

FLORE
Repository istituzionale dell'Università degli Studi di Firenze

Spatio-Temporal Closed-Loop Object Detection

Questa è la Versione finale referata (Post print/Accepted manuscript) della seguente pubblicazione:

Original Citation:

Spatio-Temporal Closed-Loop Object Detection / Galteri, Leonardo; Seidenari, Lorenzo; Bertini, Marco; Del Bimbo, Alberto. - In: IEEE TRANSACTIONS ON IMAGE PROCESSING. - ISSN 1057-7149. - ELETTRONICO. - (2017), pp. 0-0. [10.1109/TIP.2017.2651367]

Availability:

This version is available at: 2158/1071440 since: 2019-04-11T19:37:52Z

Published version:

DOI: 10.1109/TIP.2017.2651367

Terms of use:

Open Access

La pubblicazione è resa disponibile sotto le norme e i termini della licenza di deposito, secondo quanto stabilito dalla Policy per l'accesso aperto dell'Università degli Studi di Firenze (https://www.sba.unifi.it/upload/policy-oa-2016-1.pdf)

Publisher copyright claim:

Conformità alle politiche dell'editore / Compliance to publisher's policies

Questa versione della pubblicazione è conforme a quanto richiesto dalle politiche dell'editore in materia di copyright.

This version of the publication conforms to the publisher's copyright policies.

(Article begins on next page)

19 March 2024

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2017.2651367, IEEE Transactions on Image Processing

Spatio-Temporal Closed-Loop Object Detection

Leonardo Galteri, Lorenzo Seidenari, Marco Bertini, and Alberto Del Bimbo, *Member, IEEE*

***Abstract*—Object detection is one of the most important tasks of computer vision. It is usually performed by evaluating a subset of the possible locations of an image that are more likely to contain the object of interest. Exhaustive approaches have now been superseded by object proposal methods. The interplay of detectors and proposal algorithms has not been fully analyzed and exploited up to now, although this is a very relevant problem for object detection in video sequences. We propose to connect, in a closed-loop, detectors and object proposal generator functions exploiting the ordered and continuous nature of video sequences. Different from tracking we only require a previous frame to improve both proposal and detection: no prediction based on local motion is performed, thus avoiding tracking errors. We obtain 3 to 4 points of improvement in mAP and a detection time that is lower than Faster R-CNN, which is the fastest CNN based generic object detector known at the moment.**

***Index Terms*—Object Detection, Video Analysis, Objectness**

I. Introduction

Object detection is one of the most important tasks of computer vision and as such has received considerable attention from the research community. Typically object detectors identify one or more bounding boxes in the image containing an object and associate a category label to it. These detectors are specific for each class of objects, and for certain domains exist a vast literature of specialized methods, such as face detection [9], [27], [42] and pedestrian detection [11], [17].

In recent years the objectness measure, that quantifies how likely an image window is containing an object of any class [2], has become popular [3], [8], [12], [32], [38]. The popularity of objectness proposal methods lies in the fact that they can be used as a pre-processing step for object detection to speed up specific object detectors.

The idea is to determine a subset of all possible windows in an image with a high probability of containing an object, and feed them to specific object detectors. Object proposals algorithms perform two main operations: generate a set of bounding boxes and assign an objectness score to each box.

The window proposal step is typically much faster than the exhaustive evaluation of the object detector. Considering that a "sliding window" detector has typically to evaluate 10^6 windows, if it is possible to reduce this number to 10^3–10^4, evaluating only these proposals, then the overall speed is greatly improved. In this sense objectness proposal methods can be related to cascade methods which perform a preliminary fast, although inaccurate, classification to discard the vast majority of unpromising proposals [21]. Reducing the search space of object bounding boxes has also the advantage of reducing the false positive rate of the object detector.

The great majority of methods for objectness proposal have dealt with images, while approaches to video objectness proposal are oriented toward segmentation in supervoxels [41], deriving objectness measures from the "tubes" of superpixels that form them [29], [40]. This process is often computationally expensive and requires to process the whole video.

In this paper we present a novel and computationally efficient spatio-temporal objectness estimation method, that takes advantage of the temporal coherence of videos. The proposed method exploits the sequential nature of videos to improve the quality of proposals based on the available information on previous frames determined by detector outputs. We define this approach as closed-loop proposals, since we exploit not only the current frame visual feature but also the proposals evaluated on a previous frame. Integrating the output of objectness proposals with object detection, we obtain a higher detection rate when computing spatio-temporal objectness in videos and we also improve the detection running time.

We point out that our approach is different from tracking and is not based on any form of it. Object tracking, especially in the multi-target setting, is usually addressed using object detectors and some data association strategy that can be either causal [5] and non-causal [28]. In the proposed approach we exploit the temporal coherence of sequences causally, but we do not estimate motion of objects, either implicitly or explicitly. Moreover, our end goal differs from the one of tracking, that is to precisely locate an object instance in order to keep its identity correct as long as possible. Our goal is to enhance the quality of object proposals so to improve both detection quality and speed.

II. Related works

The problem of quantifying how likely a part of an image is showing an object of some class is related to saliency detection. Works in this area typically aim at predicting salient points of human eye fixation [34] or modeling visual attention [4]. However, a detector may need to handle objects that are not visually conspicuous or that do not draw human gaze, thus an object proposal method should be able to deal also with objects that are not salient. Desirable properties for an object proposal method are:

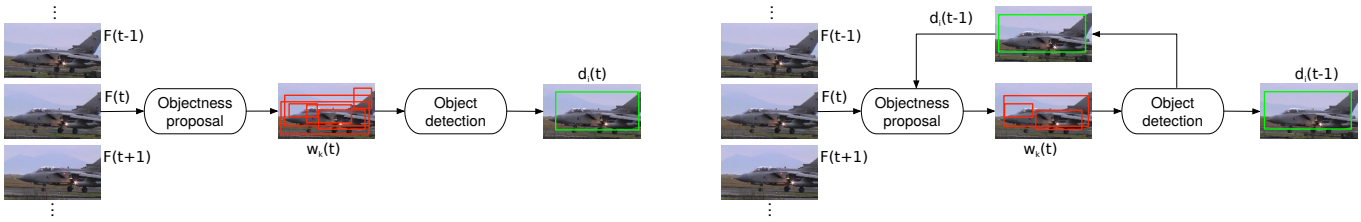
- **High object detection rate / proposal recall:** to avoid discarding good candidate windows that are not processed by a specific object detector at a later stage.
- **High computational efficiency / low processing time:** to allow using the method in real-time applications or to effectively use it as pre-processing step in an object

L. Galteri, L. Seidenari, M. Bertini, and Alberto Del Bimbo are with the Media Integration and Communication Center (MICC), Università degli Studi di Firenze, Viale Morgagni 65, 50134 - Firenze, Italy.

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the author. The material includes a video showing the behavior of our closed-loop approach. Contact leonardo.galteri@unifi.it for further questions about this work.

1057-7149 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2017.2651367, IEEE Transactions on Image Processing



(a) Open-Loop (b) Closed-Loop

Fig. 1: Schemes of: (a) typical objectness/detection pipeline; (b) our spatio-temporal objectness interaction. In our method window proposals are passed to the detector at time t and the detector output obtained at time $t-1$ is fed back to the proposal algorithm to improve window ranking. This approach reduces the number of proposals w.r.t. typical pipeline.

detection pipeline. This property is related to the number of candidate window proposals that are computed.

- **Good object generalization:** to detect a large number of different objects, so that proposals can be used with many different specific object detectors.
- **Good cross-dataset generalization:** to maintain an acceptable detection rate on a testing dataset that is different from that of training, without need of retraining.
- **High repeatability:** to consistently propose windows on similar image content, despite image perturbation or changes, thus allowing to exploit proposals for a better training of object detectors [21].

Hosang *et al.* [21] have very recently presented a comparison of twelve object proposal methods for images, applying them to Pascal VOC 2007, MS COCO and ImageNet 2013 datasets, comparing some of these properties.

A. Spatial Objectness

These methods propose a relatively small number of proposals (e.g. 10^3–10^4) that should cover all the objects of an image, independently from their class. Typically they rely on low-level segmentation such as the method proposed by Felzenszwalb and Huttenlocher [16], or use their own segmentation algorithm.

Gu *et al.* [20] have presented a framework for object detection and segmentation that groups hierarchically segments to detect candidate objects, evaluating performance using the bounding boxes that encompass these regions.

The method proposed by Alexe *et al.* [2], [3] uses different cues such as multi-scale saliency, color contrast, edge density, superpixels segments, location and size of the proposal window, combining them in a Bayesian framework.

Enders *et al.* [13] generate a set of segmentations by performing graph cuts based on a seed region and a learned affinity function. Regions are ranked using structured learning based on a mix of a large number of cues.

Uijlings *et al.* [38] propose a method that requires no parameter learning, combining exhaustive search and segmentation in a data-driven selective search. The approach is based on hierarchical grouping of regions, using color, texture and region features. The work of Manén *et al.* [26] is similar in spirit to that of [38], but randomizing the merging process and learning the weights of the merging function.

Instead of following a hierarchical approach, the method proposed by Carreira and Sminchisescu [6] generates sets of overlapping segments, obtained solving a binary segmentation problem, initialized with different seeds. Segments are ranked by objectness using a trained regressor.

Differently from the methods reported above, the two methods proposed by Zitnick and Dollár [43], and Cheng *et al.* [8] do not use image segmentation.

The method of [43], called Edge Boxes, computes a scoring function in a sliding window fashion. Scoring is performed measuring the number of edges that exist in the box minus those that are members of contours that overlap the box's boundary.

The method of [8] is the fastest approach, as reported in the comparison of [22], and uses a simple linear classifier over edge features, that is trained and applied in a sliding window manner. The efficiency of the approach is due to the use of approximated features, binarized normed gradients that give the name (BING) of the method.

State of the art object detection is nowadays achieved by region based convolutional neural network methods [14], [18], [19], [33]. R-CNN pioneered this task by simply applying a pre-trained network to regions. Improved accuracy in detection is then achieved fine-tuning the network on object boxes and learning a bounding box regressor.

More recent approaches [18], [33] have applied a similar idea but avoiding a full computation of the convolutional feature for each region, sharing instead a single image feature map for all the evaluated boxes.

Recently Ren *et al.* [33] presented Faster R-CNN, an integrated approach of proposal and detection computation. Faster R-CNN adds a Region Proposal Network (RPN) to Fast R-CNN thus exploiting the same convolutional feature computation pipeline to compute proposals. This approach is efficient in terms of computation time since it avoids the burden of proposal generation from an external module, by sharing the features among RPN and Fast R-CNN detection.

Following this setting a few objectness methods have been built on top of convolutional features. Multibox [14] approaches exploit a saliency based approach and after classifying an image they propose a few boxes per class on salient regions.

Different from the fully integrated approach of [33], DeepBox and DeepMask [30] learn to generate windows, or even

1057-7149 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2017.2651367, IEEE Transactions on Image Processing

masks with a deep convolutional architecture. These methods have a higher recall with respect to EdgeBoxes although they are more than an order of magnitude slower.

B. Spatio-Temporal Objectness

Objectness proposal in videos is typically cast as a problem of supervoxel segmentation, although supervoxel evaluation measures - such as those used in [41] - are reported as not being directly indicative of the performance of such methods when applied to spatio-temporal objectness proposal [29]. Van den Bergh *et al.* [40] have addressed the problem by tracking windows aligned with supervoxels, obtained from frame superpixel segmentation [39], over multiple frames using an online optimization; the proposed method runs at 30fps on a single modern CPU. Oneata *et al.* [29] follow a similar approach, in principle, by segmenting individual frames into a superpixel graph, then computing supervoxels through temporal hierarchical clustering. Spatio-temporal object detection proposals are based on supervoxel segmentation, obtained using a version of the region growing method of Manén *et al.* [26] extended to the temporal domain.

Spatio-temporal objectness measures have been used to perform co-localization, i.e. spatial localization of common objects in a set of videos. Prest *et al.* [31] have proposed a fully automatic pipeline to learn object detectors from object proposals; segments of coherent motion are extracted from video shots, and spatio-temporal bounding boxes are fit to each segment, forming video "tubes" that are then used to train detectors, following a selection process based on objectness probability. The approach proposed by Joulin *et al.* [23] extends the method of image co-localization of [36] to videos, extending it with temporal terms and constraints, and solves efficiently the resulting quadratic problem applying the Frank-Wolfe algorithm. Unlike [31], the method does not use video tubes. Kwak *et al.* [25] address video object detection as a combination of two processes, i.e. object discovery and tracking, that complement each other. During discovery, regions containing similar objects are matched across different videos, while tracking associates prominent regions within each video. Motion statistics of individual regions and temporal consistency between consecutive regions are used to improve tracking and obtain the video tubes for object detection.

C. Video Object Detection

Recently, convolutional neural networks have been applied to the problem of video object detection. Tripathi *et al.* [37] have proposed a video object proposal method based on spatio-temporal edge contents, and a deep-learning based method for video object detection applied to clusters of these proposals. Class labels are propagated through streaming clusters of spatio-temporal consistent proposals, speeding up detection by $3\times$ with respect to per-frame detection. Kang *et al.* [24] have proposed a framework for video object detection based on CNNs that detect and track proposals. In a first stage video tubelets are proposed, combining object detection, to provide high-confidence anchors to the tracker, and tracking, to generate new proposals and to aggregate detections. In a second stage tubelets are classified and re-scored through spatial max-pooling and temporal convolution, for robust box-scoring and for incorporating temporal consistency.

III. The proposed method

The method is based on the intuition that since objectness proposals are used as a pre-processing step followed by object detection, it is possible to exploit the joint statistics of window proposals and detections to compute spatio-temporal objectness in a video sequence, improving both detection rate and speed. Detection accuracy is improved by eliminating possible false detections, while processing speed is improved by selecting a reduced number of areas to be tested by the detector.

Typically window proposal methods require 10^3 windows to cover more than 90% of the objects shown in an image. In case objects are very small the number of proposals may become 10^4. Considering video frame sequences, it is natural to use the detection of an object to improve the next proposal, since objects will likely be in about the same position in the next frame. Based on this consideration, we propose a feedback model accounting for spatio-temporal consistency of detections and window proposals over time, that re-ranks object proposals based on the overlap with detections and detector scores obtained in the previous frame. Using the outcome of a detector on a frame reduces the number, and improves the quality, of the proposals in a later frame. On the other hand those proposals are used to speed and improve the quality of detection in the following frame. In contrast to classical object detection pipelines, shown in Fig. 1a, our approach exploits previous frame detections to improve proposals. As shown in Fig. 1b, providing the detection as a feedback will allow to select a reduced number of higher quality proposals.

Given a video sequence with T frames, consider a set of object proposals

$$\mathcal{W} := \{w_1(1), \ldots, w_P(1), \ldots, w_1(T), \ldots, w_P(T)\} \quad (1)$$

for the ease of notation we assume the proposal method computes a fixed amount of proposals P for each frame, but this is not a fixed requirement.

Considering the task of detecting objects from multiple classes, a set of models $\mathcal{M}$ will be trained to output a vector of $|\mathcal{M}|$ scores for every window. A detector $C(F, w, \mathcal{M}) : \mathcal{F} \times \mathbb{R}^4 \rightarrow \mathbb{R}^{|\mathcal{M}|}$ is a function evaluating a proposal for a frame F according to some set of models $\mathcal{M}$ and image features $\mathcal{F}$. Given a proposal $w_i(t)$ the detector C will associate it to a score vector $\mathbf{s}_i(t) \in \mathbb{R}^{|\mathcal{M}|}$.

Let $\mathcal{D}_t$ be the set of scored proposals at time t defined by the tuples $d_i(t) := \langle w_i(t), \mathbf{s}_i(t) \rangle$. The final set of detections $\overline{\mathcal{D}}_t$ is obtained preserving tuples d_i such that

$$||\text{sign}(\mathbf{s}_i(t) - \tau_{\mathcal{M}})||_1 > 0 \quad (2)$$

and performing non maximal suppression [18], where $\tau_{\mathcal{M}}$ is a model specific threshold vector on the soft-max per class

1057-7149 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2017.2651367, IEEE Transactions on Image Processing

output. To obtain detection windows useful for proposal re-ranking, we want to retain only the ones that have been assigned to at least one object class. This condition is ensured by the strict positivity of the L_1-norm of the signs of thresholded classifier outputs vector as expressed by Eq. 2.

An object proposal method can be seen as a function, $P(w, F) : \mathcal{F} \times \mathbb{R}^4 \rightarrow \mathbb{R}$ evaluating the probability that a given window w in a frame F contains an object, independently from the object category, namely $p(\text{object}|w)$.

For a given frame at time t, our goal is to induce an ordering on set $\mathcal{W}_t$ of proposals, exploiting information of previously evaluated ones $d(t-1) \in \mathcal{D}_{t-1}$, thus defining the ordered set $\widehat{\mathcal{P}}_t := \{\widehat{w}_1(t), \ldots, \widehat{w}_P(t)\}$ such that

$$\text{p}(\text{object}|\widehat{w}_i(t)) > \text{p}(\text{object}|\widehat{w}_{i-1}(t)) \quad (3)$$

$$\text{p}(\text{object}|\widehat{w}_i(t)) > \text{p}(\text{object}|w_i(t)), i < \theta \quad (4)$$

The new ranking should keep the *objectness* property, defined by Eq. 3, meaning that highly ranked windows are more likely to contain an object than lowly ranked ones. According to Eq. 4, our re-ranked set $\widehat{\mathcal{P}}_t$ should have a better ranking than $\mathcal{W}_t$, meaning that, in the first θ windows, the probability of finding an object for the i-th window of our re-ranked set $\widehat{\mathcal{P}}_t$ is higher than for the same-rank window in $\mathcal{W}_t$.

We can define the likelihood of finding a generic object on the whole frame at time t as

$$\mathcal{L}_o = \sum_{i=1}^{|\mathcal{W}_t|} \text{p}(\text{object}|w_i) \quad (5)$$

and similarly

$$\widehat{\mathcal{L}}_o = \sum_{i=1}^{|\widehat{\mathcal{P}}_t|} \text{p}(\text{object}|\widehat{w}_i) \quad (6)$$

Considering that $\widehat{\mathcal{P}}_t$ is a re-ordered version of $\mathcal{W}_t$ and that $|\mathcal{W}_t| = |\widehat{\mathcal{P}}_t|$, it is true that $\widehat{\mathcal{L}}_o = \mathcal{L}_o$. However, if Equation 3 and Equation 4 hold, a more interesting result is obtained considering only a subset of the proposals; with the improved ranking we have that, for a $K < \theta$, in a truncated sum $\mathcal{L}_o^K = \sum_{i=1}^{K} \text{p}(\text{object}|w_i)$:

$$\widehat{\mathcal{L}}_o^K > \mathcal{L}_o^K. \quad (7)$$

This means that we can evaluate a set of lower cardinality K instead of the full proposal set without compromising the chance of finding the objects we are seeking with our classifier. Evaluating less proposals also means reducing the chance of finding false detections. This is an important benefit of our model that is useful to reduce the computational complexity and also to improve the accuracy of classifiers.

Since object detectors are trained to output a maximal score when the evaluated windows have high overlap with ground truth object windows, we can exploit detector scores as proxies of the probability of finding an object in the area occupied by an evaluated window w_i.

Therefore to obtain the new set of proposals $\widehat{P}$ we link the detector and the proposal functions in a causal manner. Consider a set of N detections $d_i(t-1) \in \overline{\mathcal{D}}_{t-1}$, obtained from a frame at time $t-1$, and a set of proposals in frame at time t, it is possible to compute a spatio-temporal objectness at time t using for proposal window $w_k(t)$:

$$\begin{aligned}\widehat{o}_k(t) =& o_k(t) + \\ & \alpha \sum_{m=1}^{|\mathcal{M}|} \sum_{i=1}^{N} \text{IoU}\left(w_k(t), d_i(t-1)\right) \cdot s_{im}(t-1)\end{aligned} \quad (8)$$

where $o_k(t)$ represents the objectness score and

$$\text{IoU}\left(w, d\right) = \frac{area\left(w \cap d\right)}{area\left(w \cup d\right)} \quad (9)$$

is the overlap measure of the windows computed according to the PASCAL overlap criterion [15]. Term s_i is obtained via soft-max normalization therefore is comparable across classes without further calibration.

The IoU term makes sure that s_i can increase the objectness score of a proposal only if the detection window and the proposal window are overlapping, weighting the increase in objectness score by the overlap.

Finally, α is a parameter that weights the two parts of the function, and its optimal value is dependent on the dataset and the performance of the proposal algorithm that is used. In the following experiments we tuned this parameter by cross-validation, maximizing detection rate with 1000 proposals (DET@1000) for each dataset and object detector used.

The function of Eq. 8, is composed by two parts:

- **Objectness measure**. The objectness score computed using a spatial objectness measure obtained from an object proposal algorithm such as BING or EdgeBoxes.
- **Feedback Term**. This term combines two terms via multiplication: *i)* the overlap measure $\text{IoU}(\cdot, \cdot)$ accounting for the fact that proposal windows that have larger overlap with detection windows are more likely to contain the objects detected in the next frame, and the higher the overlap the higher the probability of this; *ii)* the detection score s_{im} accounting for the fact that not all detection windows really contain objects, and this is more likely for windows with a low detector confidence score. Thus detection windows with higher detector confidence are to be weighted more, to rank higher the objectness windows that contain objects.

Using the spatio-temporal objectness measure of Eq. 8 allows to greatly reduce the number of object proposal windows.

The main differences of the proposed method with respect to previous approaches can be summarized as follows. Differently from the [25], [31] video object proposal methods, and from the video object detection methods of [24], [37], the proposed method does not perform any tracking although it is possible, in principle, to track the $\widehat{P}$ proposal windows to obtain video tubes. However, experimental results show that even without this additional processing it is possible to outperform the methods of [24], [25], [31] on two standard datasets. Differently from [23], [25], [31] the proposed method is supervised, as [24]. Differently from [37], that extends EdgeBoxes from image object proposals to videos exploiting

1057-7149 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2017.2651367, IEEE Transactions on Image Processing

temporal edge responses, the proposed method is based on image objectness measures, and the temporal aspect is included in Eq. 8. This allows to choose different proposal methods, e.g. depending on the needed speed or performance.

IV. Experimental Evaluation

In the following experiments we evaluate the performance of the proposed method on videos, comparing it with three fast state-of-the art methods – BING[1], Edge Boxes[2] and Region Proposal Networks used by Faster R-CNN[3] – in terms of detection rate and speed. The method has been tested on the YouTube Objects dataset (YTO) [31], commonly used to test video object detection and proposal methods, and on the ILSVRC 2015 VID dataset [1], commonly used to test video object detection.

The YouTube Objects dataset (YTO) [31] contains 10 classes and consists between 9 and 24 videos for each class; to eliminate issues due to video compression artifacts 570,000 decompressed frames are provided. We report the results, in terms of localization metric (CorLoc) [10] that is typically used for evaluation on YTO; this experimental setup allows to compare the proposed method with the approaches of Prest *et al.* [31], Joulin *et al.* [23], Kwak *et al.* [25] and Kang *et al.* [24].

The ILSVRC 2015 VID dataset release used is the initial one, containing 30 object classes and consisting of 3 splits: a training set of 1952 fully-labeled video snippets with a length between 6 to 5213 frames per snippet; a validation set of 281 fully-labeled video snippets with a length between 11 to 2898 frames per snippet; a test set of 458 snippets whose ground truth annotation is not publicly available. We report the results, in terms of mean average precision (mAP), on the validation set; this experimental setup allows to compare the proposed method with the approach of Kang *et al.* [24].

The ILSVRC 2015 DET dataset comprises the fully annotated synsets from 200 basic level categories selected to provide various challenges such as object scale, level of image clutterness and average number of object instances.

We used Fast R-CNN as object detector using the implementation from [33]. For the YouTube Objects dataset our model has been trained using the Faster R-CNN framework starting from the pre-trained network named VGG_CNN_M_1024 [7], fine-tuning both the classifier and the region proposal net on PASCAL VOC 2007, since the YouTube Objects dataset object classes are a subset of the PASCAL VOC 2007 dataset.

For the ILSVRC 2015 VID dataset we trained the model using the pre-trained network named VGG_16 [35] as a starting point, fine-tuning both the classifier and the region proposal on the whole ILSVRC 2015 DET training set and some additional images from the training set of the ILSVRC 2015 VID dataset, choosing the ratio of $4:1$ between DET and VID sets.

[1]We used the code publicly available at http://mmcheng.net/bing/

[2]We used the code publicly available at http://research.microsoft.com/en-us/downloads/389109f6-b4e8-404c-84bf-239f7cbf4e3d/

[3]We used the code publicly available at https://github.com/rbgirshick/py-faster-rcnn

Faster R-CNN learns a Region Proposal Network (RPN) and an object detector which is architecturally equivalent to Fast R-CNN. Therefore the object detector weights are transferable to Fast R-CNN on which Faster R-CNN is based on. Indeed we used the same object detector weights in both frameworks. We refer to the detector as Faster R-CNN when we used Fast R-CNN and RPN as proposal sharing the weights, as referred by Ren *et al.* [33], and we refer to Fast R-CNN when proposals are computed externally.

A. Spatio-temporal objectness performance

In this set of experiments we evaluate the performance of the proposed spatio-temporal objectness method in terms of proposal correct localization.

The analysis of the behavior of our re-ranking process is shown in Fig. 2. We report the score of the detector on boxes of each rank, averaged over all frames and classes – we do not consider the scoring of detectors of classes different from the one present in the ground truth. This experiment shows that our boxes have a higher average detector score, meaning they are more precisely located on the object; moreover it can be seen how the highly scored boxes are all concentrated in the first 30-50 proposal while for the baseline methods they are more spread along the tail of the curve. A first qualitative glance at how our closed-loop spatio-temporal proposal improves over static baselines can be given in Fig. 3. It is clear, in this subset of frames, that our method increases the accuracy and quality of proposals generated by all baselines.

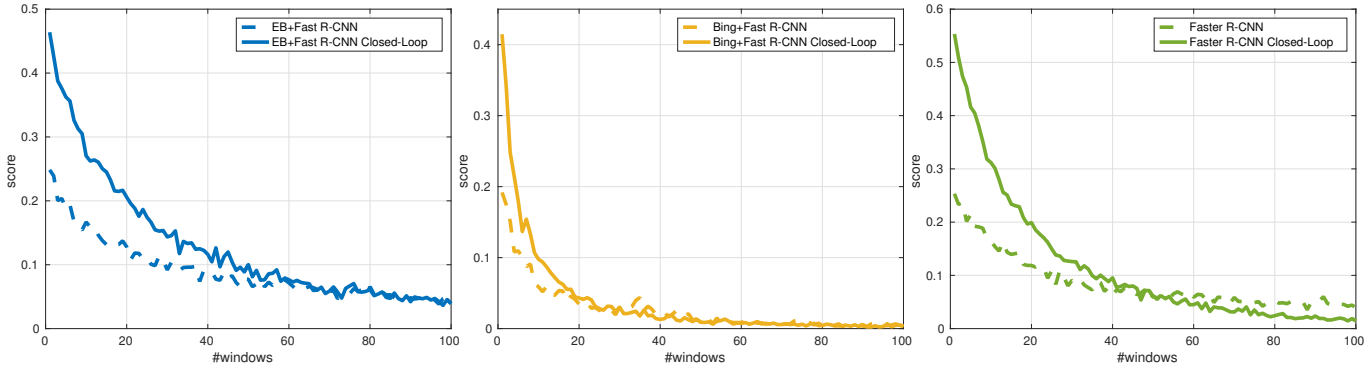
In Fig. 4 we evaluate the performance of proposals alone in terms of CorLoc on YTO. In this experiment we do not test if objects are correctly classified but only if proposal bounding boxes overlap with objects of any class. We compare all open-loop baselines and our closed-loop proposals with the method proposed by Oneata *et al.* [29]. The method of [29] has a performance close to BING, when using very few windows, but as the number of window proposals increases this is reverted. Our closed-loop proposal ranking obtains very high recall even with few tens of windows compared with open-loop baselines. Note that even if proposal recall is predictive of detector accuracy [21] evaluating detectors on proposals is necessary to assess the final detection result. This analysis is reported in the following Sect. IV-B.

Moreover, it has to be noted that the method of [29] is dominated by the LDOF optical flow computation and roughly requires 15 seconds to process each frame, instead of the 0.16 required by EdgeBoxes, 0.017 required by BING and 0.006 by RPN. Note that RPN timing is reported on a high-end GPU (NVIDIA Titan X) while BING, EdgeBoxes and the timing from [29] are reported using a single-core implementation on a 3.6 GHz CPU.

In Tab. I we compare with previously published methods [23], [24], [25], [31]. Our method is above the state-of-the art reported by Kang *et al.* [24]. Note that our method and the one by Kang *et al.* both use an algorithm trained with object class supervision, while [23], [25], [31] are unsupervised.

1057-7149 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2017.2651367, IEEE Transactions on Image Processing



(a) EdgeBoxes (b) BING (c) RPN

Fig. 2: Average box detector score varying box rank on Youtube Objects. Proposals obtained with our method have higher scores in average and highly scored proposal have higher rank with respect to the baselines.

Method											Avg.
Prest *et al.* [31]	51.7	17.5	34.4	34.7	22.3	17.9	13.5	26.7	41.2	25.0	28.5
Joulin *et al.* [23]	25.1	31.2	27.8	38.5	41.2	28.4	33.9	35.6	23.1	25.0	31.0
Kwak *et al.* [25]	56.5	66.4	58.0	76.8	39.9	69.3	50.4	56.3	53.0	31.0	55.7
Kang *et al.* [24]	**94.1**	69.7	**88.2**	79.3	76.6	18.6	**89.6**	**89.0**	87.3	75.3	76.8
RPN Closed-Loop	70.7	76.0	70.2	93.2	76.5	88.6	87.4	84.4	81.4	67.9	79.6
RPN	48.5	56.3	55.7	61.2	68.7	69.6	62.2	80.5	34.0	53.6	59.0
EdgeBoxes Closed-Loop	87.8	**94.8**	81.7	**95.1**	**84.3**	**97.5**	78.0	61.0	**94.8**	**76.8**	**85.2**
EdgeBoxes	71.9	72.9	75.6	86.4	52.2	91.1	79.5	62.3	74.2	71.4	73.8
BING Closed-Loop	71.1	87.5	54.2	90.3	80.0	92.4	89.0	85.7	79.4	69.6	79.9
BING	35.2	55.2	42.0	55.3	67.8	54.4	46.5	64.9	25.8	50.0	49.7

TABLE I: Localization performances on the YTO dataset. We run all proposal methods with 10 windows per frame in the baseline and Closed-Loop (CL) version.

B. Detection performance on video

In the following set of experiments we evaluate the closed-loop object detector on videos. We perform several comparisons to assess the behavior of our technique using three state-of-the art proposals EdgeBoxes, BING and RPN. We focus on the former since it runs in under 200ms per frame and it obtains state-of-the art results in terms of recall and detection accuracy [21]. We also evaluate the quality of our approach using BING which is less performant in terms of recall and detection accuracy but has a much lower run-time; indeed BING proposals can be computed in less than 20ms on modern CPUs. Finally we test our strategy with Faster R-CNN, the fastest and most performing detector tested on still images [33].

First we assess the effect of the number of evaluated proposals on detection accuracy. In Figure 5 it is clear that even with a very low number of proposals, as low as 30 per frame, we can obtain a mAP figure that is similar or better than the open-loop baselines using one order more of proposals.

The best performing proposal method on YTO is EdgeBoxes. Faster R-CNN is the second best. BING performs the worst but is surprisingly close to Faster R-CNN. Note that our closed-loop detection improves all three open-loop baselines.

As it can be seen from Fig. 5 our method improves the detection accuracy on both datasets, reducing false positives and selecting a set of higher quality proposals for the detector down stream. In this experiment we show how reducing the set of windows to a very compact set, 30 windows per frame, we are able to perform as well or better than with the full set of non re-ranked windows with the further benefit of speeding up the computation.

Considering the curves in Fig. 2 the RPN proposal appears to be the best although in term of detection is outperformed by EdgeBoxes. This happens because EdgeBoxes provides a better recall covering a higher percentage of objects in frames as is measured in Fig. 4. Being EdgeBoxes dataset agnostic it is likely that RPN is suffering from overfitting with respect to PASCAL VOC 2007, on which it is trained. We believe that this behavior depends on the fact that the model used on YTO has not been tuned on video frames. Instead, on ILSVRC we trained the detectors using frames from the DET and the VID training subsets. We believe that this improved performance is due to the additional tuning of the CNN on this larger set of data which also comprises video frames.

In Tab. II we report a comparison on YTO of our closed-loop detector using 50 proposals computed from BING, EdgeBoxes and using Faster R-CNN with the respective baselines.

Our method obtains from 3 to 4 points increase in term of mean average precision. We improve on all classes except

1057-7149 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2017.2651367, IEEE Transactions on Image Processing

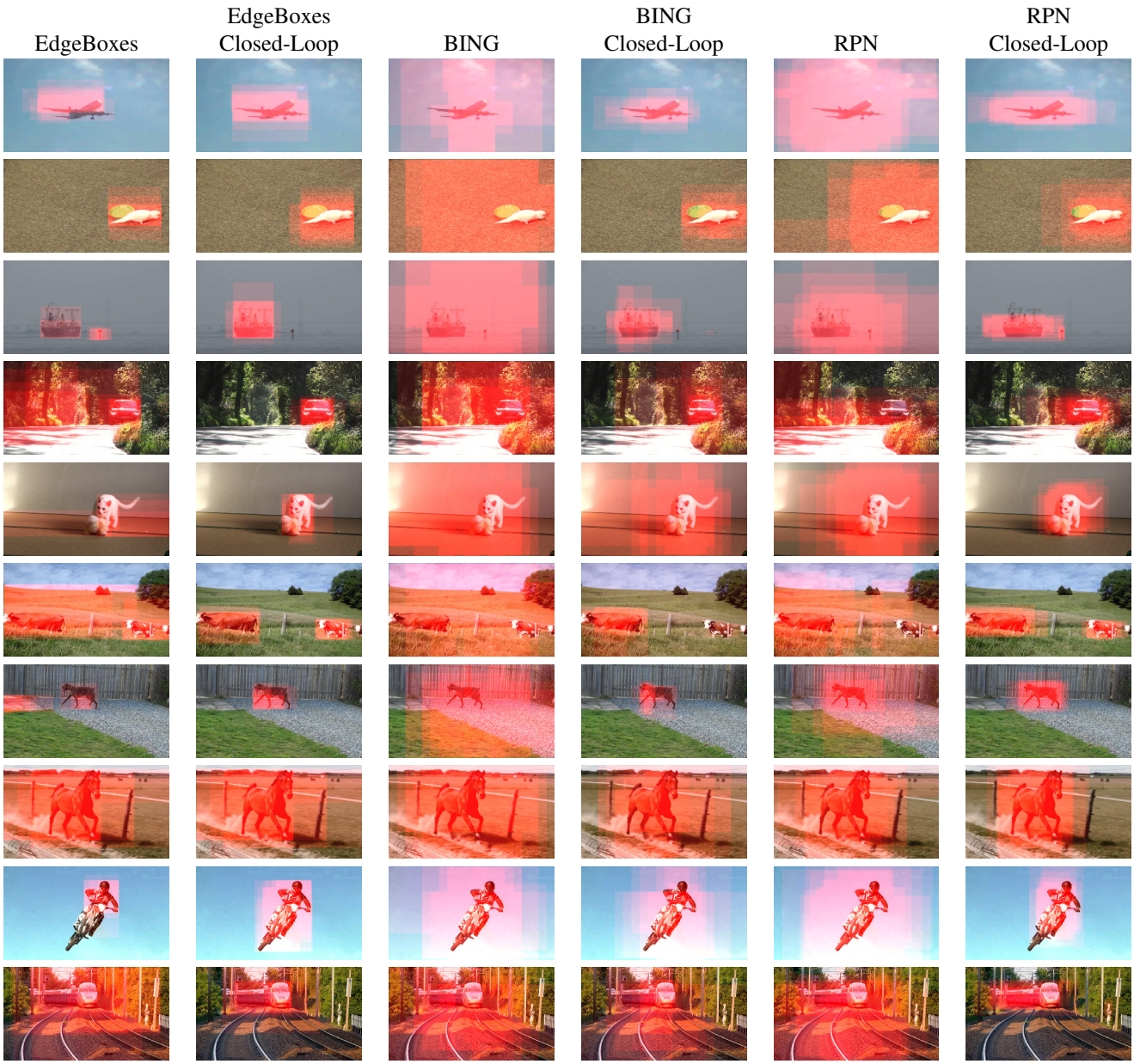


Fig. 3: Sample frames from the 10 classes from YouTube Objects dataset with the 10 highest ranked boxes. Baselines are presented in odd columns and our improved closed-loop proposal on even columns. Each box is represented as an overlapping transparent red box on the image. Our closed-loop proposal are more concentrated and accurate with respect to baseline methods.

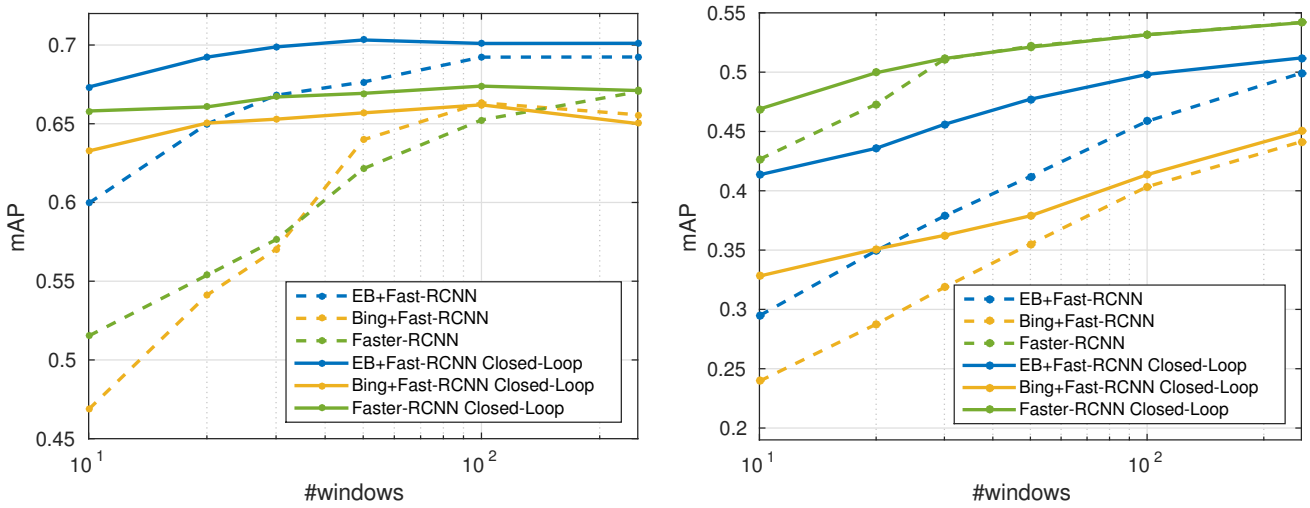
for "boat", that is the hardest class to detect. In this case the detection feedback has not enough quality to obtain a good re-ranking of proposals, therefore the exhaustive proposal evaluation may perform better.

Our method is able to increase the detection performance by reducing the amount of false positives per frame since it process a set of proposal with a high likelihood of containing an object.

Tab. III reports results of our method applied to RPN, EB and BING baselines on the validation set from ILSVRC 2015 VID using just 20 windows per frame. It can be observed that our closed-loop approach improves for most of the 30 classes. The only severe issues are on the "monkey" and "squirrel" classes. These classes are the most challenging and the detection quality is not adequate to provide any benefit in the loop. Interestingly we can boost the mAP on "squirrel" from 3.3 to 29.6 for RPN. Another challenging class is "lion", on this class our method obtains a high improvement for EB and BING, while on RPN we have a similar result. Out of 30 classes, closed-loop improves RPN on 20, EB on 25 and BING on 28. Finally, our Faster R-CNN model (RPN) using closed-loop improves over Kang *et al.* [24] using just 20 window proposals per frame. In our preliminary experiments, training only on frames from DET reported a lower mAP, e.g. 41.0 for

1057-7149 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2017.2651367, IEEE Transactions on Image Processing



(a) YouTube Objects (b) ILSVRC 2015 VID

Fig. 5: Detection accuracy with different proposals techniques and detectors on YouTube Objects and ILSVRC VID. Our Closed-Loop proposal improves mean average precision with respect to all baseline proposals. The gain is larger for a little amount of windows (10-50)

Proposal											mAP
RPN Closed-Loop	**68.3**	**72.7**	44.3	**88.8**	**58.3**	60.2	**71.5**	**69.1**	**77.3**	**58.6**	**66.9**
RPN	58.6	63.6	**47.2**	85.3	53.4	**60.8**	67.1	65.5	67.5	52.3	62.1
EB Closed-Loop	**72.7**	**81.3**	58.6	**90.5**	**64.8**	**63.0**	65.3	**62.5**	**79.7**	**66.0**	**70.4**
EB	71.3	75.2	**59.2**	86.2	54.1	62.5	**65.9**	62.7	78.8	60.7	67.6
BING Closed-Loop	**62.2**	**79.7**	50.0	**84.3**	53.3	56.9	**69.5**	**66.2**	**76.4**	**62.5**	**66.1**
BING	56.6	74.9	**51.3**	82.6	53.3	**61.0**	66.7	65.2	68.4	59.9	64.0

TABLE II: Comparison of open-loop and closed-loop proposals on YouTube Objects dataset using Fast R-CNN as a detector with 50 boxes. Using less or more boxes per frame resulted with worst or equal performance for all proposals in open- and closed-loop setting.

RPN closed-loop. We believe that the distribution of visual features in video, mostly because of blur and compression artifacts differs from the one in still images, and adding a small amount, i.e. a $4:1$ ratio, of VID frames to the training set helps fine-tuning the CNN and the proposal network, and leads to an improvement of almost 10 mAP points.

Our algorithm is based mainly on the re-ranking process expressed in Eq. 8, where the only free parameter is α. We show how the value of α influences detection performance for different proposal algorithms and amount of evaluated windows in Figure 6. The alpha parameter appears to correlate negatively with the amount of windows evaluated. Our understanding of this behavior is that since the set of feedback windows $\overline{\mathcal{D}}_{t-1}$ is the signal from which we obtain our information, if this signal is weak the feedback term must compensate this lack of information. Finally the behavior of α depends on the distribution of objectness scores o_k which can differ quite significantly between the analyzed methods.

In real-time applications such as automotive or visual surveillance it is likely not possible to analyse a stream at 30 frames per second, therefore a certain frame drop will occur causing the video to be processed at a lower frame rate. We are interested in analysing the performance of our approach in this more realistic setting. To assess the behavior of a closed-loop proposal we test it dropping frames, meaning that instead of using the frame before the one to be analysed as a source for detection windows $d_i(t-1)$, we use $d_i(t-n), n \in [2, 15]$.

In Fig. 7 we show how much detection accuracy of our method degrades if the source of detection windows is farther in time with respect to the current frame. It can be seen that our closed-loop method always performs better than its open-loop baseline.

C. Execution speed

In Tab. IV we report timing and mAP for our proposed closed-loop object detection method compared with the open-loop baselines. Our closed-loop method is able to produce a significant speed-up without loosing detection accuracy; for EdgeBoxes we even obtain a better mAP with our closed-loop proposal with respect to the open-loop baseline.

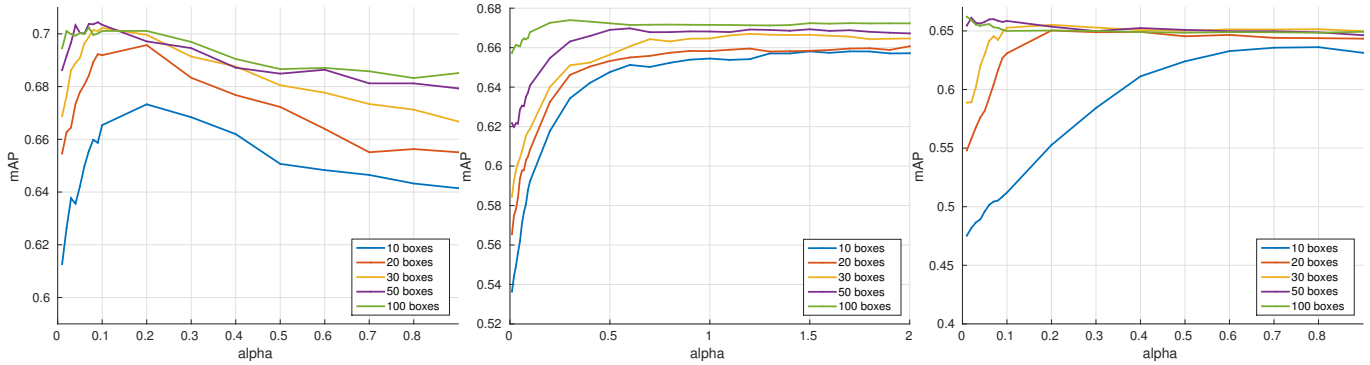
The gain in speed is higher for faster proposals since the full set of proposal has always to be computed before re-ranking and we can only reduce the amount of windows to be evaluated by the object detectors later in the pipeline.

1057-7149 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2017.2651367, IEEE Transactions on Image Processing

Method																mean AP
Kang [24]	72.7	**75.5**	42.2	**39.5**	25.0	**64.1**	**36.3**	51.1	24.4	48.6	**65.6**	73.9	**61.7**	**82.4**	30.8	**34.4**
RPN Closed-Loop	**74.8**	59.3	44.8	35.9	**37.0**	56.7	31.9	**54.3**	26.2	**74.1**	58.1	91.8	53.3	63.5	57.1	23.5
RPN	61.8	55.4	42.8	26.9	35.4	56.5	23.8	52.2	**26.6**	71.9	46.9	**92.3**	51.0	76.4	**57.3**	24.8
EB Closed-Loop	44.3	56.4	**50.6**	17.3	25.1	61.8	16.4	45.9	26.0	72.7	53.0	36.2	60.9	76.1	55.4	16.3
EB	54.2	38.1	22.5	14.3	20.8	46.2	13.0	54.2	21.0	63.4	51.1	58.0	39.7	33.7	19.5	0.2
BING Closed-Loop	29.1	35.9	37.4	23.2	22.5	46.1	15.6	35.1	16.3	54.6	58.2	44.7	50.4	72.1	49.5	9.6
BING	16.2	36.2	29.3	18.5	16.5	42.0	11.2	31.9	9.5	45.7	57.0	30.6	46.2	62.9	22.6	3.3
																mean AP
Kang [24]	54.2	**1.6**	61.0	**36.6**	19.7	55.0	**38.9**	2.6	**42.8**	54.6	**66.1**	69.2	26.5	**68.6**	47.5	
RPN Closed-Loop	**68.7**	0.0	**66.7**	15.2	19.1	73.1	34.9	**29.2**	34.1	85.1	59.4	72.1	**36.6**	62.0	**50.0**	
RPN	68.2	0.0	61.0	14.5	20.6	64.3	37.6	3.3	34.0	**86.6**	59.8	73.1	35.9	57.9	47.3	
EB Closed-Loop	67.4	0.0	55.2	20.9	35.9	65.0	27.8	0.1	33.0	84.3	63.3	**81.4**	16.4	42.4	43.6	
EB	30.7	0.0	59.0	5.4	**40.8**	**74.9**	25.5	0.0	18.4	74.5	60.2	73.7	5.5	30.3	35.0	
BING Closed-Loop	60.4	0.0	52.7	8.6	29.0	49.9	3.6	0.3	28.0	68.2	41.4	62.7	12.8	34.7	35.1	
BING	56.4	0.0	48.9	3.1	26.0	47.8	2.6	1.7	15.0	66.6	28.4	56.0	5.7	24.9	28.8	

TABLE III: Comparison of our method with Kang *et al.* [24] on ILSVRC VID dataset using 20 boxes per frame. Closed-Loop improves the map of RPN on 20, EB on 25 and BING on 28 out of 30 classes. Moreover our approach using RPN improves over the current state-of-the art.



(a) EdgeBoxes (b) BING (c) Faster R-CNN

Fig. 6: Effect of parameter α on detection accuracy for EdgeBoxes and BING varying the amount of proposals.

Proposal	Detector	Time/frame	Speed-up	mAP	GPU
RPN Closed-Loop	Faster R-CNN	56 ms	34%	66.9	yes
RPN	Faster R-CNN	75 ms		67.0	yes
EB Closed-Loop	Fast R-CNN	206 ms	21%	70.3	no
EB	Fast R-CNN	250 ms		69.2	no
BING Closed-Loop	Fast R-CNN	63 ms	70%	65.6	no
BING	Fast R-CNN	107 ms		65.6	no

TABLE IV: Timing of our Closed-Loop proposals combined with Fast R-CNN detector, also compared with region proposal networks (RPN) and Faster R-CNN detector. The GPU flag indicates whether the proposal set is generated using GPU. Detection is always performed on GPU.

V. Conclusion

In this paper we presented a novel closed-loop proposal strategy to be used on video sequences for object detection. Existing object proposal methods do not exploit the temporal ordering of frames. To the best of our knowledge we are the first to analyse and exploit the interplay between object detection and proposals. We show that our closed-loop strategy to generate proposals can improve speed and accuracy at the same time.

Our model is general and can be applied to any object detection pipeline on videos, which is based on window evaluation. We reported results using three state of the art object proposals in conjunction with Faster R-CNN, which is the fastest and most accurate object detector available. We measured a consistent improvement in proposal correct localization, detection accuracy and overall speed. The main limitation of our approach is constituted by the performance of the object detectors. If the open-loop detection quality is poor, the feedback can not provide any benefit.

Finally our method exploits the information of detectors in a causal manner and is robust to frame drop, thus providing ground for real-time applications.

References

[1] Imagenet object detection from video (VID) task dataset. http://image-net.org/challenges/LSVRC/2016/#vid, 2016.

[2] B. Alexe, T. Deselaers, and V. Ferrari. What is an object? In *Proc. of IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 73–80, 2010.

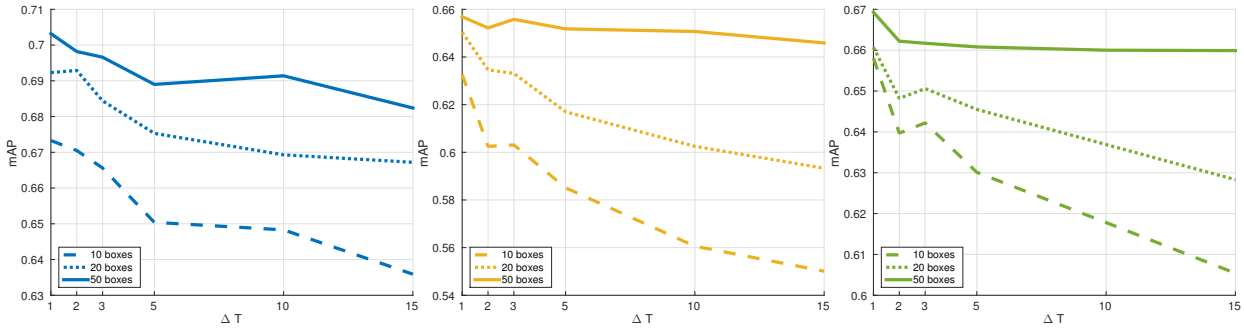
[3] B. Alexe, T. Deselaers, and V. Ferrari. Measuring the objectness of image windows. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 34(11):2189–2202, 2012.

[4] A. Borji and L. Itti. State-of-the-art in visual attention modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 35(1):185–207, Jan 2013.

[5] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool. Online multiperson tracking-by-detection from a single, uncalibrated camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 33(9):1820–1833, 2011.

1057-7149 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2017.2651367, IEEE Transactions on Image Processing



(a) EdgeBoxes (b) BING (c) Faster R-CNN

Fig. 7: Mean Average Precision of Fast R-CNN with our spatio-temporal proposal varying the framerate. Full, dotted and dashed lines are referred to results obtained using the most relevant 50, 20 and 10 proposal respectively.

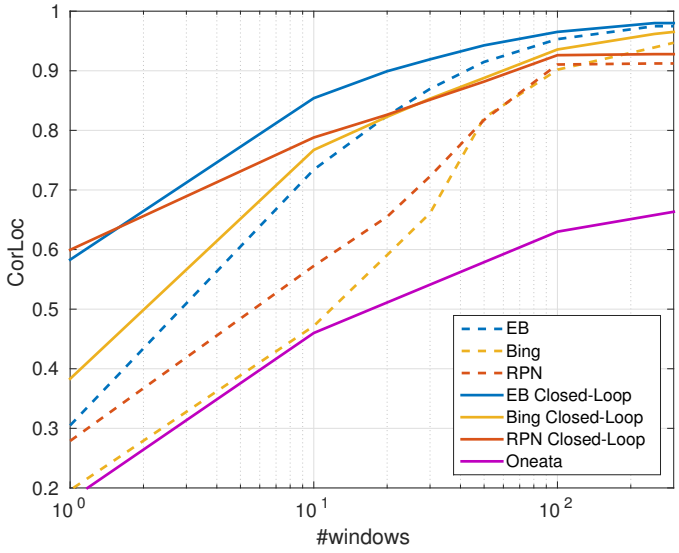


Fig. 4: Trade-off between detection rate and number of window proposals for the YouTube Objects dataset. Comparison between the proposed method with temporal information using Fast R-CNN object detector, the proposed method without temporal information and the method of Oneata *et al.* [29]. The proposed spatio-temporal objectness measure greatly improves the performance w.r.t. image based objectness.

[6] J. Carreira and C. Sminchisescu. Constrained parametric min-cuts for automatic object segmentation. In *Proc. of IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 3241–3248, 2010.
[7] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *Proc. of the British Machine Vision Conference (BMVC)*, 2014.
[8] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. Torr. BING: Binarized normed gradients for objectness estimation at 300fps. In *Proc. of IEEE Computer Vision and Pattern Recognition (CVPR)*, 2014.
[9] N. Degtyarev and O. Seredin. Comparative testing of face detection algorithms. In *Proc. of International Conference on Image and Signal Processing (ICISP)*, pages 200–209, 2010.
[10] T. Deselaers, B. Alexe, and V. Ferrari. Localizing objects while learning their appearance. In *Proc. of European Conference on Computer Vision (ECCV)*, 2010.
[11] P. Dollar, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: An evaluation of the state of the art. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 34(4):743–761, 2012.
[12] I. Endres and D. Hoiem. Category independent object proposals. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 575–588, 2010.
[13] I. Endres and D. Hoiem. Category-independent object proposals with diverse ranking. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 36(2):222–234, Feb 2014.
[14] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov. Scalable object detection using deep neural networks. In *Proc. of IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 2155–2162. IEEE, 2014.
[15] M. Everingham, L. Van Gool, C. 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, 2008.
[16] P. Felzenszwalb and D. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision (IJCV)*, 59(2):167–181, 2004.
[17] D. Geronimo, A. M. Lopez, A. D. Sappa, and T. Graf. Survey of pedestrian detection for advanced driver assistance systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 32(7):1239–1258, 2010.
[18] R. Girshick. Fast R-CNN. *arXiv preprint arXiv:1504.08083*, 2015.
[19] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proc. of IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 580–587. IEEE, 2014.
[20] C. Gu, J. J. Lim, P. Arbeláez, and J. Malik. Recognition using regions. In *Proc. of IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 1030–1037, 2009.
[21] J. Hosang, R. Benenson, P. Dollár, and B. Schiele. What makes for effective detection proposals? *arXiv preprint arXiv:1502.05082*, 2015.
[22] J. Hosang, R. Benenson, and B. Schiele. How good are detection proposals, really? In *Proc. of the British Machine Vision Conference (BMVC)*, 2014.
[23] A. Joulin, K. Tang, and L. Fei-Fei. Efficient image and video co-localization with Frank-Wolfe algorithm. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 253–268. Springer, 2014.
[24] K. Kang, W. Ouyang, H. Li, and X. Wang. Object detection from video tubelets with convolutional neural networks. In *Proc. of IEEE Computer Vision and Pattern Recognition (CVPR)*, 2016.
[25] S. Kwak, M. Cho, I. Laptev, J. Ponce, and C. Schmid. Unsupervised object discovery and tracking in video collections. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 3173–3181, 2015.
[26] S. Manen, M. Guillaumin, and L. V. Gool. Prime object proposals with randomized Prim's algorithm. In *Proc. of IEEE International Conference on Computer Vision (ICCV)*, pages 2536–2543, 2013.
[27] M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool. Face detection without bells and whistles. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 720–735, 2014.
[28] A. Milan, S. Roth, and K. Schindler. Continuous energy minimization for multitarget tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 36(1):58–72, 2014.
[29] D. Oneata, J. Revaud, J. Verbeek, and C. Schmid. Spatio-temporal object detection proposals. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 737–752. Springer, 2014.
[30] P. O. Pinheiro, R. Collobert, and P. Dollar. Learning to segment object candidates. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1981–1989, 2015.
[31] A. Prest, C. Leistner, J. Civera, C. Schmid, and V. Ferrari. Learning object class detectors from weakly annotated video. In *Proc. of IEEE*

1057-7149 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2017.2651367, IEEE Transactions on Image Processing

Computer Vision and Pattern Recognition (CVPR), pages 3282–3289, June 2012.
[32] E. Rahtu, J. Kannala, and M. Blaschko. Learning a category independent object detection cascade. In *Proc. of IEEE International Conference on Computer Vision (ICCV)*, pages 1052–1059, 2011.
[33] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 91–99, 2015.
[34] B. Schauerte and R. Stiefelhagen. Quaternion-based spectral saliency detection for eye fixation prediction. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 116–129, 2012.
[35] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proc. of International Conference on Learning Representations (ICLR)*, 2015.
[36] K. Tang, A. Joulin, L. Li, and L. Fei-Fei. Co-localization in real-world images. In *Proc. of IEEE Computer Vision and Pattern Recognition (CVPR)*, 2014.
[37] S. Tripathi, S. Belongie, Y. Hwang, and T. Nguyen. Detecting temporally consistent objects in videos through object class label propagation. In *Proc. of IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2016.
[38] J. Uijlings, K. van de Sande, T. Gevers, and A. Smeulders. Selective search for object recognition. *International Journal of Computer Vision (IJCV)*, 104(2):154–171, 2013.
[39] M. Van den Bergh, X. Boix, G. Roig, B. de Capitani, and L. Van Gool. SEEDS: Superpixels extracted via energy-driven sampling. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 13–26, 2012.
[40] M. Van den Bergh, G. Roig, X. Boix, S. Manen, and L. Van Gool. Online video SEEDS for temporal window objectness. In *Proc. of IEEE International Conference on Computer Vision (ICCV)*, pages 377–384, 2013.
[41] C. Xu and J. J. Corso. Evaluation of super-voxel methods for early video processing. In *Proc. of IEEE Computer Vision and Pattern Recognition (CVPR)*, pages 1202–1209, 2012.
[42] C. Zhang and Z. Zhang. A survey of recent advances in face detection. Technical report, Tech. rep., Microsoft Research, 2010.
[43] C. L. Zitnick and P. Dollár. Edge boxes: Locating object proposals from edges. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 391–405, 2014.



Leonardo Galteri received a master degree magna cum laude in computer engineering from the University of Florence in 2014 with a thesis on semantic video compression and object detection. Currently he is a PhD student and research fellow at the Media Integration and Communication Center of the University of Florence. His research interest focus on objecteness estimation, visual saliency and video compression.



Lorenzo Seidenari is a Postdoctoral researcher at the Media Integration and Communication Center of the University of Florence. He received his Ph.D. degree in computer engineering in 2012 from the University of Florence. His research focuses on object and action recognition in video and images. On this topics he addressed RGB-D activity recognition, embedding learning for multimodal-fusion, anomaly detection in video and people behavior profiling. He was a visiting scholar at the University of Michigan in 2013. He organized and gave a tutorial at ICPR 2012 on image categorization. He is author of 8 journal papers and more than 20 peer-reviewed conference papers.



Marco Bertini received the Laurea degree in Electronic Engineering from the University of Florence in 1999, and Ph.D. in 2004. He is working at the Media Integration and Communication Center of the University of Florence and is Associate Professor at the School of Engineering of the University of Florence. His interests are focused on digital libraries, multimedia databases and social media. On these subjects he has addressed semantic analysis, content indexing and annotation, semantic retrieval and transcoding. He is author of 22 journal papers and more than 100 peer-reviewed conference papers, with h-index: 24 (according to Google Scholar). He is associate editor of IEEE Transactions on Multimedia.



Alberto Del Bimbo is a Full Professor of Computer Engineering, and the Director of the Media Integration and Communication Center with the University of Florence. His scientific interests are multimedia information retrieval, pattern recognition, image and video analysis, and natural humancomputer interaction. From 1996 to 2000, he was the President of the IAPR Italian Chapter and the Member-at-Large of the IEEE Publication Board from 1998 to 2000. He was the General Co-Chair of ACM Multimedia 2010 and the European Conference on Computer Vision in 2012. He was nominated as ACM Distinguished Scientist in 2016. He received the SIGMM Technical Achievement Award for Outstanding Technical Contributions to Multimedia Computing, Communications and Applications. He is an IAPR Fellow, and an Associate Editor of Multimedia Tools and Applications, Pattern Analysis and Applications, the Journal of Visual Languages and Computing, and the International Journal of Image and Video Processing, and was an Associate Editor of Pattern Recognition, the IEEE Transactions on Multimedia, and the IEEE Transactions on Pattern Analysis and Machine Intelligence. He serves as the Editor-in-Chief of the ACM Transactions on Multimedia Computing, Communications, and Applications.

1057-7149 (c) 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.