Multiple Frames Matching for Object Discovery in Video

Otilia Stretcu
otiliastr@gmail.com
Computer Laboratory
University of Cambridge, UK

Marius Leordeanu
marius.leordeanu@imar.ro
Institute of Mathematics of the
Romanian Academy
Teamnet International, Romania

Abstract

Automatic discovery of foreground objects in video sequences is an important problem in computer vision with applications to object tracking, video segmentation and classification. We propose an efficient method for the discovery of object bounding boxes and the corresponding soft-segmentation masks across multiple video frames. We offer a graph matching formulation for bounding box selection and refinement using second and higher order terms. Our objective function takes into consideration local, frame-based information, as well as spatiotemporal and appearance consistency over multiple frames. First, we find an initial pool of candidate boxes using a novel and fast foreground estimation method in video, based on Principal Component Analysis. Then, we match the boxes across multiple frames using pairwise geometric and appearance terms. Finally, we refine their location and soft-segmentation using higher order potentials that establish appearance regularity over multiple frames. We test our method on the large scale YouTube-Objects dataset and obtain state-of-the-art results on several object classes.

1 Introduction

The unconstrained discovery of objects in video sequences is an open problem in computer vision, with potential impact on many different tasks, such as object tracking, weakly supervised learning of category models, robotic systems, video mining and classification. In this paper we focus on the problem of co-localization, which is that of finding object bounding boxes automatically. We also propose an efficient method for rapidly estimating object soft-segmentation masks, for better localization and shape estimation.

Usually, video sequences contain objects that display relatively stable geometric and appearance patterns over time. Their change in shape and appearance is often smooth and coherent between frames that are not very far away from each other. The mild transition between nearby frames should be exploited for efficient discovery of objects. There are several assumptions that could be made in practice, with minimal loss of information: single objects stand out against the background. They tend to have their own unique distribution of colors and texture. Their shape obeys certain grouping properties, with a smooth, strong boundary response along its edges. Foreground objects are also more difficult to model than their backgrounds, as their movements and appearance are more complex. They are likely to

Pages 186.1-186.12
DOI: https://dx.doi.org/10.5244/C.29.186

occupy a relatively small region of the scene and are often close to the image center. These observations constitute the basis of our approach (Sec. 2).

The task of object discovery in video is strongly related to co-segmentation [12, 13, 15, 16, 27, 28, 31] and weakly supervised localization [9, 22, 29]. The task has been tackled for more than a decade in computer vision, with initial works mainly based on local feature matching and detection of their co-occurring patterns [18, 20, 24, 30]. Our approach is also based on matching, and has at its core an integer quadratic formulation that is related to the graph matching and MAP inference literature [6, 19]. Note that graph matching has been used, under different forms, in related problems for weakly supervised learning and discovery, such as [21]. Our method is related to the works mentioned and differs in important ways: it rapidly discovers and establishes bounding box matches across multiple frames. It also encourages spatiotemporal and appearance uniformity in order to improve box locations and produce high quality soft-segmentation masks. This is also different from recent works [14, 26] that discover object tubes through only through links between consecutive frames, without refining their location. They are more vulnerable to temporary occlusions, strong blur and other appearance or geometric noises.

Here we introduce an efficient method for the discovery of objects in video, composed of three main stages: 1) find potential bounding boxes; 2) match them across multiple frames; 3) obtain a soft segmentation mask for each frame and refine the boxes' locations by iteratively shifting their centers towards regions of maximum density of foreground pixels. We make the following main contributions:

1. A novel formulation with efficient discrete and continuous optimization for joint automatic selection and refinement of object bounding boxes in video. Our approach encourages appearance, geometric and spatiotemporal consistency over multiple frames, with a formulation that considers relations between neighboring as well as farther away frames. This brings robustness against the common difficulties of complete or partial occlusion, drifting and missing data.

2. A fast method for estimating foreground and occlusion regions based on Principal Component Analysis of the video content. Different from classical background subtraction approaches, our novel method estimates a linear subspace model of the video content and manages to handle cases of slowly moving or changing backgrounds.

2 Method Overview

Our goal is to automatically discover the main foreground object that appears in a video sequence. We aim to estimate both its bounding box and its soft-foreground mask. We formulate bounding box selection and location refinement as a discrete-continuous optimization task. While solving the problem, we also generate soft-segmentation object masks. Our approach is related to integer programming techniques from graph matching and MAP inference [19] in graphical models, as well as co-segmentation methods in video [28]. The algorithm consists of three main phases, as described next. First, we rapidly generate initial foreground-background segmentations and form a pool of potential object bounding boxes. Next, we match the boxes from frames that are nearby or farther away in time, in order to encourage and preserve appearance and spatiotemporal consistency. Finally, we refine both the bounding boxes locations and the object co-segmentations over the sequence. All stages aggregate information from multiple frames in the video:

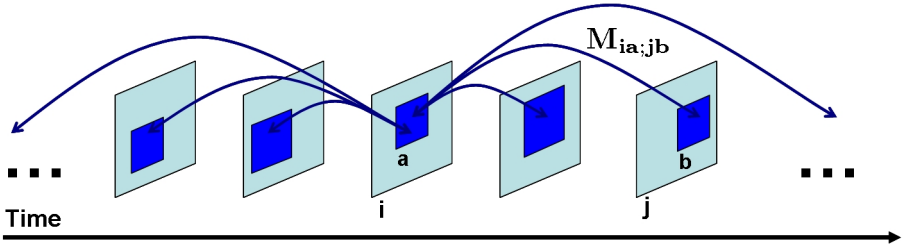

Figure 1: The structure of our box-matching formulation: we allow links between neighboring frames (e.g. *i* and *j*), as well as farther away ones, in order to better preserve the appearance and shape consistency between the matched boxes over time (e.g. boxes *a* and *b*). This results into a quadratic assignment problem that can be optimized efficiently.

1) Initial segmentation and generation of candidate bounding boxes: we rapidly estimate the foreground object segmentation using our novel method, termed VideoPCA (Section 4.1), based on Principal Component Analysis (PCA) of the entire video content. It works in conjunction with a simple pixel-wise foreground/background inference routine using color distributions, termed SoftSeg (Sec. 4.2). VideoPCA is able to return in realtime ($50-100$ fps in Matlab on a 2.2GHz Laptop PC) regions that are likely to belong to objects, foreground or the occluding regions caused by their movements. The procedure differs from classical background subtraction approaches [1, 7] in that it is able to handle many cases of moving or changing backgrounds. While the regions detected are not always correct, the output of VideoPCA is very effective when fed into the recent method for generating bounding box proposals based on image contours [33] (termed Edge Boxes). VideoPCA object soft-segmentations are also used to filter out boxes with a relatively few foreground pixels inside, based on a very permissive threshold.

2) Efficiently matching and selecting boxes over multiple frames: we formulate the matching and selection of bounding boxes as a quadratic assignment problem (QAP) with pairwise constraints (Fig. 1), directly related to recent formulations of graph matching and MAP inference with Integer Quadratic Programming (IQP) [4, 6, 19]. We use both unary properties that measure the quality of the candidate boxes and are computed per frame, and pairwise properties that encourage spatiotemporal and appearance consistency over multiple connected frames. We look at how the individual boxes separate themselves from the background in terms of velocity and appearance and how well they match each other in terms of geometry (size and location) and appearance. We allow pairwise constraints (links) between boxes that are several frames apart. Each frame is connected to its k forward and k backward neighbors (in our case $k = 10$). We consider only every 5-th frame in a sequence, thus we connect boxes that are up to 50 frames apart.

3) Localization and segmentation refinement: after matching and selecting object bounding boxes, we re-estimate, for each given frame, a foreground-segmentation mask using color information from the matched boxes and their surrounding background (Sec. 4.2). Once the soft-segmentation is re-estimated we apply the Mean-Shift algorithm [5] in order to move the current matched bounding boxes towards the location with highest density of foreground pixels. After convergence, we again estimate the segmentation using the new box positions. Our mathematical formulation and detailed algorithm follow next.

3 Mathematical Formulation

Given a video shot V as a sequence of temporally ordered frames $V = \{I_1, I_2, ..., I_n\}$, the goal is to discover a potential object of interest and output its bounding box and soft-segmentation mask. For each frame I_i we have a pool of n_i candidate bounding boxes B_{ia}'s, obtained automatically. For each box a we store its xy location in image i in θ_{ia}. Let x_{ia} be an indicator variable corresponding to bounding box B_{ia} (- the a-th bounding box of frame i), such that x_{ia} is 1 if the bounding box B_{ia} is selected, and 0 otherwise. The indicator variables are arranged in vector form $\mathbf{x}$ such that its ia-th element corresponds to x_{ia}. We impose the constraint that a single box can be selected per frame: $\sum_a x_{ia} = 1$. Thus, vector $\mathbf{x}$ represents a discrete solution that indicates which box is selected. Similarly, we keep the continuous location parameters in a global vector θ, with θ_{ia} being the parameters of box B_{ia}. We simultaneously optimize for both $\mathbf{x}$ and θ. Our mathematical formulation (Eq. 2), considers the joint problem of bounding box selection and location refinement. It is a discrete-continuous optimization problem with second-order terms for matching multiple frames and higher order terms for refining the bounding boxes. These potentials are defined below:

Pairwise Potential: we include both second order relations between boxes and unary features per box into the pairwise potentials. The unary cues capture how likely is a given box to represent the foreground object given its properties (e.g. appearance, speed) vs. its surrounding background. We estimate the average speed of a given region with an efficient state-of-the-art dense flow method [32]. At the second-order level, we consider spatiotemporal and appearance consistency cues: how well boxes from different frames match in appearance (using Euclidean distances between their HOG descriptors [8]) and geometry (e.g. area, aspect ratio, overlap and location). We form a matrix $\mathbf{M}$, whose elements $M_{ia;jb}$ use these cues and estimate how well box a from frame i matches box b from frame j, and also how likely they are to represent foreground objects. These terms have the following form:

$$M_{ia;jb} = \exp(\mathbf{w}^{\mathbf{T}} \mathbf{g}_{ia;jb}), \tag{1}$$

where $\mathbf{g}_{ia;jb} = [(f_{ia} + f_{jb}), (v_{ia} + v_{jb}), (c_{ia} + c_{jb}), m_{ia;jb}, o_{ia;jb}, d_{ia;jb}, s_{ia;jb}, r_{ia;jb}]$, such that: 1) f_{ia} (and f_{jb} respectively) measure the absolute difference between the average foreground soft-segmentation values in box a (and b respectively) vs. average foreground values in the surrounding background from its frame i (and j respectively). 2) Similarly, v_{ia} and v_{jb} measure the absolute difference in the relative mean speed between the box and the surrounding background, computed using the DeepFlow method [32]. 3) c_{ia} and c_{jb} measure the distance between the box center and the image center. 4) $m_{ia;jb}$ reflects the quality of the match between the standard HOG descriptor of box a and that of box b. 5) $o_{ia;jb}$ measures the overlap-over-union between the boxes. 6) $d_{ia;jb}$ is the distance between the boxes' centers. 7) $s_{ia;jb}$ is the ratio of the difference between the boxes' areas to the maximum of the two areas. 8) $r_{ia;jb}$ estimates the change in shape, as difference between the boxes' aspect ratios.

We learn $\mathbf{w}$ such that $\exp(\mathbf{w}^{\mathbf{T}} \mathbf{g}_{ia;jb})$ approximates a target $t = 1$ if the matched pair is correct and is equal to a small positive value ($t = 0.1$) otherwise. We want $\exp(\mathbf{w}^{\mathbf{T}} \mathbf{g}_{ia;jb}) \approx t$. We take the log on both sides $\mathbf{w}^{\mathbf{T}} \mathbf{g}_{ia;jb} \approx \log t$ and obtain a linear system of equations over a set of training samples. We estimate the parameters using ridge regression (least squares minimization with L2-norm regularization). Since the number of parameters is relatively small ($= 8$) overfitting is unlikely, thus we use a small sample of 100 positively matched box pairs (manually selected) and 300 randomly selected pairs for the negative class.

Higher-order Potential: in order to improve the location of the initial bounding boxes

(which might not be optimally selected by Edge Boxes) we use higher order terms, one for each frame, that model foreground-background appearance over the multiple frames connected to it. We estimate foreground and background color probability distributions from the $2k+1$ bounding boxes and their frames in the neighborhood of the current frame i (including itself) and estimate the foreground segmentation using the SoftSeg method. Then, the higher order term $H_i(\mathbf{x},\theta) = \lambda c_k(i)$ measures the difference between average foreground segmentation values inside the box defined by $(\mathbf{x},\theta)$ and outside of it using the estimated color distribution from $2k+1$ multiple frames. The higher order terms estimate a more consistent segmentation by computing foreground-background models from the current solution $(\mathbf{x},\theta)$ over multiple frames connected to frame i. Note that $H_i(\mathbf{x},\theta)$ is sensitive only to the elements in θ that belong to the matched bounding boxes - an important aspect for efficient optimization.

Optimization Problem: the problem becomes one of joint box matching and location refinement over multiple frames, in which we optimize over both $\mathbf{x}$ and θ:

$$\begin{aligned}(\mathbf{x}^*,\theta^*) &= \underset{x,\theta}{\operatorname{argmax}}(\mathbf{x}^T\mathbf{M}\mathbf{x} + \sum_{i=1}^{n} H_i(\mathbf{x},\theta)) \\ \text{s.t.} &\quad \sum_a x_{ia} = 1 \;\; \forall i, \;\; \mathbf{x} \in \{0,1\}^n.\end{aligned}$$

Ideally the two terms, the quadratic discrete matching term $\mathbf{x}^T\mathbf{M}\mathbf{x}$ and the continuous function $\sum_{i=1}^{n} H_i(\mathbf{x},\theta)$ should be optimized simultaneously, but that is computationally prohibitive. We adopt a two stage approach, as briefly presented previously. The first stage performs discrete optimization in which the quadratic function is optimized by finding the correct frame-to-box matches, given a fixed θ - the initial bounding box locations. In the second stage, when $\mathbf{x}^*$ is fixed, the location θ is refined by the non-parametric Mean-Shift in order to locally optimize the foreground pixels density.

4 Algorithm

The structure of our method is presented in Algorithm 1. After finding the initial candidate bounding boxes we match them across frames using IPFP [19] (Algorithm 2), an efficient algorithm for graph matching and MAP inference. Step 1 of IPFP can be optimally solved in linear time by picking, for each site i the label a^* that maximizes $(\mathbf{x_t}^\top\mathbf{M})_{ia}$ over all boxes a belonging to frame i. Step 2 can also be solved efficiently with closed-form solution, as the line search becomes an optimization of a quadratic function in one dimension. Starting from a uniform solution, IPFP converges in $5-10$ iterations and quickly selects bounding boxes of state-of-the-art quality on several classes. After the final boxes are selected we proceed to optimize the continuous θ^*. We improve the location of the bounding boxes to maximize the higher order appearance/co-segmentation terms, using Algorithm 3 (Stage 3 of our approach). In practice Stage 3 improves over Stage 2 by a significant margin. (Table 2). The pseudo-codes of our methods are presented in Algorithms 1, 2, 3.

Algorithm 1 Multiple Frames Matching for Object Discovery

Input: video sequence $V = \{I_1, I_2, ..., I_n\}$.
Stage 1:
create pool of candidate bounding boxes B (Edge Boxes with VideoPCA).
initialize potentials $\mathbf{M}$ and H_i over the entire video sequence.
set initial $\mathbf{x}_{ia} \leftarrow 1/n_i, \forall\ (i,a)$.
Stage 2:
$\mathbf{x}^* \leftarrow \text{argmax}_{\mathbf{x}}\, \mathbf{x}^T\mathbf{M}\mathbf{x}$ (Alg. 2).
Stage 3:
find θ^*_{ia} for each selected box a in each frame i that optimizes $H_i(\theta, \mathbf{x}^*)$ (Alg. 3).
compute soft-segmentation masks (Alg. 3).
return $\mathbf{x}^*, \theta^*$ and soft-segmentation masks.

Algorithm 2 Multiple Frames Matching with IPFP

Initialize $\mathbf{x}_0$, $t \leftarrow 0$.
repeat
Step 1: $\mathbf{y}^* \leftarrow \text{argmax}\, \mathbf{x}_t^\top \mathbf{M}\mathbf{y}$ s.t. $\sum_a y_{ia} = 1$, $\mathbf{y} \in \{0,1\}^n$.
if $\mathbf{x}_t^\top \mathbf{M}(\mathbf{y} - \mathbf{x}_t) = 0$ stop.
Step 2: $\alpha^* \leftarrow \text{argmax}\, S((1-\alpha)\mathbf{x}_t + \alpha\mathbf{y}^*)$, $\alpha \in [0,1]$.
Step 3: $\mathbf{x}_{t+1} \leftarrow (1-\alpha^*)\mathbf{x}_t + \alpha^*\mathbf{y}^*$, $t \leftarrow t+1$.
until convergence.
$\mathbf{x}^* \leftarrow \mathbf{x}_t$.
return $\mathbf{x}^*$.

4.1 Background Subtraction by VideoPCA

We present our novel method based on Principal Component Analysis for rapidly estimating the frame pixels that are more likely to belong to the foreground object [1]. We make the observations that usually the object of interest has more complex and varied movements than its background scene, it often causes occlusions, it has a distinctive appearance, it usually occupies less space. All these differences make the foreground more difficult to model with a simple PCA-based scheme, than the background. Since the background contains the bulk of the information in the video and varies less than the foreground, we expect that it is better captured by the lower dimensional subspace of the frames from a given video shot. Several competitive methods for detecting potentially interesting, foreground objects as salient regions in images are also based on the general idea that objects are different from their backgrounds and that this foreground-background contrast can be best estimated by computing global image statistics over the input test image or by learning a background prior [2, 3]. For example, the successful spectral residual approach [10] is an efficient method that finds interesting regions in an image by looking at the difference between the average Fourier spectrum of the image, estimated using filtering, and the actual raw spectrum. The more recent discriminative regional feature integration approach (DRFI) [11], learns a background prior and finds objects that distinguish themselves from the global background using regression.

Different from the current literature, our method takes advantage of the spatiotemporal consistency that naturally exists in video shots and learns, in an unsupervised manner using PCA, a linear subspace of the background. It takes advantage of the redundancy and also of

[1] Code available at: https://sites.google.com/site/multipleframesmatching/

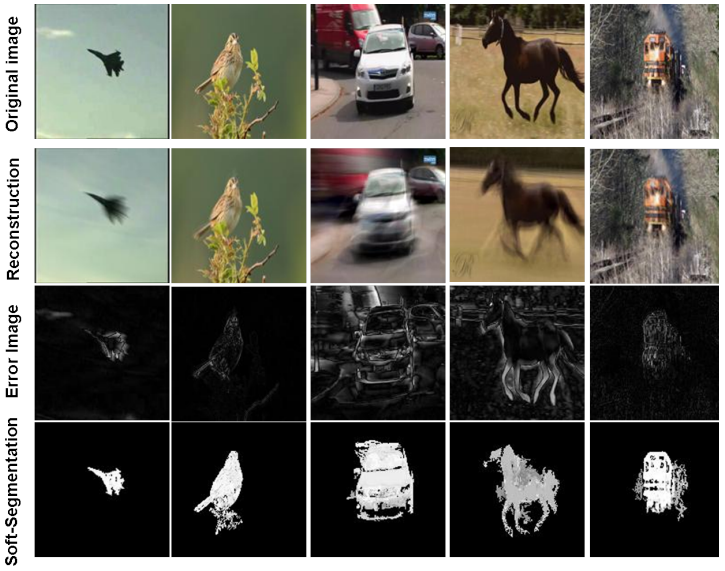

Figure 2: First row: original images. Second row: reconstructed images with PCA and the first 8 principal components chosen. Third row: error image between the original and the reconstructed. Fourth row: final foreground segmentation computed with the SoftSeg method using color models obtained from foreground regions estimated with VideoPCA.

Algorithm 3 Box Location Refinement over Multiple Frames

for $i = 1, \ldots, n$ **do**
  Step 1: from $\mathbf{x}^*$ and the frames connected to frame i estimate a color model (Sec. 4.2).
  Step 2: compute the foreground segmentation of frame i using the color model from 1.
  Step 3: improve the location θ_{ia} of the current box a in frame i using Mean-Shift on the estimated foreground segmentation, until convergence.
end for

the rich information available in the whole video sequence. Relative to the main subspace of variation, the object is expected to be an outlier, an element of noise, harder to reconstruct. Note that every single change in appearance from one frame to the next, and every rapid movement, would be hard to capture by blindly using PCA on whole frames. We used this intuition to find pixels belonging to potential foreground objects and their occlusion regions, by a method related to background subtraction. In our case the background is, in fact, the image reconstructed in the reduced subspace. Let the principal components be $\mathbf{u}_i$, $i \in [0 \ldots n_u]$ (we used $n_u = 8$) and the reconstructed frame $\mathbf{f}$ be $\mathbf{f}_r \approx \mathbf{f}_0 + \sum_{i=1}^{n_u} ((\mathbf{f} - \mathbf{f}_0)^\top \mathbf{u}_i)\mathbf{u}_i$. We obtained the error image $f_{diff} = |\mathbf{f} - \mathbf{f}_r|$. We notice that the difference image enhances the pixels belonging to the foreground object or to occlusions caused by the movement of this object (Fig. 2). By smoothing these regions with a large enough Gaussian and then thresholding, we obtain masks whose pixels tend to belong to objects rather than to background. Then, by applying another large and centered Gaussian to the obtained masks, we get a refined mask that is more likely to belong to the object of interest. Next, by accumulating such masks

Table 1: Comparison to recent state-of-the-art methods on Youtube-Objects Dataset. Note that we obtain state-of-the-art results on four classes by a significant margin of at least 12% on each, while being on others close to or better than [23].

Method	aeroplane	bird	boat	car	cat	cow	dog	horse	motorbike	train	Average
Our method	38.3	62.5	**51.1**	54.9	**64.3**	**52.9**	44.3	43.8	**41.9**	**45.8**	49.9
[25]	51.7	17.5	34.4	34.7	22.3	17.9	13.5	26.7	41.2	25.0	28.5
[23]	65.4	67.3	38.9	65.2	46.3	40.2	65.3	48.4	39.0	25.0	50.1
[14]	25.1	31.2	27.8	38.5	41.2	28.4	33.9	35.6	23.0	25.0	31.0
[26]	57.5	39.8	29.4	52.0	17.3	45.1	38.4	22.9	10.5	14.6	32.8

over the entire video shot we can construct a relatively stable, robust foreground-background color model in order to estimate a soft segmentation, using the SoftSeg method presented next. While the mask is not optimal, it is computed at a high speed ($50-100$ fps), and, in our extensive experiments, it was always useful at obtaining high quality candidate bounding boxes and initial foreground soft-segmentations (see Fig. 2 and Table 2).

4.2 Soft-Segmentation

Foreground-background segmentation should separate well the object of interest from the background, based on statistical differences between the object and its surroundings. Here we present a simple and effective way (termed SoftSeg) of producing soft object masks by capturing global object and background color properties, related to the method for soft foreground segmentation in static images presented in [17]. For both the object, represented as a bounding box, and the background, considered as a border surrounding the bounding box (of thickness half the size of the bounding box), we estimate the empirical color distributions, such that for a given color c the foreground color likelihood is estimated as $p(c|F) = N_c^{(F)}/N^{(F)}$, where $N_c^{(F)}$ is the number of times the color c appeared inside the foreground region and $N^{(F)}$ is the total number of foreground pixels. Similarly, we compute the background color likelihood $p(c|B)$. Given the two distributions we estimate the probability of foreground for each pixel of color c in the image, using Bayes rule with equal priors: $p(F|c) = p(c|F)/(p(c|F)+p(c|B))$. In order to obtain the soft foreground segmentation mask, we simply estimate the foreground probability for each pixel with the above formula. In the case of multiple frames, when estimating the higher order terms $H_i(\mathbf{x},\theta)$ these two distributions are computed from pixels accumulated from all frames considered. Segmentations obtained with probabilities estimated from multiple frames are of higher quality, less prone to accidental box misalignments and other noises.

5 Experiments

We run experiments on the large-scale YouTube-Objects video dataset [25], which contains challenging sequences of ten object categories (aeroplane, bird, boat, car, cat, cow, dog, horse, motorbike, train) filmed *in the wild*. The dataset has 5484 video shots for a total of 571089 frames. The videos display significant clutter, with foreground objects coming in and out of focus and often out of sight, undergoing occlusions and significant changes in scale and viewpoint. We present final comparative results in Table 1. The numbers show

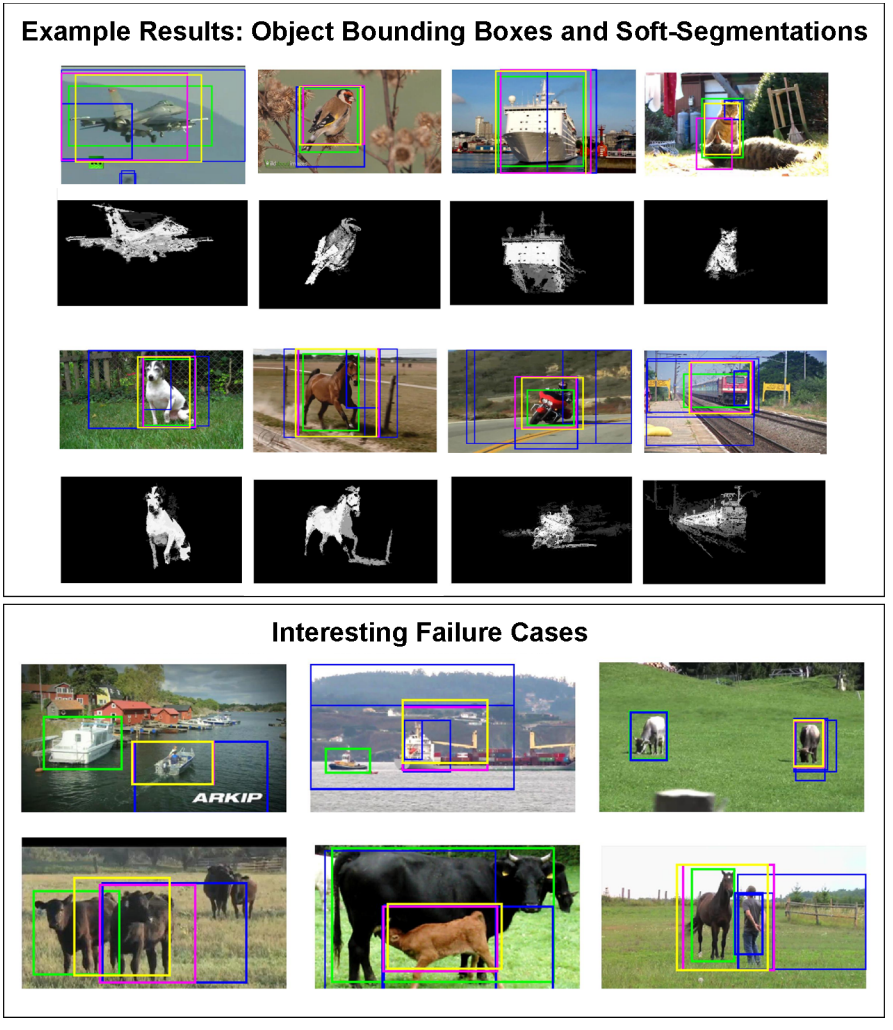

Figure 3: Top: example results using our method. In blue we show the candidate bounding boxes that survive the first filtering stage. In magenta we show the boxes matched, before location refinement. The final boxes are in yellow and the green boxes are the ground truth. We also present the final soft-segmentations, after the final boxes are produced. Bottom: some interesting failure cases, showing how ambiguous the problem of main object discovery could be. Also note the high quality of our soft foreground segmentations.

the percentage of correct bounding boxes found, per class, where a detection is considered successful if the agreement with the ground-truth box, measured as overlap over union, is greater than 0.5.

We also present bounding box accuracies after each stage of our approach, demonstrating how each phase improves the quality of detection (Table 2). The last two rows show, in percentages, how many times the center of mass of the soft-segmentation mask hits inside the ground truth bounding box (denoted as *hit ratio*). We evaluated the soft-segmentation

Table 2: Average results per class after each stage. Stage 1 selects a random candidate. Note how Stage 3 using Mean-Shift refinement, improves over Stage 2 by 3.6%. Last two rows: evaluation of soft-segmentation modules, presenting the frequency with which the mass center of the segmentation mask hits inside the ground truth object box.

Evaluation after different stages	aeroplane	bird	boat	car	cat	cow	dog	horse	motorbike	train	Average
After stage 1	17.2	39.7	22.5	32.4	35.7	28.6	22.8	20.3	37.1	20.8	27.7
After stage 2	37.2	60.2	50.5	49.3	60.2	48.6	36.7	40.6	38.7	41.7	46.4
After stage 3	38.3	62.5	51.1	54.9	64.3	52.9	44.3	43.8	41.9	45.8	49.9
Soft-segmentation evaluation	aeroplane	bird	boat	car	cat	cow	dog	horse	motorbike	train	Average
Final segmentation hit ratio	76.5	82.9	72.6	76.1	86.3	72.9	74.7	70.3	52.4	70.8	73.6
SegVideoPCA hit ratio	90.4	77.5	84.0	69.2	79.8	71.2	73.3	62.9	58.1	68.1	73.5

produced by VideoPCA by itself, as well as the final soft-segmentation, after Stage 3. Note that VideoPCA alone matches the accuracy of the final segmentation module. The high numbers indicate that the foreground masks are, in general, well centered on the object. As the ground truth mask is not available, a more accurate evaluation of these masks was not possible. See qualitative results in Figures 3 and 2.

Computation time: on a 2.2 GHz Laptop PC using unoptimized Matlab code, the average times per frame, per different modules, are: Fast DeepFlow: 1.3 sec; VideoPCA: $0.01-0.02$ sec; Bounding box proposals and filtering: 2 sec; Creating the potentials: 3 sec per shot; Matching with IPFP: 0.007 sec per shot; Stage 3: 1 sec; Total time of the whole method from beginning to end, per frame: 6.9 sec.

6 Conclusions

We have presented an efficient method for automatic discovery of foreground objects in video sequences with state-of-the-art performance on several classes from the large scale YouTube-Objects dataset. Different from most current methods, ours is able to efficiently discover object bounding boxes and their soft-segmentation masks by considering foreground/background separation cues along with appearance and geometric matching consistency over multiple frames in the sequence. Additionally, we propose an efficient procedure with realtime performance for locating foreground regions in video based on Principal Component Analysis, which helps significantly in producing high quality bounding boxes. Our approach, by proposing efficient bounding box generation, location refinement and object soft-segmentation, covers and extends current approaches in object discovery in video. For future work we plan to extend our method to the case of multiple objects discovery. We will also continue to develop our VideoPCA algorithm and evaluate it independently on several datasets, as a stand-alone saliency detection method.

Acknowledgements: Marius Leordeanu was supported by CNCS-UEFICSDI, under project PNII PCE-2012-4-0581.

References

[1] Olivier Barnich and Marc Van Droogenbroeck. Vibe: A universal background subtraction algorithm for video sequences. *Image Processing*, 20(6), 2011.

[2] A. Borji, D. Sihite, and L. Itti. Salient object detection: A benchmark. In *ECCV*, 2012.

[3] M. Cheng, N. Mitra, X. Huang, P. Torr, and S. Hu. Global contrast based salient region detection. *PAMI*, 37(3), 2015.

[4] Minsu Cho, Jungmin Lee, and Kyoung Mu Lee. Reweighted random walks for graph matching. In *ECCV*, 2010.

[5] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *Pattern Analysis and Machine Intelligence*, 24(5), 2002.

[6] T. Cour and J. Shi. Solving Markov Random Fields with spectral relaxation. In *International Conference on Artificial Intelligence and Statistics*, 2007.

[7] Rita Cucchiara, Costantino Grana, Massimo Piccardi, and Andrea Prati. Detecting moving objects, ghosts, and shadows in video streams. *Pattern Analysis and Machine Intelligence*, 25(10), 2003.

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

[9] T. Deselaers, B. Alexe, and V. Ferrari. Weakly supervised localization and learning with generic knowledge. *IJCV*, 100(3), 2012.

[10] X. Hou and L. Zhang. Saliency detection: A spectral residual approach. In *CVPR*, 2007.

[11] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li. Salient object detection: A discriminative regional feature integration approach. In *CVPR*, 2013.

[12] A. Joulin, F. Bach, and J. Ponce. Discriminative clustering for image co-segmentation. In *CVPR*, 2010.

[13] A. Joulin, F. Bach, and J. Ponce. Multi-class cosegmentation. In *CVPR*, 2012.

[14] Armand Joulin, Kevin Tang, and Li Fei-Fei. Efficient image and video co-localization with frank-wolfe algorithm. In *Computer Vision–ECCV 2014*, pages 253–268. Springer, 2014.

[15] G. Kim, E.P. Xing, Li Fei-Fei, and T. Kanade. Distributed cosegmentation via submodular optimization on anisotropic diffusion. In *ICCV*, 2011.

[16] D. Kuettel, M. Guillaumin, and V. Ferrari. Segmentation propagation in imagenet. In *ECCV*, 2012.

[17] M. Leordeanu and M. Hebert. Smoothing-based optimization. In *CVPR*, 2008.

[18] M. Leordeanu, R. Collins, and M. Hebert. Unsupervised learning of object features from video sequences. In *CVPR*, 2005.

[19] M. Leordeanu, M. Hebert, and Rahul Sukthankar. An integer projected fixed point method for graph matching and map inference. In *NIPS*, 2009.

[20] D. Liu and T. Chen. A topic-motion model for unsupervised video object discovery. In *CVPR*, 2007.

[21] J. Liu, J. Luo, and M. Shah. Recognizing realistic actions from videos "in the wild". In *CVPR*, 2009.

[22] M.H. Nguyen, L. Torresani, F. De la Torre, and C. Rother. Weakly supervised discriminative localization and classification: a joint learning process. In *CVPR*, 2009.

[23] Anestis Papazoglou and Vittorio Ferrari. Fast object segmentation in unconstrained video. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 1777–1784. IEEE, 2013.

[24] D. Parikh and T. Chen. Unsupervised identification of multiple objects of interest from multiple images: discover. In *Asian Conference on Computer Vision*, 2007.

[25] A. Prest, C. Leistner, J. Civera, C. Schmid, and V. Ferrari. Learning object class detectors from weakly annotated video. In *CVPR*, 2012.

[26] Mrigank Rochan and Yang Wang. Efficient object localization and segmentation in weakly labeled videos. In *Advances in Visual Computing*, pages 172–181. Springer, 2014.

[27] M. Rubinstein, A. Joulin, J. Kopf, and C. Liu. Unsupervised joint object discovery and segmentation in internet images. In *CVPR*, 2013.

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

[29] P. Siva, C. Russell, T. Xiang, and L. Agapito. Looking beyond the image: Unsupervised learning for object saliency and detection. In *CVPR*, 2013.

[30] J. Sivic, B. Russell, A. Efros, A. Zisserman, and W. Freeman. Discovering objects and their location in images. In *ICCV*, 2005.

[31] S. Vicente, C. Rother, and V. Kolmogorov. Object cosegmentation. In *CVPR*, 2011.

[32] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid. DeepFlow: Large displacement optical flow with deep matching. In *ICCV*, 2013.

[33] C. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, 2014.