.1 Our Approach

The type of input processed by our method is a video with an object tag, e.g., "cow". In our work, we focus on videos that are relatively simple. In particular, we make the following two assumptions about the videos: 1) the tag corresponds to the main object in the video; 2) there is only one instance of the tagged object in the video. More concretely, if a video is tagged with "cow", there should be a cow somewhere in the video. We assume the cow is the dominant object in the video, i.e., it is not too small. We also assume there is only one cow in the video. Previous work (e.g., [37]) in this area makes similar assumptions.

Based on these assumptions, our proposed approach involves four major steps:

1) Generating object proposals: Given a video with an object tag, the first step of our approach is to generate a collection of *object proposals* (also called *hypotheses*) on each frame in the video. Each object proposal is a bounding box that is likely to contain an object. The method we use for generating object proposals is generic and is not tuned for any specific object classes.

2) Building object appearance model: Many of the object proposals obtained

from the previous step might not correspond to the object of interest. In the second step, we use some simple heuristics to choose a few bounding boxes from the collection of all object proposals. The hope is that these selected bounding boxes are likely to correspond to the object of interest. We then build an appearance model for the object based on the selected bounding boxes. Note that the appearance model is built for a specific video. If the video contains a "black cow", our appearance model will try to detect this "black cow", instead of other generic cows.

3) Object localization: We localize the object by selecting one bounding box in each frame of a video. We could use the learned appearance model from the previous step to re-score the object proposals from the first step. After re-scoring, a bounding box will have a high score only if it is likely to contain an object instance specific to this video, e.g., a "black cow". However, this strategy alone may not be efficient enough to localize the object correctly. In this work, we assume that the object of interest in a video does not undergo drastic change in their properties such as size, position and appearance between two consecutive frames. Previous work (e.g., [41; 15]) has also made similar assumptions. Therefore, we enforce these constraints by incorporating a temporal consistency model between adjacent video frames. We model the object localization problem as performing the maximum a posteriori (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.

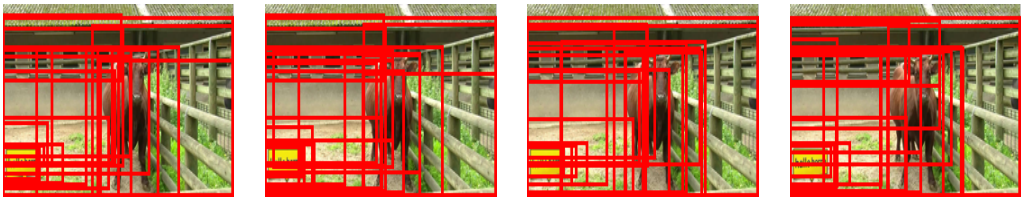


Figure 3.1: Examples of generating object proposals on frames within a video. Given a frame, the Edge Boxes algorithm [42] is applied. It returns a collection of bounding boxes in an image that are likely to be *any* object. For each bounding box, the algorithm also assigns a score indicating how likely it is to be an object.

4) Segmenting objects: After localizing an object in each frame, the GrabCut [29] algorithm is applied on the selected bounding box to segment the object from the background.

We describe the details of each step in the following.

3.1.1 Generating Object Proposals

Given an input video, the first step of our approach is to generate a set of candidate object bounding boxes on each frame. For certain object categories (e.g., people, car, etc.), one might be able to use state-of-the-art object detectors, e.g., [8]. But the limitation of this approach is that there are only a handful of object categories (e.g., 20 object categories in the PASCAL object detection challenge) for which we have reasonably reliable detectors. Since we are interested in localizing objects in a video regardless of the object class, we choose not to use object detectors.

Instead, we use the Edge Boxes algorithm [42] to generate object bounding box proposals. This algorithm works on one simple observation: the number of contours that are wholly enclosed by a bounding box indicates the presence of an 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 returns a collection of bounding boxes that are likely to contain any object, we choose to use this algorithm to generate the object proposals in our approach.

Given an input video, we apply the Edge Boxes algorithm [42] 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 3.1 shows some examples of applying the Edge Boxes algorithm on frames within a video.

3.1.2 Building Object Appearance Model

Given a video, the Edge Boxes algorithm approach (see Section 3.1.1) gives us a collection of bounding boxes. These bounding boxes correspond to image windows that are likely to contain *any* object. This algorithm is generic for any object class, i.e., the algorithm is not specifically tuned for any specific object categories. Figure 3.2 shows some examples of bounding boxes with high objectness scores, but that do not correspond to the object of interest (aeroplane) in the video. The next step of our approach is to select a few bounding boxes from all the generated object proposals. Ideally, the bounding boxes being selected will correspond to the object of interest in the video.

Our bounding box selection strategy is based on the following two observations. First, if a video is tagged with an object, say "cow", the image windows corresponding to the "cow" in the video tend to have high objectness scores. The reason is that

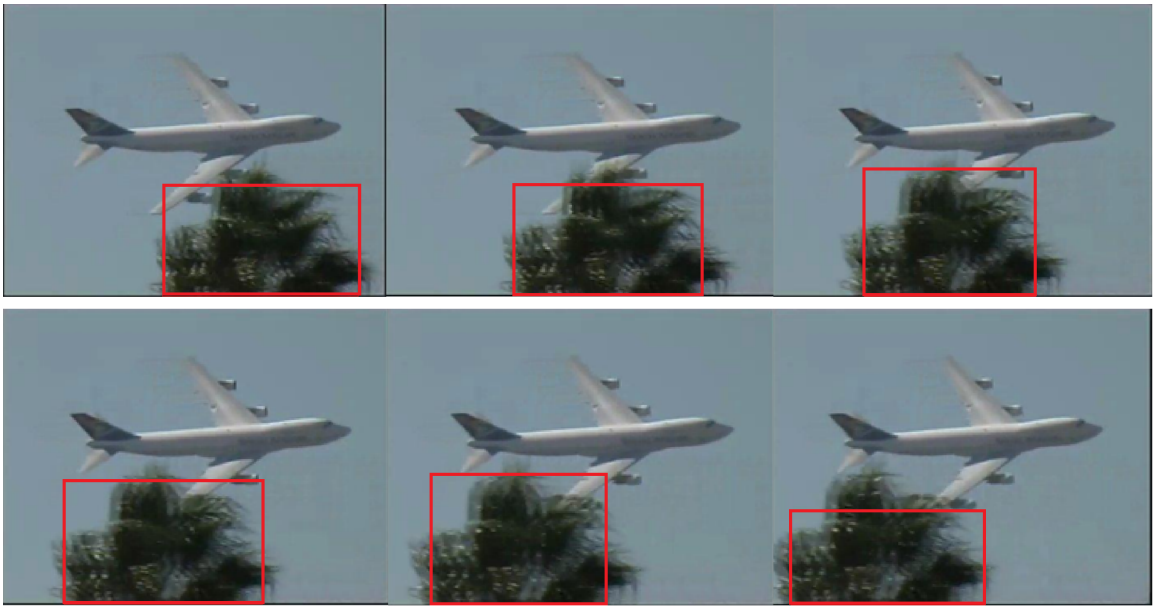


Figure 3.2: Example of high scoring bounding boxes on an image that do not correspond to the object of interest (aeroplane).

people are less likely to tag an object if it is not salient (e.g., too small) in the video. Second, we assume there is only one instance of the object of interest in the video. I.e., if a video is tagged as "cow", we only consider segmenting one "cow" in the video. In this case, the object of interest tends not to change appearance across different frames in the video. For example, if we know a "cow" is black in one frame, we know that it must be black in other frames as well. If we can somehow build an appearance model for this specific "black cow", we can use this appearance model to find "cow" bounding boxes in other frames.

Note that since our goal is to build an appearance model for the object of interest, our bounding box selection strategy does not necessarily have to retrieve all the true positive examples. As long as most of the bounding boxes being selected are positive examples of this object, we will be able to build a good appearance model for this

object. In other words, we would like our bounding box selection to have a precision, but can tolerate a low recall.

In our work, we use a simple yet effective strategy. We observe that if a video is tagged as "cow", most of the bounding boxes with the highest objectness scores tend to correspond to this object. This suggests that we can simply sort the bounding boxes in a video according to their objectness scores. Then we select K bounding boxes with the highest objectness scores. We empirically find the number of frames within a video to be a good choice for K and use this value in all of our experiments.

Based on the selected K bounding boxes, we build a video-specific appearance model for the object. We first extract the visual feature from each bounding box. In our experiments, we have used both the normalized color histogram and the CNN-based features implemented in Caffe [14]. We define two methods for building the appearance model. (1) *Averaging*: in this method, we simply take the average of the feature vectors extracted from all selected bounding boxes. Let A be the appearance model obtained by this method and $\mathbf{x}$ be the feature vector of an object proposal. We can use $(-||A-\mathbf{x}||_2)$ as a measure of how likely it is that $\mathbf{x}$ is the object in this video. (2) *SVM-based*: in the second method, we learn a model of the object of interest from the object proposals extracted from the video frames. We consider the selected K bounding boxes as positive examples of the object present within the video. We then choose a set of negative examples by randomly selecting object proposals from videos that do not correspond to the object of interest. Given this set of positive and negative examples, we train a linear SVM (with either color histogram or CNN features) to learn the video-specific object appearance model. Let $\mathbf{x}$ be the feature

vector (normalized color histogram or CNN features) of an object proposal in a video $\mathbf{v}$, the video-specific object appearance model is represented by parameter vector $\mathbf{w}_v$. The dot product $\mathbf{w}_v^\top \mathbf{x}$ indicates the likelihood of $\mathbf{x}$ being the specific object in the video $\mathbf{v}$.

3.1.3 Object Localization

We have a set of bounding boxes for every frame in a video. In this section, our goal is to localize the object in the video by selecting one bounding box for each frame. We could use the learned appearance model from Section 3.1.2 to localize the object of interest within a given video. I.e., we can use the learned appearance model to re-score the bounding boxes with the frames of that video. A bounding box will have a high score only if it is likely to contain an object instance specific to this video, e.g., a "black cow". However, this strategy alone may not be sufficient to localize the object correctly. We know that within a video it is very unlikely that objects will undergo drastic change in their properties such as size, position and appearance between two consecutive frames of a video. This prior is often used in tracking [15; 40; 36; 23; 21; 11; 1; 2] objects in videos. Therefore, we enforce a temporal consistency model between consecutive video frames.

We model the object localization problem within a video 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:

$$\max_{P_1,P_2,...,P_k} \sum_i \phi(P_i, X_i) + \sum_{i,i+1} \psi(P_i, P_{i+1}) \tag{3.1}$$

This optimization problem in Eq. 3.1 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. We use two different ways to define the unary potential. Firstly, we define the unary potential for each frame as follows:

$$\phi(P_i, X_i) = \exp\Big(-\|A - f_h(P_i, X_i)\|_2\Big) \tag{3.2}$$

where A is the appearance model (obtained by averaging) of the object of interest within the video (see Sec. 3.1.2) and $f_h(P_i, X_i)$ is the feature vector (color histogram or CNN feature) of the image patch corresponding to the bounding box P_i in the frame X_i.

Secondly, we also use the video-specific object appearance model learned using SVM to define the unary potential for each frame. In this case, the unary potential

is computed as follows:

$$\phi(P_i, X_i) = \Big(\mathbf{w}_v^\top \cdot f_h(P_i, X_i)\Big) \tag{3.3}$$

where $\mathbf{w}_v$ is the learned video-specific object appearance model and $f_h(P_i, X_i)$ is the feature vector from the image patch corresponding to the bounding box P_i in the frame X_i.

The unary potential in Eq. 3.2 and Eq. 3.3 will encourage each frame to choose a bounding box whose appearance (i.e., color histogram or CNN feature vector) is consistent with the video-specific appearance model of the object. We conduct experiments with both the definitions of unary potential with both color histograms and CNN features.

Pairwise Potentials

The pairwise potential is a term which encourages the temporal consistency between the bounding boxes selected in two adjacent frames. It ensures that the bounding boxes selected between adjacent frames do not undergo drastic changes in their properties such as size and position.

Following [15], we define the temporal consistency $C_{temporal}(P_i, P_j)$ between two bounding boxes P_i and P_j) of adjacent video frames as follows:

$$C_{temporal}(P_i, P_j) = \alpha\Big(\|f_c(P_i) - f_c(P_j)\|_2^2 + \|f_a(P_i) - f_a(P_j)\|_2^2\Big) \tag{3.4}$$

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 frame, and $f_a(P_i)$ by the maximum area between the two object proposals.

Using the above temporal consistency definition, we compute the pairwise potential between two bounding boxes of adjacent video frames as follows:

$$\psi_v(P_i, P_j) = \exp\left(-\left(C_{temporal}(P_i, P_j)\right)^2\right) \tag{3.5}$$

The parameter α in Eq. 3.4 control the relative influence of the pairwise potential in the model.

The pairwise potential is very intuitive because if two object bounding boxes of adjacent frames contain the same object then they should not be far apart and their area should not vary either. In summary, the pairwise potential encourages the algorithm to select bounding boxes that are consistent in terms of positions and sizes between adjacent video 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. 3.1. Figure 3.3 illustrates this inference problem. Each column in Fig. 3.3 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. 3.3. It can be efficiently solved by dynamic programming.

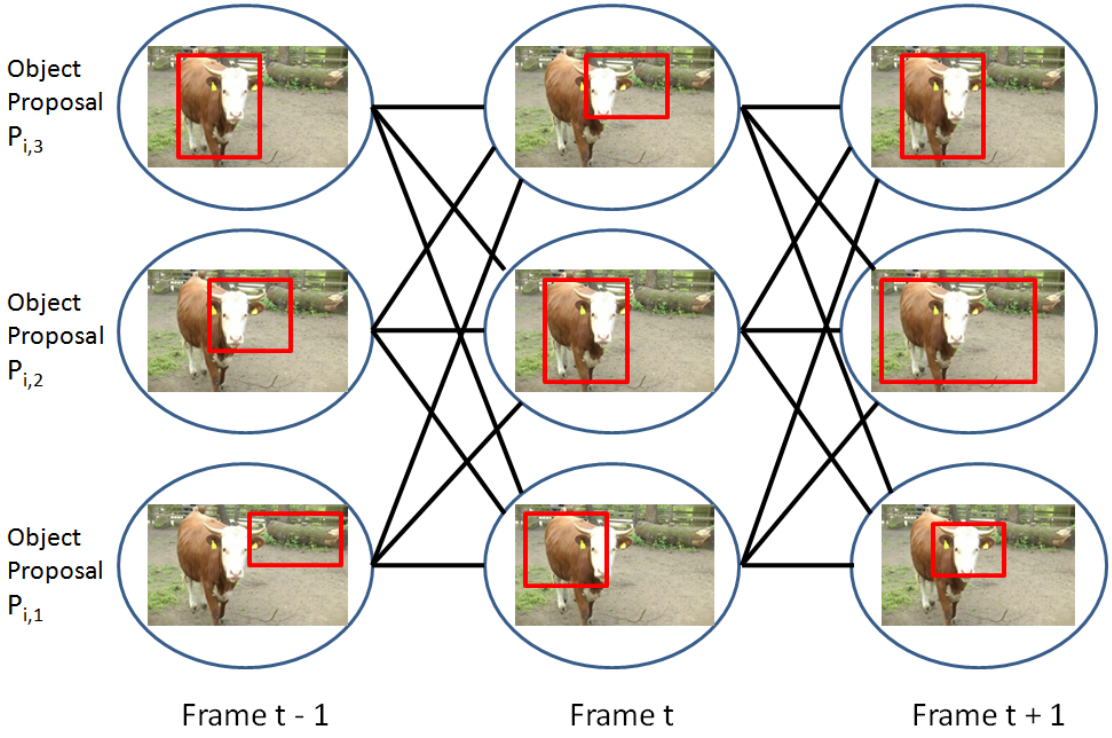
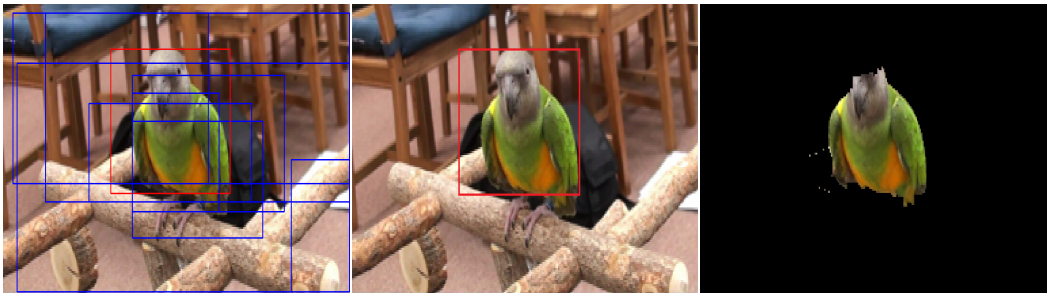


Figure 3.3: For the given consecutive frames of a video, the inference problem for object localization can be represented as finding the optimal path in a graph. Each frame in the graph represents the node and their object proposals (blue circle) represent the possible state that node can take. The edges 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.

3.1.4 Segmenting Object of Interest

Finally, we apply GrabCut [29] to segment out the object in each frame. GrabCut is an efficient algorithm for foreground segmentation in images. The standard GrabCut is not fully automatic. It requires the user input in the form of marking a rectangle around the foreground object. In contrast, our approach does not require user interaction. We simply consider the one bounding box selected by our localiza-



(a) (b) (c)

Figure 3.4: An illustration of our approach. (a) A frame in the video with selected bounding boxes (see Sec. 3.1.2). An appearance model is built based on the selected bounding boxes from all frames of this video. (b) After applying the appearance model on this frame, we obtain a single bounding box that is most likely to contain the object of interest (bird) in this frame. (c) The GrabCut algorithm is applied to segment the object in this frame. The standard GrabCut algorithm requires users to draw a rectangle around the foreground object as the part of the input. In our case, we use the bounding box obtained from (b) as the user input. So our method is fully automatic and does not require any user interactions.

tion algorithm within each frame as the user input. Figure 3.4 illustrates the pipeline of our approach.

3.2 Experiments

In this section, we first describe the dataset and evaluation metrics (Sec. 3.2.1). We then present our experimental results in Sec. 3.2.2.

3.2.1 Dataset and Setup

We evaluate our proposed approach using a subset of the dataset in Tang et al. [37]. This dataset consists of video shots collected for 10 different object classes, including aeroplane, bird, boat, car, cat, cow, dog, horse, motorbike, and train. Each frame

of the video shot is annotated with the segmentation of the object of interest in the video. Table 3.1 shows the summary of this dataset. We use 144 video shots with a total of 24,723 frames in our experiments.

Class	Number of Shots	Number of Frames
Aeroplane	9	1423
Bird	6	1206
Boat	17	2779
Car	8	601
Cat	13	3870
Cow	20	2978
Dog	27	3803
Horse	17	3990
Motorbike	10	827
Train	18	3270
Total	144	24723

Table 3.1: Summary of the dataset used in the experiments.

We define a quantitative measurement in order to evaluate our approach. Our quantitative measurement is inspired by the measurement used in the PASCAL challenge [6]. Given a video frame, let P_b be the foreground pixels returned by our method and P_{gt} be the ground-truth foreground pixels provided by the annotation in the dataset. We measure the quality of P_b by the ratio of $|P_b \cap P_{gt}|$ and $|P_b \cup P_{gt}|$:

$$r = |P_b \cap P_{gt}| \,/\, |P_b \cup P_{gt}| \tag{3.6}$$

If this ratio r is greater than 50%, we consider the segmentation on this frame to be correct. We evaluate the performance of our algorithm by computing the percentage of frames that are correctly segmented.

We extract 10 object proposals (or bounding boxes) from each frame of a video

shot. We use normalized color-histograms and state-of-the-art 4096 dimensional fine-tuned CNN features [14] as our feature representations for an object proposal. We randomly choose one video shot from every object class for setting the free parameter α (see Section 3.1.3) in our experiments.

3.2.2 Results

method	aeroplane	bird	car	cow	mbike	
top proposal only	52.54	**46.27**	42.48	33.25	4.95	
appearance only	54.6	37.81	49.73	42.3	9.27	
our approach	**57.53**	38.11	**50.27**	**44.35**	**10.54**	
method	boat	cat	dog	horse	train	average
top proposal only	24.75	**17.38**	34.12	21.04	10.85	28.76
appearance only	25.52	16.06	34.81	21.62	12.28	30.4
our approach	**27.08**	17.35	**36.04**	**22.88**	**12.38**	**31.65**

Table 3.2: Quantitative results using the averaging-based appearance model on color histogram features. For each object class, we compare segmentation accuracy across the sequence of video frames. A frame is considered to be correctly segmented if the ratio of intersection over union defined in Eq. 3.6 is greater than 50%. We compare four different methods: (1st row) bounding box with highest objectness score selected on each frame; (2nd row) video-specific appearance model generated 3.1.2 using normalized color-histogram feature from top-scored bounding boxes 3.1.2; (3rd row) incorporating temporal consistency between two consecutive frames with the color histogram based video-specific object appearance model.

In order to measure the performance of our proposed approach, we perform several experiments.

We first consider using the averaging-based appearance model based on color histogram (see Sec. 3.1.2). We compare our method with several baseline approaches. The first baseline simply chooses the bounding box with the highest objectness score (from Edge Boxes algorithm [42]) for each frame within a video. We call this baseline

method	aeroplane	bird	car	cow	mbike	
top proposal only	52.54	**46.27**	42.48	33.25	4.95	
appearance only	58.12	41.59	42.65	34.87	7.51	
our approach	**58.9**	42.39	**46.73**	**37.09**	**7.67**	
method	boat	cat	dog	horse	train	average
top proposal only	24.75	**17.38**	34.12	21.04	10.85	28.76
appearance only	26.53	11.7	34.23	22.85	**11.53**	29.16
our approach	**27.08**	12.51	**35.9**	**22.96**	11.27	**30.25**

Table 3.3: Quantitative results using the averaging-based appearance model on CNN features.

"top proposal only". The second baseline applies the video-specific object appearance model (averaging based on color histogram) to re-score the object proposals on each frame, then selects the proposal with the highest score. Note that this baseline does not consider the temporal consistency information between the object proposals selected from adjacent frames of a video. We call this baseline "appearance only". Table 3.2 shows the performance of three methods: 1) using first baseline method, i.e., "top proposal only"; 2) using second baseline method, i.e., "appearance only"; 3) using our method that combines video-specific object appearance with temporal consistency. Our approach achieves the best performance on most of the object classes.

Table 3.3 shows the performance of different methods using the averaging-based appearance model based on CNN features. Similar to Table 3.2, we compare the performance of three methods: 1) using "top proposal only"; 2) using the averaging-based appearance model based on CNN feature, i.e., "appearance only"; 3) using our approach that combines video-specific object appearance model (CNN feature based) with temporal consistency information. Our final method again outperforms

the other baseline methods.

method	aeroplane	bird	car	cow	mbike	
appearance only	52.94	36.52	41.59	29.98	**10.06**	
our approach	**60.27**	**38.61**	**56.28**	**36.1**	9.9	
method	boat	cat	dog	horse	train	average
appearance only	15.17	10.23	26.32	21.48	9.84	25.41
our approach	**16.29**	**13.98**	**30.34**	**25.05**	**11.27**	**29.81**

Table 3.4: Quantitative results using the SVM-based appearance model based on color histogram. We learn a video-specific appearance model using a linear SVM without the bias term. We select the object proposal with highest objectness score on each frame of a given video as positive example and select a set of negative examples by randomly choosing object proposals from videos of different object class. We compare performance of two methods: (1st row) using only the learned video-specific appearance model; (2nd row) incorporating temporal consistency between two consecutive frames with the video-specific appearance model.

method	aeroplane	bird	car	cow	mbike	
appearance only	**60.76**	53.63	56.28	41.07	**11.82**	
our approach	**60.76**	**54.63**	**57.35**	**42.13**	11.66	
method	boat	cat	dog	horse	train	average
appearance only	34.21	**19.49**	34.73	30.15	**11.73**	35.39
our approach	**34.72**	19.23	**35.84**	**30.4**	11.37	**35.81**

Table 3.5: Quantitative results on using SVM-based appearance model based on CNN features. Similar to Table 3.4, we compare the performance of two methods: (1st row) using appearance model only; (2nd row) incorporating temporal consistency to the framework.

We further investigate and evaluate the performance of video-specific object appearance model learned using SVM 3.1.2. Table 3.4 shows the performance of two methods: 1) using video-specific object appearance model learned with normalized color histogram feature from object proposals , i.e., "appearance only"; 2) using temporal information with the SVM-based appearance model. Similar to Table 3.4, we also compare the two methods when CNN feature is used to learn the video-specific

object appearance model (see Table 3.5). In both the cases, our final approach outperform the other baseline method. Note that, in contrast to Table 3.2 and Table 3.3, we obtain better performance with CNN feature rather than color histogram for SVM-based methods. The main reason is that SVM learns a better appearance model using the high-dimensional discriminative CNN feature representation than the low dimensional color histogram feature. These results are also in agreement with many CNN feature representation based visual recognition algorithms where the representation has proved to be one of the state-of-the-art.

Tables 3.2–3.5 show that our final approach (video-specific object appearance model with temporal consistency) outperforms the baseline methods on most of the object categories. Firstly, from various results, we observe that building a video-specific object appearance model (averaging-based or SVM-based) is an effective strategy to tackle the localization problem in weakly labeled video. Secondly, we show that incorporating temporal consistency information to the framework further improves the performance. Qualitative results of our approach are shown in Fig. 3.7.

Figure 3.5 shows two examples demonstrating the benefit of having the pairwise potential in the model. Without the pairwise potential (1st row and 3rd in Fig. 3.5), the selected bounding boxes between adjacent frames of a video can vary dramatically in terms of size and position. The pairwise potential alleviates this problem and enforces the consistency across the selected bounding boxes between consecutive frames of a video (2nd row and 4th row in Fig. 3.5).

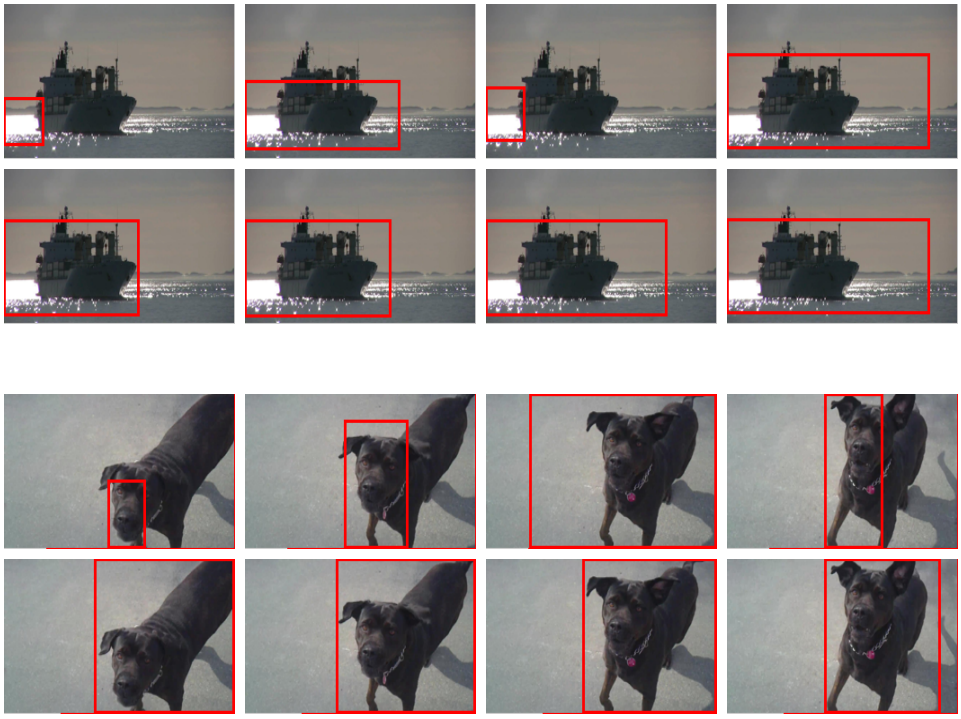


Figure 3.5: Examples illustrating the benefit of enforcing consistency between adjacent frames of videos. (1st and 3rd row) Without the pairwise potential, the selected bounding boxes can be dramatically different. (2nd and 4th row) With the pairwise potential, the bounding boxes are more consistent across all frames.

3.2.3 Failure Cases

In Fig. 3.6, we show some representative failure cases of our approach. The failures are often caused by occlusion, multiple instances of the object of interest and the object of interest being too small in the scene.

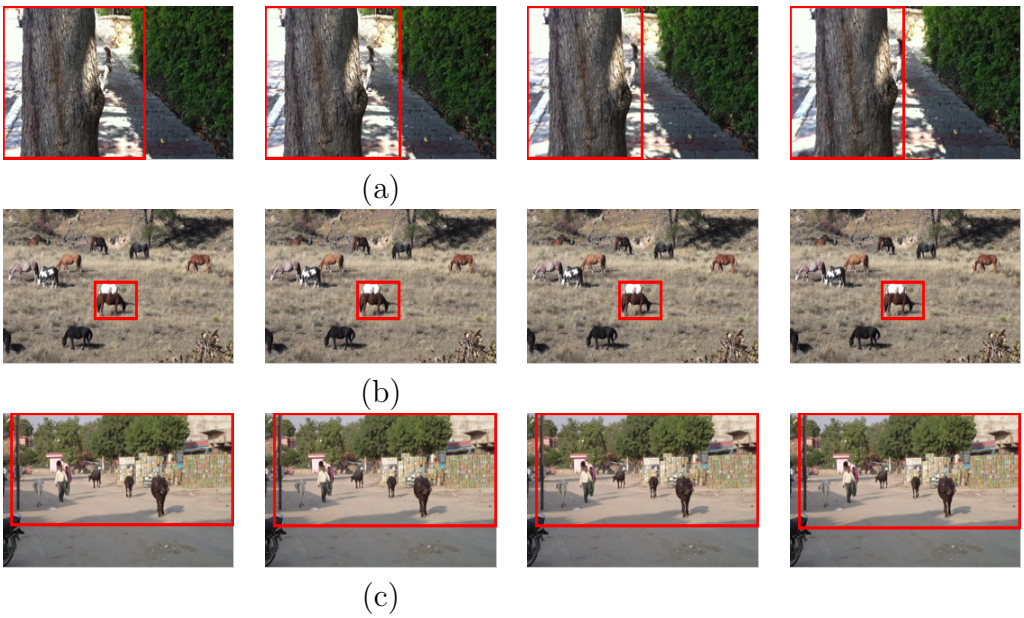


Figure 3.6: Some typical failure cases of our approach: (a) occlusion; (b) multiple instances of the object of interest; (c) object of interest is too small in the scene.

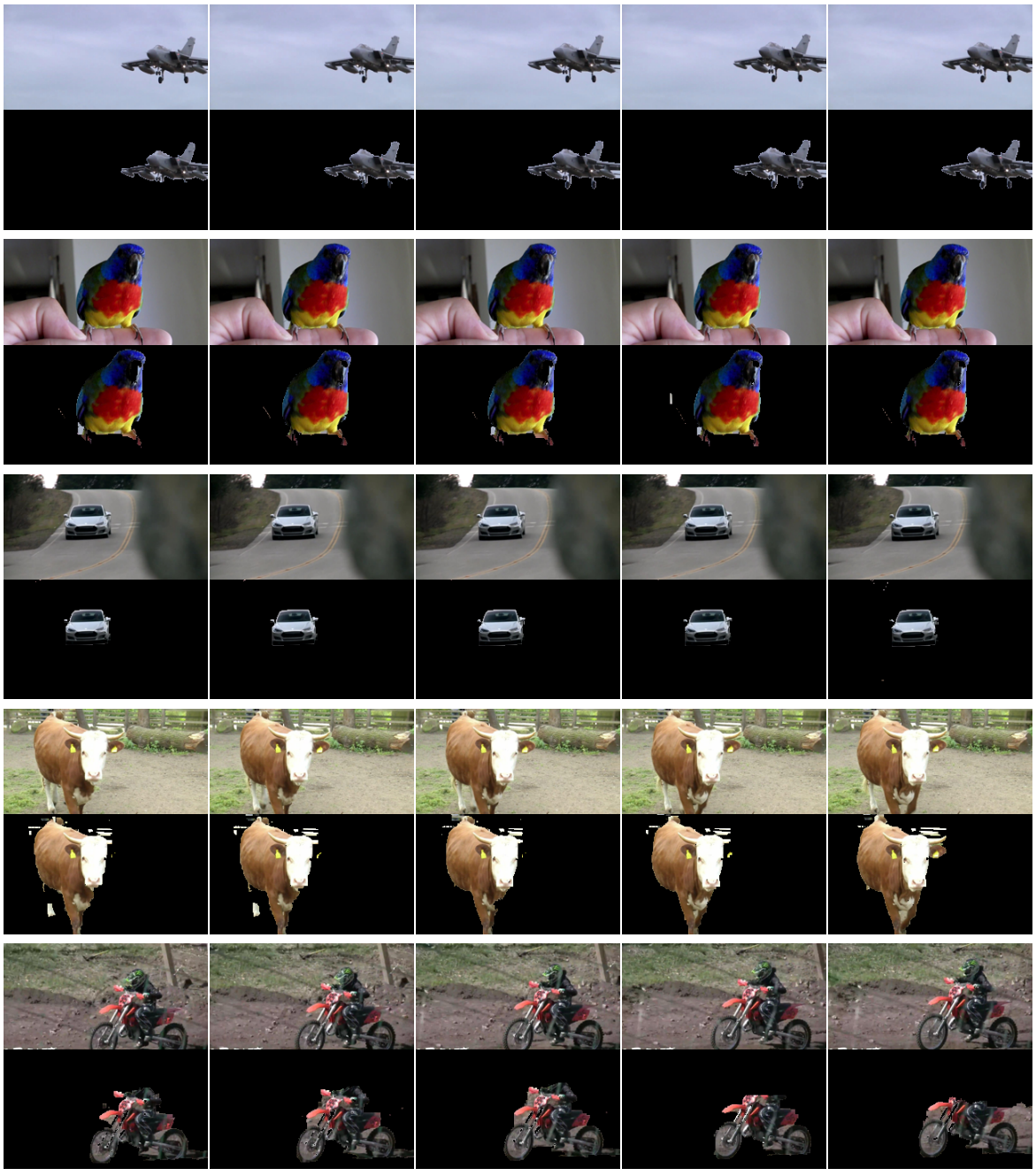


Figure 3.7: Example results on videos tagged as (from top to bottom) "aeroplane", "bird", "car", "cow", and "motorbike" respectively. For each video, we show the original frames (1st row) and the segmentation results obtained after localization (2nd row).

Chapter 4

Object Localization with Appearance Transfer

In this chapter, we develop a method for localizing objects in weakly labeled images/videos by transferring the knowledge from other related objects.

One major weakness of traditional approaches in visual recognition is that even if we have appearance models for 1000 object classes, we have to start from scratch when building the appearance model for the 1001-st object class. This is somewhat unintuitive and unsatisfying – it should be easier to build the appearance model for a new object class if it is related to other known object categories.

Our work is motivated by the following observations. 1) Large datasets with bounding box annotations exist for some object categories, e.g., the 20 objects in PASCAL [6] and a subset of objects in ImageNet [30]. For these object categories (we will call them "familiar objects"), we have access to detectors with reasonably good performance. 2) For most of other object categories (we call them "novel objects"),

fully annotated data are scarce. But it is easy to collect weakly labeled images/videos for them. We use the term "novel objects" to denote objects for which we do not have fully annotated data. It is different from the "novel object" used in object discovery (e.g., [17]). 3) Recent work in text analysis has produced valuable resources on word semantics. For example, a word is represented as a fixed length vector (called "word embedding") in [19]. The embedding vectors of words are learned from large collections of text documents. Semantically related words (e.g., "cat" and "dog") are being mapped closer in this embedding space. The word embedding provides a way for us to infer how two object classes are related. 4) Objects that are semantically close often have similar visual appearances. We acknowledge that some people might not agree with the last point – indeed one can find object categories that are semantically close, but visually very different. But previous work (e.g., [4; 7; 16]) in computer vision has demonstrated that semantic knowledge can still be useful for solving vision-related tasks, even when it is constructed from non-visual information. In this work, we show that it is possible to transfer appearance model from one object class to another based on their semantic relationship in term of the word vectors.

Given a collection of images labeled with an object category (e.g., "car"), our method will output the bounding box of this object in each image. Our method can also be applied in videos. In this case, we are given one single video of the novel object. Our method will treat the frames of the video as the image collection and localize the object in each frame.

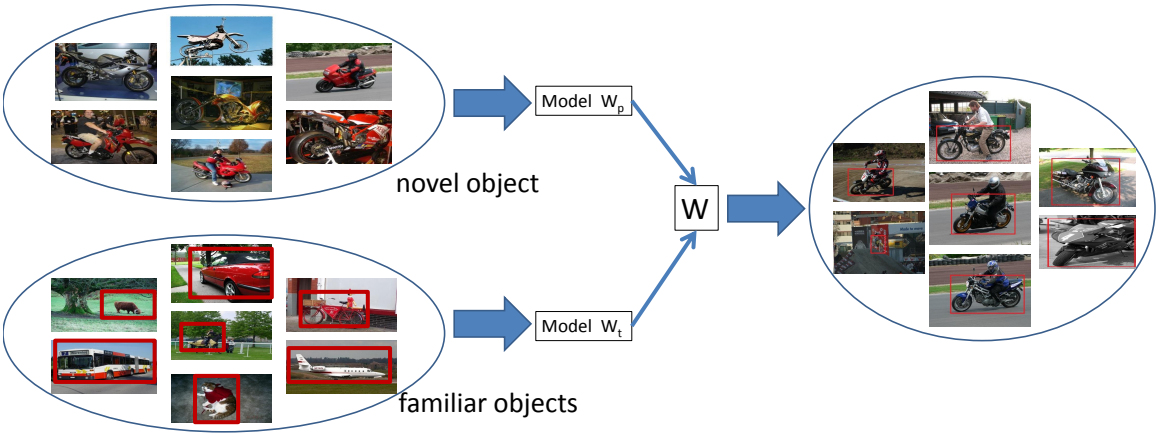


Figure 4.1: An overview of our approach. (Top left) Given a collection of weakly labeled images of a novel object (e.g., motorbike), we learn an appearance model $\mathbf{w}_p$ from the object proposals (see Sec. 4.1.1). (Bottom left) We also have access to fully annotated data (or pre-trained models) for a set of familiar objects, e.g., car, bus, dog, etc. We transfer the knowledge of familiar objects to obtain another appearance model $\mathbf{w}_t$ for the novel object (see Sec. 4.1.2). (Middle) The final appearance model $\mathbf{w}$ for the novel object is a combination of $\mathbf{w}_p$ and $\mathbf{w}_t$. (Right) We can then use $\mathbf{w}$ to localize the novel object in the image collection (see Sec. 4.1.3).

4.1 Our Approach

An overview of our approach is illustrated in Fig. 4.1. To localize a novel object in a collection of weakly labeled images, we build two initial appearance models. The first appearance model is obtained from the image collection using object proposals (Sec. 4.1.1). The second appearance model is obtained by transferring knowledge from other familiar objects (Sec. 4.1.2). Our final appearance model of the novel object is a combination of these two initial models. We then use the final appearance model to localize the novel object in each image of the collection. Our method can also be applied in videos. In this case, we are given one single video of the novel object. Our method will treat the frames of the video as the image collection and localize the object in each frame.

4.1.1 Appearance model from object proposals

Given a collection of weakly labeled images of a novel object, the first step of our approach is to generate a set of object proposals in each image. We use the edge boxes method (see 3.1.1) in [42] for generating bounding boxes as our object proposals.

We assume that the novel object is reasonably salient in most images in the collection. Admittedly, this assumption does not always hold. But we believe this is a reasonable assumption in many cases. For example, if we collect images by querying the name of the novel object from search engines, the novel object tends to be salient in the images returned by search engines.

Based on this assumption, we train an initial model for the novel object from the object proposals in the image collection. We select object proposals with high objectness scores and consider them as positive examples of the novel object. We then select a set of negative examples by randomly generating bounding boxes from images that do not correspond to the novel object. Given these positive and negative examples, we learn an appearance model for this novel object using a linear SVM. Let $\mathbf{x}$ denote the feature vector of an image patch, the appearance model is represented by a parameter vector $\mathbf{w}_p$. The dot product $\mathbf{w}_p^\top \mathbf{x}$ (without loss of generality, we assume a linear SVM model without the bias term) indicates the likelihood of $\mathbf{x}$ being the novel object.

4.1.2 Appearance model from familiar objects

The appearance model and the localization of the novel object appear to be a chicken-and-egg problem. If we have an appearance model of the novel object, we

can use the appearance model to localize the object in an image. Conversely, if we know the ground-truth locations of the novel object in some images, we can simply learn an appearance model of this object. Then we can use the appearance model to localize the object in other images. Sec. 4.1.1 provides one way of getting the appearance model $\mathbf{w}_p$. In this section, we propose another way of constructing the appearance model by transferring knowledge from other familiar objects. First, we use the word vectors associated with the novel object and familiar objects to establish their semantic relatedness. Then we transfer the appearance models of familiar objects based on their relatedness to the novel object.

Word vectors: We use the word vectors learned in [13]. These word vectors are learned in an unsupervised fashion from a large corpus using a neural-network-based language model. The model learns the semantics of words from their local and global context in the corpus. As a result, the model produces a vector space representation for each English word as a D-dimensional vector ($D = 200$ in our experiments). These vectors can then be used as features in various applications in text analysis, e.g., information retrieval, document classification, parsing, etc. In our work, we use the word vectors as a source of semantic knowledge to bridge the familiar and novel objects.

Figure 4.2 shows a visualization of the word vectors by projecting them on a 2D space using t-SNE [38]. We can see that words similar in their semantic meanings are close in term of their word vectors. For example, words corresponding to various music instruments are mapped together in the upper left corner in Fig. 4.2.

Novel object as sparse reconstruction: We are given a set of K familiar object

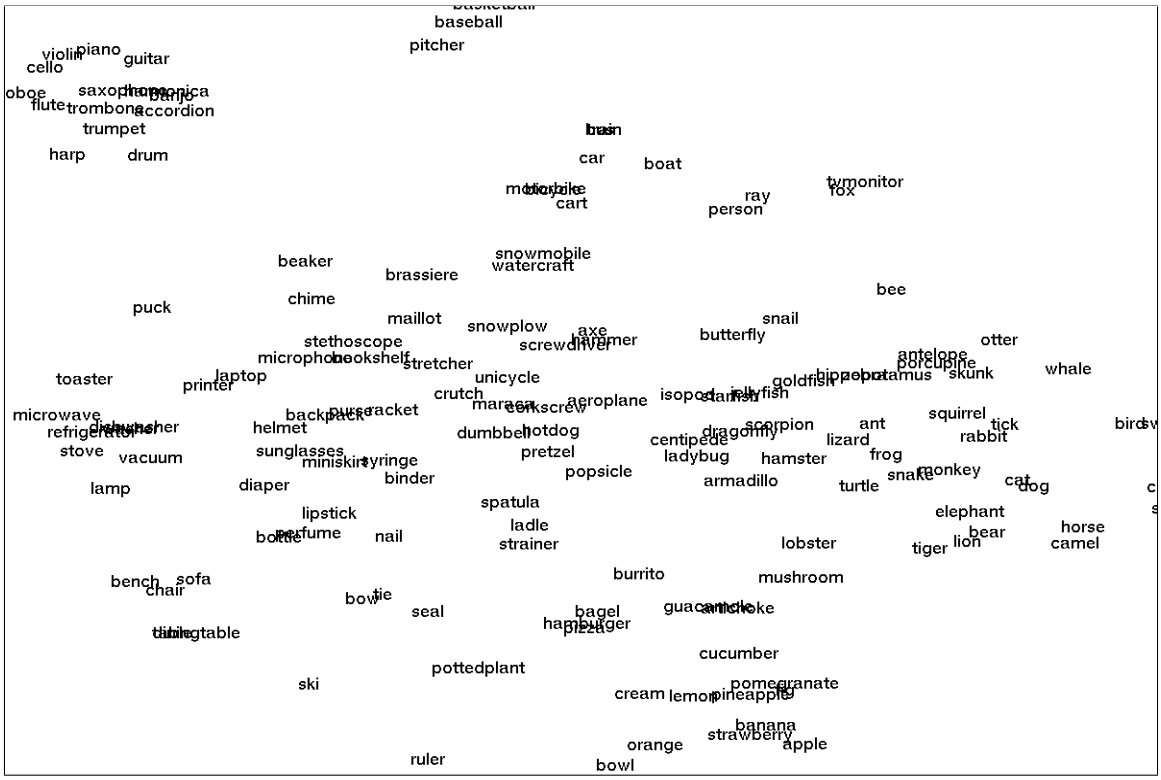


Figure 4.2: (Best viewed in PDF with magnification) Visualization of the word vectors in 2D using t-SNE [38]. The t-SNE algorithm finds a 2D embedding of the word vectors.

classes. We use $\mathbf{v}_i$ to denote the word vector associated with the i-the object class and $\mathbf{u}_i$ to denote the corresponding appearance model. For simplicity, we assume the appearance model (object detector) has a linear form:

$$f_i(\mathbf{x}) = \mathbf{u}_i^\top \mathbf{x} \tag{4.1}$$

where $\mathbf{x}$ represents the feature vector of an image patch. Given an input image, we can apply Eq. 4.1 to sub-windows at various positions and scales to detect the i-th object in the image.

For a novel object class, we denote its word vector as $\mathbf{v}$. Our goal is to obtain an appearance model (we denote it as $\mathbf{w}_t$) for this novel object class. Our approach

is based on two assumptions. First of all, the word vectors and appearance models of objects are related – if two objects i and j are similar in terms of their word vectors $\mathbf{v}_i$ and $\mathbf{v}_j$, they tend to be similar in terms of their appearance models $\mathbf{u}_i$ and $\mathbf{u}_j$. Secondly, for a novel object, we can approximate its word vector $\mathbf{v}$ as a linear combination of those of familiar objects, i.e.,:

$$\mathbf{v} \approx \theta_1 \mathbf{v}_1 + \theta_2 \mathbf{v}_2 + ...\theta_K \mathbf{v}_K \tag{4.2}$$

where the parameters θ_i $(i = 1, 2, ..., K)$ are the coefficients of the linear combination.

We estimate the coefficient vector $\Theta = [\theta_1, \theta_2, ..., \theta_K]^\top$ by solving the following optimization problem:

$$\min_{\Theta > 0} ||\mathbf{v} - (\theta_1 \mathbf{v}_1 + \theta_2 \mathbf{v}_2 + ...\theta_K \mathbf{v}_K)||_2^2 + \lambda ||\Theta||_1 \tag{4.3}$$

The first term in Eq. 4.3 minimizes the reconstruction error of the linear approximation, while the second term minimizes the L_1 norm of the parameter Θ. The L_1 norm will encourage Θ to be sparse, since we prefer to reconstruct the novel object using a small number of familiar objects.

Transferring appearance model: By solving Eq. 4.3, we get the parameter vector $\Theta = [\theta_1, \theta_2, ..., \theta_K]^\top$. If we assume that the semantic relatedness of object classes (in term of word vectors) is similar to that of appearance models, we can use the same Θ to represent the appearance model of the novel object as:

$$\mathbf{w}_t = \theta_1 \mathbf{u}_1 + \theta_2 \mathbf{u}_2 + ...\theta_K \mathbf{u}_K \tag{4.4}$$

Note that we do not require any training data of the novel object in order to get $\mathbf{w}_t$. As long as we have the word vectors of object classes (both familiar and

novel) and pre-trained appearance models for familiar objects, we can use Eq. 4.3 and Eq. 4.4 to compute $\mathbf{w}_t$. In other words, we have transferred the appearance models from familiar objects to the novel object.

4.1.3 Modeling and localizing the novel object

Sec. 4.1.1 and Sec. 4.1.2 provide two different ways of learning the appearance model of the novel object. Let $\mathbf{w}_p$ and $\mathbf{w}_t$ denote the two appearance models learned in Sec. 4.1.1 and Sec. 4.1.2, respectively. Our final appearance model $\mathbf{w}$ for the novel object is a linear combination of these two:

$$\mathbf{w} = \gamma \mathbf{w}_p + \mathbf{w}_t \tag{4.5}$$

where γ is a parameter that controls the relative importance of $\mathbf{w}_p$ and $\mathbf{w}_t$.

Intuitively, the parameter γ should vary depending on the "transferability" of the novel object. If a lot of familiar objects are closely related to the novel object, it should be easier to transfer the appearance model to the novel object. In this case, we like γ to be small, so $\mathbf{w}_t$ will have a higher influence. Conversely, if the novel object is vastly different from all the familiar objects, we like γ to be large. So we do not rely too much on transferring appearance model from the familiar objects.

One way to define the "transferability" of an novel object is to examine the reconstruction error in Eq. 4.3. Let $\Theta^* = [\theta_1^*, \theta_2^*, ..., \theta_K^*]^\top$ be the solution to Eq. 4.3, the reconstruction error is:

$$E(\Theta^*) = ||\mathbf{v} - (\theta_1^* \mathbf{v}_1 + \theta_2^* \mathbf{v}_2 + ...\theta_K^* \mathbf{v}_K)||_2^2 \tag{4.6}$$

We then set $\gamma = \beta E(\Theta^*)$, where β is a free parameter. I.e., our final appearance

model is computed as:

$$\mathbf{w} = \beta \cdot E(\Theta^*) \cdot \mathbf{w}_p + \mathbf{w}_t \tag{4.7}$$

Notice that if a novel object can be easily represented as a linear combination of familiar objects, i.e., it is easy to do the transfer learning, the reconstruction error $E(\Theta^*)$ will be small. In this case, the appearance model $\mathbf{w}_t$ obtained from the transferring learning will have a larger effect in Eq. 4.7.

We can then use this appearance model $\mathbf{w}$ to re-score the object proposals generated in Sec. 4.1.1. Let $\mathbf{x}$ be the feature vector extracted from the image patch of a proposal, we use $\mathbf{w}^\top \mathbf{x}$ to measure the score of this proposal belonging to the novel object. The top scored bounding box in each image will be our localization result.

An interesting special case is when the image collection consists of frames from a single video. This is potentially useful for video retrieval. For example, if we query a novel object, say "tiger" in YouTube. Instead of just returning the videos containing tigers, we can also localize the tiger in each video. If we apply our method on a *single* video of a novel object, we will get an appearance model for the specific instance of the object in this particular video. In other words, our approach can automatically adapt to different videos of the same novel object.

4.2 Experiments

We evaluate our approach on two video datasets (4.2.4 and Sec. 4.2.3). Additionally, we also evaluate our method on one image dataset (Sec. 4.2.2).

4.2.1 Implementation details

We use the 4096 dimensional CNN-feature implemented in Caffe [14] as our feature representation for an object proposal. This feature has been proved to be one of the state-of-the-art feature representations in many visual recognition tasks. In order to construct the set of familiar objects, we use the 200 object classes in [10]. These object models are trained from a subset of the ImageNet images with bounding box annotations using the Caffe-based CNN features. Some of the object classes do not have word vectors associated with them, possibly because they does not appear in the corpus used for learning the word vectors. We filter out those object classes and select 142 familiar object classes in the end.

We set the free parameters of our method by validating over a small set of images/videos. For the images in the PASCAL VOC 2007 dataset, we extract 100 object proposals on each image and set $\lambda = 1$ and $\beta = 0.3$. For videos in the YouTube-Objects dataset, we extract 20 object proposals on each frame and set $\lambda = 1$ and $\beta = 0.1$.

4.2.2 PASCAL VOC 2007

The dataset contains images of 20 object classes from the train+val subsets of PASCAL VOC 2007 dataset. We consider each of them as the novel object and apply our algorithm on the images that contain at least one instance of this novel object. Since the object classes in PASCAL overlap with those of the 142 familiar objects, we remove the novel object class from the set of familiar objects when doing the appearance transfer. For example, when we consider "dog" as the novel object, we

method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	
transfer only	48.32	48.97	17.58	55.25	6.15	32.26	15.85	40.36	**28.54**	70.92	
proposal only	77.31	55.55	62.73	40.88	**21.31**	77.96	72.1	54.9	14.83	68.79	
combined	**78.57**	**63.37**	**66.36**	**56.35**	19.67	**82.26**	**74.75**	**69.13**	22.47	**72.34**	
method	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	avg
transfer only	4.5	15.91	43.55	34.69	13.75	3.26	51.04	28.38	46.74	19.92	31.3
proposal only	29.5	56.29	70.38	74.69	43.18	27.35	47.91	26.2	70.88	67.19	53
combined	**31**	**62.95**	**74.91**	**78.37**	**48.61**	**29.39**	**64.58**	**36.24**	**75.86**	**69.53**	**58.84**

Table 4.1: CorLoc results on the PASCAL VOC 2007 dataset. We compare three different methods: (1st row) using only the appearance model transferred from familiar objects $\mathbf{w}_t$; (2nd row) using only the appearance model from the object proposals $\mathbf{w}_p$; (3rd row) using the combined appearance model $\mathbf{w}$.

remove the "dog" model from the 142 familiar object classes.

We use the CorLoc defined in [5] to measure the performance. It is defined as the percentage of images in which a method correctly localizes the novel object according to the PASCAL criterion $\frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})} > 0.5$, where B_p is the localized bounding box and B_{gt} is a ground-truth bounding box. Table 4.1 shows the CorLoc results of three methods: 1) using only the transferred appearance $\mathbf{w}_t$; 2) using only the appearance model from the object proposals $\mathbf{w}_p$; 3) using the combined appearance model $\mathbf{w}$. The results of using the combined appearance model achieve the best performance on 18 out of the 20 object classes.

Table 4.2 shows the comparison with other published results. Our approach significantly outperforms others. Some examples of our localization results are shown in Fig 4.3.

Figure 4.4 visualizes the Θ parameters obtained via Eq. 4.3. For each novel class, we show the top 10 familiar object classes according to the descending order of their corresponding θ values.

Figure 4.3: Qualitative examples of our approach on the PASCAL VOC 2007 dataset.

method	CorLoc
[15] (image model)	24.6
[32]	30.2
[34]	30.4
[33]	32.0
[31]	36.2
[3]	38.8
ours	58.84

Table 4.2: Comparison with previous work on the PASCAL VOC 2007 dataset in term of the average CorLoc.

4.2.3 YouTube-Objects

The Youtube-Objects dataset [24] consists of videos of 10 object classes. For each class, bounding box annotations are provided for one frame per shot for 100-290 shots. We apply our method on each video in the dataset by considering the frames in this

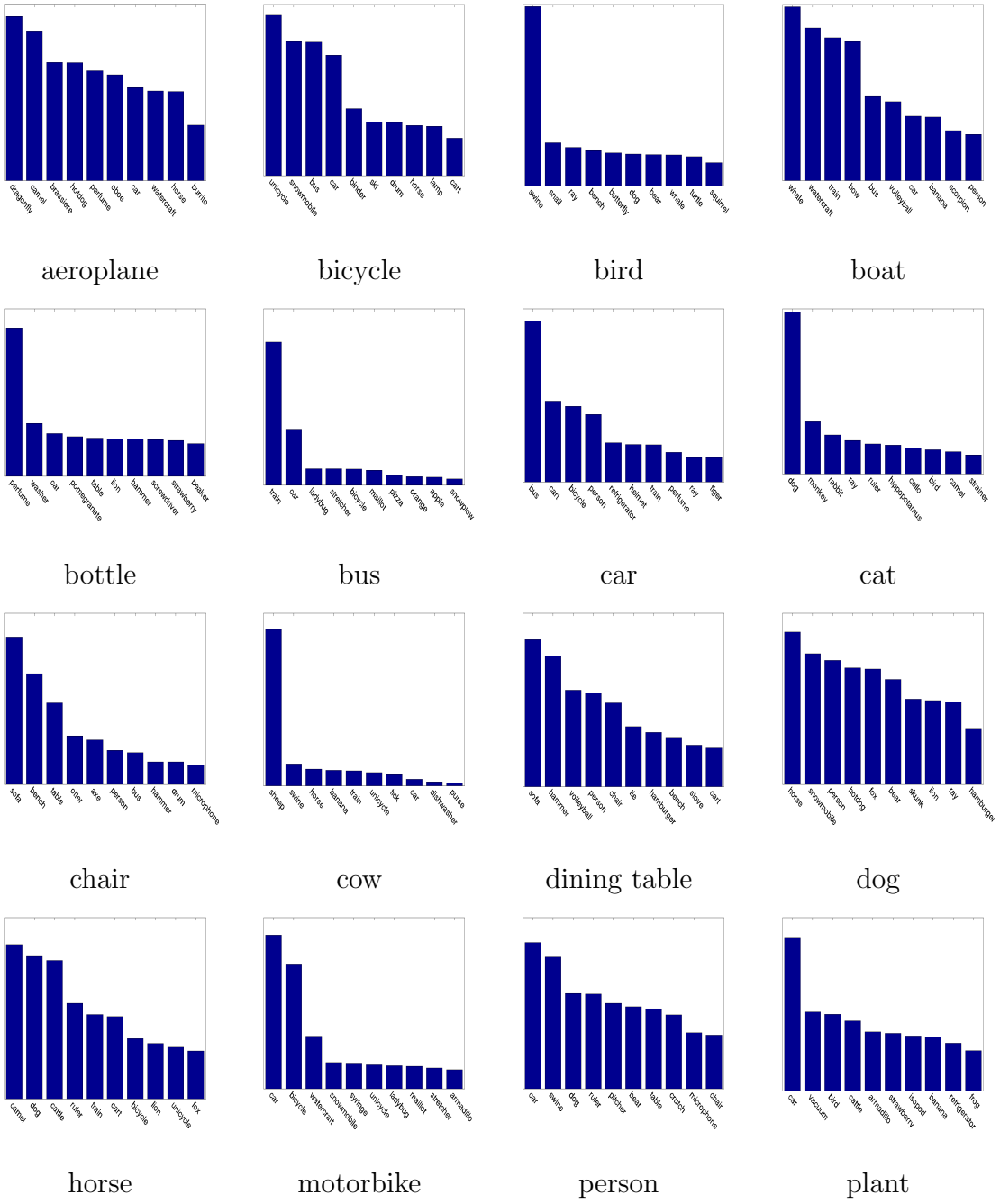


Figure 4.4: (Best viewed in PDF with magnification) Visualization of the Θ parameters for novel object classes. For each novel object class, we show the top 10 familiar objects with the corresponding θ values.

video as the image collection. Similarly, we remove the novel object class from the set of familiar objects when doing the appearance transfer. Only frames with annotations are considered in the evaluation. In Table 4.3, we compare our results with previous work that uses the same dataset.

method	aero	bird	boat	car	cat	
[15] (video)	25.12	31.18	27.78	38.46	41.18	
[22]	**65.4**	**67.3**	38.9	65.2	**46.3**	
transfer only	35.27	10.75	31.75	30.77	19.66	
proposal only	51.69	54.84	32.54	**85.71**	14.53	
combined	56.04	30.11	**39.68**	**85.71**	24.79	
method	cow	dog	horse	bike	train	avg
[15] (video)	28.38	33.91	35.62	23.08	25	30.97
[22]	40.2	**65.3**	48.4	39	25	50.1
transfer only	83.78	26.96	50.68	50.56	46.43	38.66
proposal only	75.68	55.65	53.42	51.69	39.29	51.5
combined	**87.83**	55.65	**60.27**	**61.8**	**51.79**	**55.37**

Table 4.3: CorLoc results on the YouTube-Objects dataset. Similar to the PASCAL VOC 2007 dataset, we compare three different methods: (3rd row) using only the appearance model transferred from familiar objects $\mathbf{w}_t$; (4th row) using only the appearance model from the object proposals $\mathbf{w}_p$; (5th row) using the combined appearance model $\mathbf{w}$. We also compare with previous work [15] (1st row) and [22] (2nd row) that uses the same dataset.

4.2.4 YouTube-Objects-Subset

We also evaluate our method on the subset of the YouTube-Objects dataset (same as 3.2.1) collected in [37]. This dataset contains ground-truth segment-level object annotations on all frames in many video shots. The results on this dataset are shown in Table 4.4. Fig. 4.5 shows some qualitative results on this dataset.

method	aero	bird	boat	car	cat	
transfer only	40.34	43.86	**40.41**	28.42	26.35	
proposal only	42.23	51.24	29.54	**67.76**	14.75	
combined	**45.74**	**55.47**	39.51	58.75	**26.51**	
method	cow	dog	horse	bike	train	avg
transfer only	47.15	34.37	28.67	26.12	24.28	34
proposal only	50.2	**47.02**	22.18	16.44	18.84	36.02
combined	**55**	43.51	**33.71**	**32.76**	**25.63**	**41.66**

Table 4.4: CorLoc results of different methods on the YouTube-Objects-Subset dataset.

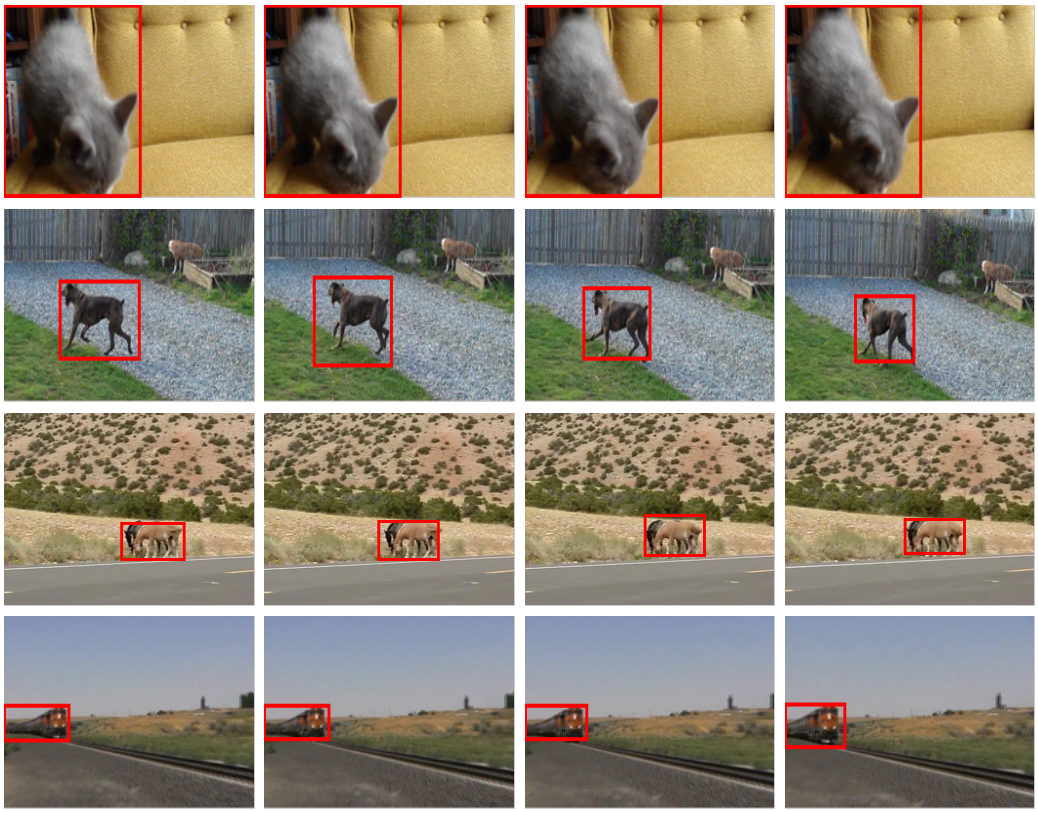


Figure 4.5: Qualitative examples of localization produced by approach on videos tagged as (from top to bottom) "cat", "dog", "horse", and "train" respectively.

Chapter 5

Conclusion and Future Work

In this thesis, we have presented novel methods to efficiently localize and segment the object of interest in weakly labeled images and videos.

We have made a threefold contribution to the research in the area of weakly supervised object localization. First, we introduce a video-specific appearance model to localize the object in weakly labeled videos. Second, we improve the performance of video-specific appearance model framework by incorporating a temporal consistency constraint between consecutive frames of a video. Third, we propose a method to transfer the appearance of reliable known object detectors to build the appearance model of the object of interest. We perform extensive experiments on a video dataset and a popular image dataset to show the effectiveness of our proposed methods.

There are many possible directions for future work. First, we would like to extend our methods to handle multiple objects in images/videos. Second, we would like to use our methods for large-scale incremental learning of object from Internet data.

Bibliography

[1] B. Babenko, M.-H. Yang, and S. Belongie. Robust object tracking with online multiple instance learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1619–1632, 2011.

[2] J. Berclaz, F. Fleuret, E. Türetken, and P. Fua. Multiple object tracking using k-shortest paths optimization. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(9):1806–1819, 2011.

[3] R. G. Cinbis, J. Verbeek, and C. Schmid. Multi-fold mil training for weakly supervised object localization. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[4] J. Deng, A. C. Berg, K. Li, and L. Fei-Fei. What does classifying more than 10,000 image categories tell us? In *European Conference on Computer Vision*, 2010.

[5] T. Deselaers, B. Alexe, and V. Ferrari. Weakly supervised localization and learning with generic knowledge. *International journal of computer vision*, 100(3):275–293, 2012.

[6] 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*, 88(2):303–338, 2010.

[7] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2009.

[8] 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*, 32(9):1672–1645, 2010.

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

[10] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[11] S. Hare, A. Saffari, and P. H. Torr. Struck: Structured output tracking with kernels. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 263–270. IEEE, 2011.

[12] 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*, pages 198–208, 2012.

[13] E. H. Huang, R. Socher, C. D. Manning, and A. Y. Ng. Improving word representations via global context and multiple word prototypes. In *50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 873–882, 2012.

[14] 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.

[15] A. Joulin, K. Tang, and L. Fei-Fei. Efficient image and video co-localization with frank-wolfe algorithm. In *European Conference on Computer Vision*, 2014.

[16] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 951–958, 2009.

[17] Y. J. Lee and K. Grauman. Object-graphs for context-aware category discovery. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010.

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

[19] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed rep-

resentations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*. MIT Press, 2013.

[20] 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*, pages 392–405. 2010.

[21] Y. Pang and H. Ling. Finding the best from the second bests-inhibiting subjective bias in evaluation of visual tracking algorithms. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 2784–2791. IEEE, 2013.

[22] A. Papazoglou and V. Ferrari. Fast object segmentation in unconstrained video. In *IEEE International Conference on Computer Vision*, 2013.

[23] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet. Color-based probabilistic tracking. In *Computer visionECCV 2002*, pages 661–675. Springer, 2002.

[24] 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*, pages 3282–3289. IEEE, 2012.

[25] D. Ramanan and D. A. Forsyth. Finding and tracking people from the bottom up. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2003.

[26] D. Ramanan and D. A. Forsyth. Using temporal coherence to build models of animals. In *IEEE International Conference on Computer Vision*, 2003.

[27] D. Ramanan, D. A. Forsyth, and A. Zisserman. Strike a pose: Tracking people

by finding stylized poses. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 271–278, 2005.

[28] M. Rohrbach, M. Stark, and B. Schiele. Evaluating knowledge transfer and zero-shot learning in a large-scale setting. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1641–1648, 2011.

[29] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM Transactions on Graphics (TOG)*, volume 23, pages 309–314. ACM, 2004.

[30] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. Berg, and L. Fei-Fei. ImageNet large scale visual recognition challenge. arXiv, 2014.

[31] Z. Shi, T. M. Hospedales, and T. Xiang. Bayesian joint topic modelling for weakly supervised object localization. In *IEEE International Conference on Computer Vision*, 2013.

[32] P. Siva, C. Russell, and T. Xiang. In defence of negative mining for annotating weakly labelled data. In *European Conference on Computer Vision*, pages 594–608. Springer, 2012.

[33] P. Siva, C. Russell, T. Xiang, and L. Agapito. Looking beyong the image unsupervised learning for object saliency and detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[34] P. Siva and T. Xiang. Weakly supervised object detector learning with model drift detection. In *IEEE International Conference on Computer Vision*, 2011.

[35] 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*, pages 1250–1257, 2012.

[36] K. Tang, V. Ramanathan, L. Fei-Fei, and D. Koller. Shifting weights: Adapting object detectors from image to video. In *Advances in Neural Information Processing Systems*, pages 638–646, 2012.

[37] 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*, pages 2483–2490, 2013.

[38] L. van der Maaten and G. E. Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

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

[40] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *Acm computing surveys (CSUR)*, 38(4):13, 2006.

[41] D. Zhang, O. Javed, and M. Shah. Video object segmentation through spatially accurate and temporally dense extraction of primary object regions. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 628–635, 2013.

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