TEMPORAL OBJECTNESS: MODEL-FREE LEARNING OF OBJECT PROPOSALS IN VIDEO

Liang Peng and Xiaojun Qi

Department of Computer Science, Utah State University, Logan, UT 84322-4205
liang.peng@aggiemail.usu.edu and xiaojun.qi@usu.edu

ABSTRACT

Intrinsic natures of different appearance between sub-regions of objects and non-objects in optical flows lead to more visual consistency for object proposals. Hence, visual variations in different sub-regions in video sequences over time is a good indicator for likeliness of objects. We propose a method that dynamically measures the objectness of each proposal by exploiting temporal consistency within each optical flow. We develop a block-based feature representation using object's spatial property and define an objectness measure using the temporal changes of this spatial representation. As a result, the proposed temporal objectness learns good object proposals over a short period (e.g., less than 1 second). The proposed method is model-free and can be used to simultaneously learn and track object proposals without training. Experiments on a video dataset shows that the proposed approach significantly outperforms state-of-the-art methods in terms of precision-recall.

Index Terms— Object Detection, Object Proposals, Video, Temporal Objectness, Model-Free Learning

1. INTRODUCTION

Object detection is a fundamental problem in computer vision. It refers to the task of locating and categorizing objects in images and videos. Earlier work of object detection typically used multiple object-dependent detectors to scan over the entire images by sliding windows [1] [2]. Each detector was responsible for finding a certain category of objects by applying binary classification on this type of objects and background. There were two major shortcomings for this approach. First, the sliding window-based search space was large; Second, the number of detectors needed to be ran depends on the number of categories of objects so the computational complexity grows linearly with the number of categories. To increase the speed, Haar-cascaded detectors [3] [4] were proposed to quickly eliminate a majority of non-object windows at the early stage. Later, segmentation-based [5] [6] and salience-based methods [7] [8] were proposed to reduce the search space and achieve better performance. In recent years, class-agnostic methods [9] [10] have been introduced to find out the regions likely containing generic objects. First, it narrowed down the search space to be much smaller and also made computation not grow regarding to the number of object categories. After generation of object proposals, classification methods were employed to classify all object categories and one background category. Recently, deep learning evolved with the ever-growing larger labeled data size and the faster computing power (e.g., gpu). Some breakthrough progress on the classification stage [11] has been made. Some deep learning methods [12] [13] used Convolution Neural Network (CNN) to train both object proposal generation and classification and fine-tune shared features. However, learning for object proposals using deep neural networks was still quite expensive and slow in training.

Based on what has been discussed above, the object proposal generation remains a bottleneck in object detection [14]. Developing fast and accurate object proposal learning methods has attracted a lot of attention. A new method named BING [15], which developed a fast way to compute gradient to generate object proposals, has been proposed and achieved a high detection rate (i.e., 0.90 with 50 proposals and 0.96 with 1000 proposals at Intersection over Union (IoU) of 0.5 on the VOC 07 data set [16]) . Another approach named EdgeBox [17] used edge as a sparse yet informative representation, together with the number of contours that are wholly contained in a bounding box, as an indication to measure the objectness. It has reached the state-of-the-art object detection rate (i.e., 0.7 at IoU of 0.7 using 1000 proposals on the VOC 07 data set).

Most of aforementioned work has been focusing on generating object proposals for images. Video composing a sequence of dense images (e.g., 25 frames per second) raises additional challenges for methods that fit for images, due to high complexity and lack of temporal consistency if detectors were applied on individual frames. Some recent studies used temporal information and image-based detectors on videos. Sharir and Tuytelaars [18] applied object proposals in each frame and linked them over frames into spatiotemporal object hypothesis. The detection on individual frames was still costly. Oneata et al. [19] proposed the supervoxel method that used hierarchical clustering of superpixels in a graph with spatial and temporal connections. Hua et al. [20] incorporated tracking and detection to improve the consistency and reduce the cost of detection on individual frames. Some motion-based methods [21] [22] for object detection in video could only address moving objects but not static objects in video.

In this paper, building upon the fast object detection method in images, we improve the object proposal generation by offering the following contributions. First, we develop a block-based feature representation to capture the object's spatial property. Second, we exploit temporal information as cues and use variations of spatial features in optical flows to formulate a temporal objectness measure. Third, we use the proposed objectness to simultaneously detect and track object proposals in video to generate object proposals tubes with high speed and improved accuracy. The rest of the paper is organized as follows. In Section 2, we introduce the proposed method in detail. Section 3 describes the experimental results. Section 4 provides conclusions and future work.

2. LEARNING OBJECT PROPOSALS USING TEMPORAL OBJECTNESS

Since applying object detection on individual frames of a video is undesirable, we seek to incorporate detection and tracking to effec-

Fig. 1. Sample images of positive bird patches tube (upper) and negative patches tube (lower), where patches tube is a set of patches across frames in the temporal order generated by tracker

tively detect objects in each frame of a video. For object detection in static images, some recent generic object detection algorithms (e.g., EdgeBox and BING) could quickly locate the patches that likely contain objects in images. These patches serve as candidates for true object locations. Since objects differ from non-objects (i.e., pure backgrounds or regions containing partial objects and partial backgrounds) by their structural closed boundary, the fast generic object detector exploits this property to detect objects. For example, EdgeBox exploits the edges of objects and uses the boundaries that are wholly enclosed in a bounding box to measure objectness. BING uses norm of gradients to measure objectness. However, these detectors tend to assign high objectness scores to patches that either contain true objects or partial objects and partial backgrounds, and assign low objectness scores to patches containing relatively pure background. Therefore, further distinguishing the true object patches and patches that contain partial objects and backgrounds could enhance the discriminative power of classifying objects and non-objects. Compared to images, videos contain additional temporal information. Since tracking methods aim to estimate optical flows over temporal frames by assuming that the objects move coherently, the consistency of patches in optical flows could be a good indicator for objectness. We did some preliminary studies and came up with the following hypothesis: the object patches tend to have more consistency than patches that include partial objects and partial backgrounds. In other words, object patches tend to have less variations than patches that include partial object and partial background in an optical flow. Fig. 1 shows an example of the comparison of negative bird patches tube and positive bird patches tube. It clearly demonstrates the less variations in the positive bird patches tube. Driven by this hypothesis, we propose a compact feature representation for spatial appearance of each patch, and formulate a temporal objectness to learn object proposals.

2.1. Block-based Spatial Feature Representation

All objects could be roughly divided into rigid objects and non-rigid objects. Examples of rigid objects include airplanes, cars, and boats. Example of non-rigid objects include human, birds, and dogs. A sub-region of an image is called a patch. In a video, a tracker generates a set of patches across frames in the temporal order. This set of patches is called a patches tube. Our preliminary studies show that patches tubes for objects in optical flow during tracking generally have much higher consistency and lower variations over time than patches tubes for non-objects. Non-rigid objects typically have consistency in optical flow as a whole part but some internal part/parts of the object might move and change appearance more than the rest parts depending on the structure of deformable objects. For example, the body of a running horse usually does not change appearance as much as the four legs.

To measure the relative consistency in appearance of both rigid and non-rigid objects, we develop the following block-based feature representation. First, we divide the patch of interest into a 3×3 grid, resulting in a total of 9 blocks. Within each of 9 blocks, we develop a compact and powerful feature vector to represent the local visual properties. Using the variation of pixel values in red, green, and blue channels, the entropy, the aspect ratio, and the spatial distribution of 3 color channels, we define the feature representation as a 20D feature vector. The 20 values in the feature vector include standard deviations of rgb color channels (3 values), the entropy of the luminance-based histograms (1 value), relative frequencies of 5 bins with top frequencies in rgb color channels (15 values), and the aspect ratio of the patch (1 value).

To compute the entropy, we first compute the histogram based on the pixel counts for each of 768 bins (i.e., 256 possible values times 3 channels), and then apply the standard entropy formula on relative frequencies on each of 768 bins. To compute the relative frequencies, firstly we compute the histogram based on the pixel counts of 64 bins on each of rgb channels. Secondly, for each channel, we normalize the frequency of each bin by dividing the total pixel counts in the channel. Lastly, we rank relative frequencies and select top 5 relative frequencies in each channel to produce a total of 15 values. The aspect ratio is the width of a patch divided by the height of it.

Each patch has 9 blocks with each block represented by a 20D feature vector. For non-rigid objects, the changes in some moving blocks are typically bigger than the changes in other blocks. Using this block-based spatial feature representation, we capture the relatively steady regions by measuring the changes of each patch in the optical flow represented by the patches tube.

2.2. Temporal Objectness

We describe how to measure temporal objectness using this block-based spatial feature representation and temporal information in detail. Our goal is to measure how the patch's visual appearance changes over consecutive frames over time by using the block-based spatial features. Since only parts of the object move in the non-rigid object patches tube, we would like to measure the overall variation of consecutive patches at relatively static regions of the object in a patches tube. Hence, we first compute 9 block-wise distances between neighbouring patches in the patches tube and only keep the least 5 distances. Next, we compute the average of these 5 distances as the neighbour change between two neighbouring patches. Finally, we compute the median of all neighbour changes across the patches tube as a measure of overall variation. Since more variation indicates less likeliness of objects, we subtract this median from zero as the temporal objectness. Algorithmically, let P_{t_i} represent a patch in frame number t_i, B_{x,y,t_i} represent the spatial feature vector of a block at the xth horizontal partition and yth vertical partition in patch P of frame t_i where x=1,2,3, y=1,2,3, t_i = 1,...,n, and n is the number of frames to be tracked. A tracked patches tube could be represented as $\{P_{t_i}\}$ and the corresponding block features could be represented as $\{B_{x,y,t_i}\}$. Fig. 2 provides visual illustration of the

9-block-based adjacent patches, which are used to compute temporal objectness o for $\{B_{x,y,t_i}\}$ as summarized in Algorithm 1.

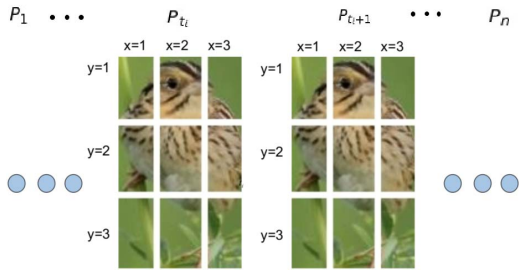

Fig. 2. Illustration of the 9-block-based adjacent patches

Algorithm 1 Compute the temporal objectness for a patches tube

input: $\{B_{x,y,t_i}\}$ (a sequence of 20-D feature vectors)
output: o (a value represents temporal objectness)
$D_T \leftarrow$ new List /*temporal distances*/
for $t_i \leftarrow 1$ to $n-1$ **do**
  $D_N \leftarrow$ new List /*neighbor patch distances*/
  for $x \leftarrow 1$ to 3 **do**
    for $y \leftarrow 1$ to 3 **do**
      $d \leftarrow$ EuclideanDist(B_{x,y,t_i}, B_{x,y,t_i+1})
      D_N.append(d)
    end for
  end for
  sort(D_N, order=ascending) /*sort 9 values in D_N*/
  $min5D \leftarrow$ first 5 elements of D_N
  $aveD \leftarrow$ average($min5D$)
  D_T.append($aveD$)
end for
$o \leftarrow$ -median(D_T) /*find 0 minus median of n-1 temporal distances in a patched tube*/

Temporal objectness o represents zero minus the central tendency measure of changes over time sequence for each patches tube. We use median on temporal distance pairs of adjacent frames to compute temporal objectness since it is more robust than mean in terms of handling outliers. Since the proposed approach uses a few temporal frames to compute objectness without need of training, it is a model-free method with the linear time complexity $O(n)$, where n represents the number of frames that we use to compute the temporal objectness.

After computing temporal objectness, we use it as a measure of likeness for an object and rank them for all object proposals tracks. Then we compute precision and recall using different thresholds ranging from top 1 to top 20 proposals.

3. EXPERIMENTAL RESULTS

To evaluate the proposed method, the experiments are conducted on the YouTube-Objects dataset V2.2 [23]. The dataset is composed of the videos that are collected from YouTube and from names of 10 object classes of the PASCAL VOC Challenge [16]. There are 9 to 24 videos for each object class. The duration of each video varies between 30 seconds and 3 minutes. The videos are weakly annotated, where each video contains at least one object of the corresponding class.

Rigid objects	Non-Rigid objects
aeroplane: 13 videos, 482 shots	bird: 16 videos, 175 shots
boat: 17 videos, 191 shots	cat: 21 videos, 245 shots
car: 9 videos, 212 shots	cow: 11 videos, 70 shots
motorbike: 14 videos, 444 shots	dog: 24 videos, 217 shots
train: 15 videos, 324 shots	horse: 15 videos, 151 shots

Table 1. The number of videos and shots for each class of objects.

The entire dataset contains a total of 720,000 frames with almost 7,000 bounding-box annotations. The annotated frames have been divided into the training set and the testing set. One instance has been annotated per frame in the training set and all instances have been annotated in the testing set. In total, there are 4306 annotated frames with 4306 bounding box annotations in the training set. There are 1781 annotated frames with 2669 bounding box annotations in the testing set. Each video contains multiple shots with their annotated starting and ending frame numbers. Table 1 lists the 10 classes of objects categorized as rigid and non-rigid objects together with the number of videos and shots for each class.

Since not all frames are annotated and all instances have been annotated in the testing set for annotated frames, we select up to 10 annotated frames from each shot in each video in the testing set as the initial frame for object detection. Specifically, if a shot has 10 or more annotated frames, we select the first 10 annotated frames. If a shot has fewer than 10 annotated frames, we select all annotated frames. Due to the sparse annotation of frames, the first 10 annotated frames have sufficient temporal gaps along the frame numbers and we do not have duplicated frames for initializing the tracker. After selecting up to 10 frames per shot for all shots, this data set contains frames from different shots and therefore is called a shot representative frame set. For each frame in this shot representative frame set, we apply EdgeBox detector [17], to generate object proposals and use EdgeBox objectness to select top 20 proposals whose locations are used as the initial object locations for tracking. The parameters of EdgeBox are set as follows: the step size of the sliding window is 0.65; the non-maximum suppression threshold is 0.75, the min score of boxes to be detected is 0.01, and the maximum number of boxes to be detected equals 1000. After selecting top 20 bounding boxes, we use them as initial positions and apply SPOT tracking [24] for each of boxes for a total of 20 consecutive frames (i.e., n=20) to generate optical flows in video. We chose 20 frames for tracking because empirical studies suggested that it is long enough to have discriminative temporal objectness. The duration of 20 frames is typically less than 1 second so most tracked patches tubes still cover the initial region without drifting, even in an unconstrained video, unless the video shot has big changes or the tracked objects move out of the scene. Since the SPOT tracker [24] uses constraints among multiple objects for initialization, we only use frames that have at least 2 detected positive objects to initialize tracker. For each object track covered in 20 frames, we compute temporal objectness using the proposed method described in section 2. Since our goal is to evaluate the proposed temporal objectness for object proposals and we used the annotated positions of objects in the initial frames to compute recall rate and precision of top n proposals, we need to make sure that each tracked patches tube correctly contains object without drifting (i.e., object become non-object or vice versa). To this end, we manually remove last few patches to make sure each patch in a tube is consistently tracked (i.e., containing objects from start to end).

In the end,we use 425 remaining frame sequences (i.e., 260

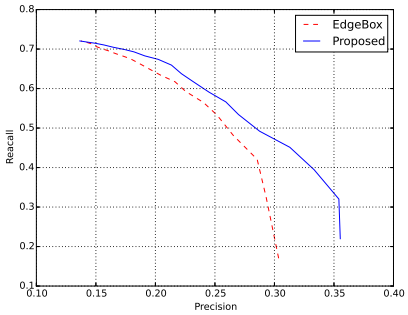

Fig. 3. Comparison of PR curves of objects generated from the proposed temporal objectness and the EdgeBox objectness for top N=1,...,20 (IoU=0.7)

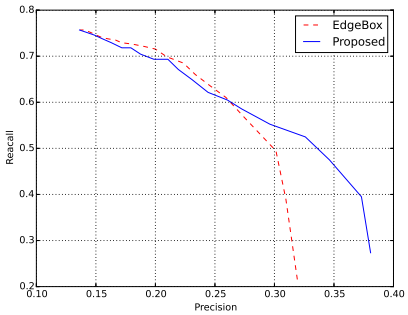

Fig. 4. Comparison of PR curves of rigid objects generated from the proposed temporal objectness and the EdgeBox objectness for top N=1,...,20 (IoU=0.7)

rigid and 165 non-rigid frame sequences) which contain 687 annotated objects (i.e., 362 rigid objects and 325 non-rigid objects) in our experiments. For comparison, we rank patches tubes based on EdgeBox objectness and select top 1 to top 20 patches tubes from each of 20 frames in the frame sequence to compute precision and recall at each level from 1 to 20. We also use the proposed temporal objectness to select from top 1 to top 20 patches tubes from each frame sequence to compute precision and recall. Fig. 3 shows the precision-recall(PR) curves of the proposed objectness versus EdgeBox objectness for all 10 categories of objects for IoU of 0.7, a common value chosen for PR comparison. It clearly shows that the proposed method significantly outperforms state-of-the-art method EdgeBox overall. The curves merge at top 20 patches tubes because we use all 20 tubes generated from EdgeBox. From this comparison, we can see that the proposed objectness learns better object proposals than EdgeBox within top 20 proposals by taking advantage of temporal information. Fig. 4 and Fig. 5 show the PR curves for rigid and non-rigid objects, respectively. For rigid objects, we can see that the proposed method outperforms EdgeBox when the precision is greater than 0.26. This corresponds to extracting approximately top 10 proposals per image. For non-rigid objects,we can see from Fig. 5 that the proposed method significantly outperforms EdgeBox. From the results split by rigid and non-rigid objects, we can see that the proposed method performs especially well on non-rigid objects due to computing temporal objectness using the minimum 5 block-wise distances on block-based features. Overall, the proposed method

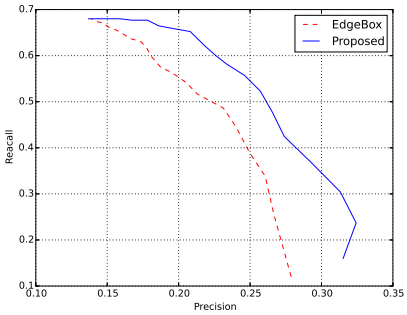

Fig. 5. Comparison of PR curves of non-rigid objects generated from the proposed temporal objectness and the EdgeBox objectness for top N=1,...,20 (IoU=0.7)

effectively assigns low scores for the proposals containing partial objects and backgrounds and therefore achieves significant improvement in generating accurate proposals for rigid and non-rigid objects compared with the EdgeBox method. For object detection in videos, which typically employs tracking, the proposed method can compute temporal objectness to improve the quality of the object proposals. In other words, incorporating the proposed temporal objectness in video can achieve higher precision (i.e., fewer false positives) with the same number of proposals or the same recall rate with the fewer number of proposals. This improvement is obtained with the model-free learning (i.e., no training is needed) and with little additional cost on detection. Hence, this approach could lead to further development of tracking and detection system with improved quality of object proposals. Fig. 6 shows a sample of qualitative results using the proposed object proposal learning method.

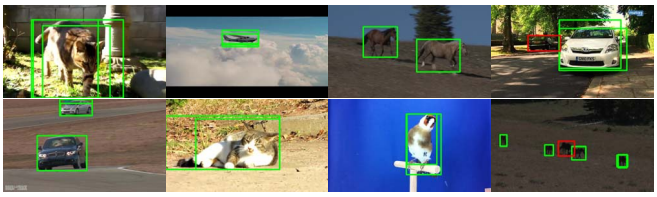

Fig. 6. Qualitative examples of the object proposals produced by the proposed method. Green boxes show matched objects (IoU=0.7), red ones show missed ground truth

4. CONCLUSIONS

We propose a temporal objectness measure that takes both spatial feature and temporal changes into consideration and achieves higher performance in generating higher quality general object proposals than state-of-the-art methods. The proposed is generally applicable to various object proposals and tracking methods. With tracking along the way, it learns better object proposal representation using temporal information. In future, we will incorporate tracker with re-detection to improve object proposals tubes along tracking in a long video sequence when tracker is less confident to develop a tracking and detection system. We will further reduce false positive object proposals by selecting top proposal tubes with higher recall and precision to improve accuracy of object recognition in videos.

5. REFERENCES

[1] Christoph H Lampert, Matthew B Blaschko, and Thomas Hofmann, "Beyond sliding windows: object localization by efficient subwindow search," in *Proceedings of IEEE Int. Conf. on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[2] Christian Wojek, Gyuri Dorkó, André Schulz, and Bernt Schiele, "Sliding-windows for rapid object class localization: a parallel technique," in *Pattern Recognition*, pp. 71–81. 2008.

[3] Paul Viola and Michael Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the IEEE Int. Conf. on Computer Vision and Pattern Recognition*, 2001, vol. 1, pp. I–511.

[4] Rainer Lienhart and Jochen Maydt, "An extended set of haar-like features for rapid object detection," in *Proceedings of IEEE Int. Conf. on Image Processing*, 2002, vol. 1, pp. I–900.

[5] Bastian Leibe, Aleš Leonardis, and Bernt Schiele, "Robust object detection with interleaved categorization and segmentation," *International Journal of Computer Vision*, vol. 77, no. 1-3, pp. 259–289, 2008.

[6] Jamie Shotton, Andrew Blake, and Roberto Cipolla, "Contour-based learning for object detection," in *Proceedings of IEEE Int. Conf. on Computer Vision*, 2005, vol. 1, pp. 503–510.

[7] Tie Liu, Zejian Yuan, Jian Sun, Jingdong Wang, Nanning Zheng, Xiaoou Tang, and Heung-Yeung Shum, "Learning to detect a salient object," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 2, pp. 353–367, 2011.

[8] Ali Borji, Dicky N Sihite, and Laurent Itti, "Salient object detection: a benchmark," in *Proceedings of European Conf. on Computer Vision*, 2012, pp. 414–429.

[9] Dumitru Erhan, Christian Szegedy, Alexander Toshev, and Dragomir Anguelov, "Scalable object detection using deep neural networks," in *Proceedings of IEEE Int. Conf. on Computer Vision and Pattern Recognition*, 2014, pp. 2155–2162.

[10] Christian Szegedy, Alexander Toshev, and Dumitru Erhan, "Deep neural networks for object detection," in *Proceedings of Advances in Neural Information Processing Systems*, 2013, pp. 2553–2561.

[11] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[12] Honglak Lee, Roger Grosse, Rajesh Ranganath, and Andrew Y Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," in *Proceedings of ACM Int. Conf. on Machine Learning*, 2009, pp. 609–616.

[13] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of IEEE Conf. on Computer Vision and Pattern Recognition*, 2014, pp. 580–587.

[14] Jan Hosang, Rodrigo Benenson, Piotr Dollár, and Bernt Schiele, "What makes for effective detection proposals?," in *arXiv preprint arXiv:1502.05082*. IEEE, 2015.

[15] Ming-Ming Cheng, Ziming Zhang, Wen-Yan Lin, and Philip Torr, "Bing: Binarized normed gradients for objectness estimation at 300fps," in *Proceedings of IEEE Int. Conf. on Computer Vision and Pattern Recognition*, 2014, pp. 3286–3293.

[16] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results," http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

[17] C Lawrence Zitnick and Piotr Dollár, "Edge boxes: Locating object proposals from edges," in *Proceedings of IEEE European Conf. on Computer Vision*, 2014, pp. 391–405.

[18] Gilad Sharir and Tinne Tuytelaars, "Video object proposals," in *IEEE Workshops on Computer Vision and Pattern Recognition*, 2012, pp. 9–14.

[19] Dan Oneata, Jerome Revaud, Jakob Verbeek, and Cordelia Schmid, "Spatio-temporal object detection proposals," in *Proceedings of IEEE European Conf. on Computer Vision*, Sept. 2014, pp. 737–752.

[20] Yang Hua, Karteek Alahari, and Cordelia Schmid, "Online object tracking with proposal selection," in *Proceedings of the IEEE Int. Conf. on Computer Vision*, 2015, pp. 3092–3100.

[21] Pakorn KaewTraKulPong and Richard Bowden, "An improved adaptive background mixture model for real-time tracking with shadow detection," in *Video-based Surveillance Systems*, 2002, pp. 135–144.

[22] D Hari Hara Santosh, Poornesh Venkatesh, LN Rao, and NA Kumar, "Tracking multiple moving objects using gaussian mixture model," *International Journal of Soft Computing and Engineering (IJSCE)*, vol. 3, no. 2, pp. 2231–2307, 2013.

[23] Alessandro Prest, Christian Leistner, Javier Civera, Cordelia Schmid, and Vittorio Ferrari, "Learning object class detectors from weakly annotated video," in *Proceedings of IEEE Int. Conf. on Computer Vision and Pattern Recognition*, 2012, pp. 3282–3289.

[24] Lu Zhang and Laurens van der Maaten, "Structure preserving object tracking," in *Proceedings of IEEE Int. Conf. on Computer Vision and Pattern*, 2013, pp. 1838–1845.