Adobe Boxes: Locating Object Proposals Using Object Adobes

Zhiwen Fang, Zhiguo Cao, Yang Xiao, Lei Zhu, and Junsong Yuan, *Senior Member, IEEE*

***Abstract*— Despite the previous efforts of object proposals, the detection rates of the existing approaches are still not satisfactory enough. To address this, we propose Adobe Boxes to efficiently locate the potential objects with fewer proposals, in terms of searching the object adobes that are the salient object parts easy to be perceived. Because of the visual difference between the object and its surroundings, an object adobe obtained from the local region has a high probability to be a part of an object, which is capable of depicting the locative information of the proto-object. Our approach comprises of three main procedures. First, the coarse object proposals are acquired by employing randomly sampled windows. Then, based on local-contrast analysis, the object adobes are identified within the enlarged bounding boxes that correspond to the coarse proposals. The final object proposals are obtained by converging the bounding boxes to tightly surround the object adobes. Meanwhile, our object adobes can also refine the detection rate of most state-of-the-art methods as a refinement approach. The extensive experiments on four challenging datasets (PASCAL VOC2007, VOC2010, VOC2012, and ILSVRC2014) demonstrate that the detection rate of our approach generally outperforms the state-of-the-art methods, especially with relatively small number of proposals. The average time consumed on one image is about 48 ms, which nearly meets the real-time requirement.**

***Index Terms*— Object proposal, Adobe Boxes, object adobes, adobe compactness, objectness.**

Manuscript received October 26, 2015; revised April 5, 2016; accepted May 26, 2016. Date of publication June 9, 2016; date of current version July 14, 2016. This work was supported in part by the National Natural Science Foundation of China under Grant 61502187 and Grant 61502358, in part by the National High-Tech R&D Program of China (863 Program) under Grant 2015AA015904, in part by the Singapore Ministry of Education Academic Research Fund Tier 2 MOE2015-T2-2-114, in part by the Fundamental Research Funds for the Central Universities under Grant HUST: 2015QN036, and in part by the Youth Foundation of Hunan University of Humanities, Science and Technology under Grant 2015QN03. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Guoliang Fan. *(Corresponding author: Yang Xiao.)*

Z. Fang is with the National Key Laboratory of Science and Technology on Multi-Spectral Information Processing, School of Automation, Huazhong University of Science and Technology, Wuhan 430074, China, and also with the School of Energy and Mechanical-Electronic Engineering, Hunan University of Humanities, Science and Technology, Loudi 417000, China (e-mail: fzw310@hust.edu.cn).

Z. Cao and Y. Xiao are with the National Key Laboratory of Science and Technology on Multi-Spectral Information Processing, School of Automation, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: zgcao@hust.edu.cn; Yang_Xiao@hust.edu.cn).

L. Zhu is with the School of Information Science and Engineering, Wuhan University of Science and Technology, Wuhan 430081, China (e-mail: zhulei@wust.edu.cn).

J. Yuan is with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 (e-mail: jsyuan@ntu.edu.sg).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the author. The material includes a demo video for Adobe Boxes. The total size of the file is 13.4 MB. Contact Yang_Xiao@hust.edu.cn for further questions about this work.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2016.2579311

I. Introduction

DURING the past decades, a lot of efforts have been spent towards the efficient and robust object detection [1]–[3] and tracking-by-detection [4]–[6]. To accurately locate the target object in the cluttered scene, most of the previous object detection methods rely on the sliding window searching scheme. However, usually millions of subwindows need to be evaluated, thus resulting in the high computational cost. To overcome this issue, the object proposals [3] have shown the promising results to help efficiently localize the potential object regions first, via objectness estimation [7]–[12]. Thanks to the object proposal procedure, the object detection task can be narrowed down to a much smaller searching space of subwindows (e.g., thousands of subwindows), while still ensuring the detection performance. To locate the object proposals, some existing works focus on designing or learning the good object descriptive visual features [7], [11], [12]. Although they are often computationally efficient, the obtained features are not descriptive enough to characterize the generic objects well. This leads to the high false positive rate in the top scored proposals. On the other hand, using the rich information of superpixels, several other methods [10], [13] pay more attention to proposing the superpixel merging mechanisms for object proposal generation. Despite the high time consumption paid, they do not always achieve high detection rate, especially within the top scored proposals. The reason seems that, these methods do not define the effective objectness measures to evaluate the merged regions for ranking. While, objectness measure indeed plays a vital role to ensure the availability of the yielded object proposals [8]. Obviously, efficiently generating a small number of proposals with the high detection rate is indeed preferred by the practical applications [7], [12]. However, the aforementioned two kinds of object proposal approaches cannot fully satisfy this demand.

In this paper, we propose Adobe Boxes–a novel object proposal method–able to achieve high detection rate (DR) with a few number of proposals. Meanwhile, it can also improve DR of other approaches. Our main research idea is shown in Fig. 1. First, according to [15], the bridge between looking at an image and seeing an object is the regions of interest (ROI). Without any prior knowledge, ROI may contain a whole object or parts of it [16]. Thus, we define a new element, object adobes, for discovering the potential objects. In Fig. 1, different from searching all the superpixels of the train [10], our method pays attention to the salient "parts", i.e., superpixels of objects using the contrast analysis in the local region. We call these salient "parts" as the object adobes. Next, inspired by the visual organization rule [17], which

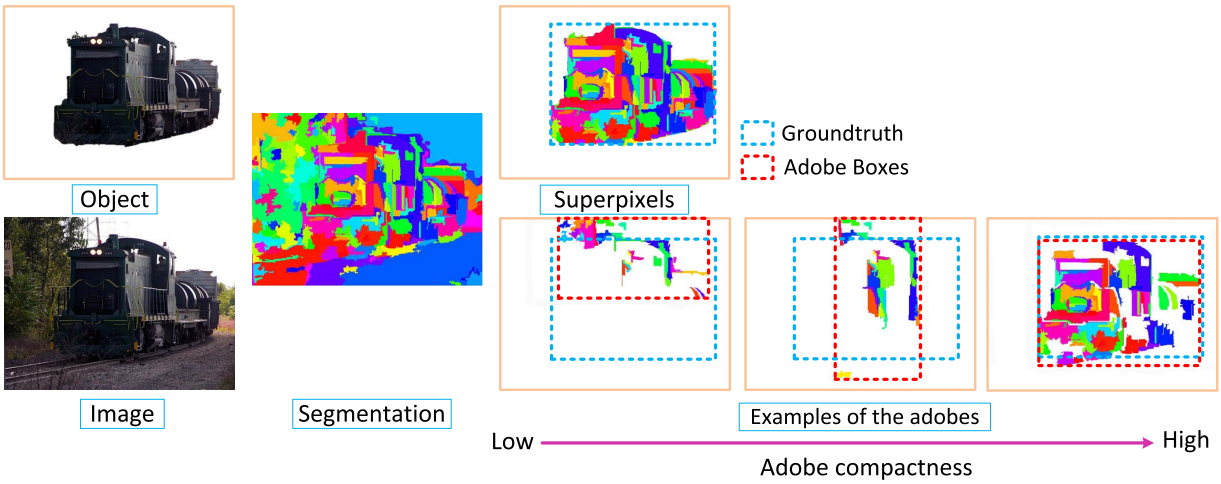


Fig. 1. Our main research idea. From left to right, after acquiring the superpixels using the image segmentation methods in [14], the adobes will be extracted. Adobe Boxes are then fixed according to the principle of the adobe compactness, to locate the object proposals. This figure is best viewed in color.

implies that spatially scattered parts are not likely to form an object, we introduce a new measure of objectness named adobe compactness to rank the proposals and restrain the redundant proposals. Several proposal samples of the different adobe compactness values are shown in Fig. 1, respectively.

Our approach declares three main contributions. *First*, the concept of object adobes is proposed to extract the salient object "parts". Using the object adobes, the object proposals (i.e., Adobe Boxes) can be located without knowing the whole objects. *Secondly*, adobe compactness is introduced as a new objectness measure, to answer the question "What is an object?" [8] from the perspective of object adobe spatial distribution. *Last*, Adobe Boxes can also be reemployed as the refinement tool to enhance the detection rate of other object proposal approaches (e.g., [7], [10], [12], [18]).

We test Adobe Boxes on four challenging datasets (PASCAL VOC2007 [19], VOC2010 [20], VOC2012 [21] and ILSVRC2014 [22]). The experimental results demonstrate that, compared to the state-of-the-arts object proposal generation approaches (e.g., BING [7], EdgeBoxs [12], SEL [10], OBJ [18], MTSE [23]), our method outperforms all of them on DR. Especially, Adobe Boxes achieve higher DR with fewer windows, e.g. over 90% DR with 200 proposals. Our model is also efficient. The average time that consumed on one image is $48ms$, which is almost in real-time. Next, the object adobe and adobe compactness are used for [7], [10], [12], and [18] as a DR refinement mechanism. For example, by equipping BING [7] with our object adobes, denoted as Adobe Boxes$_B$, it achieves over 97% DR with 1000 proposals and effectively promotes DR with the 1^{st} proposal.

The source code of this work is published online.[1]

The remaining of this paper is organized as follows. The related work is discussed in Sec. II. Then Adobe Boxes and the generation of the initial window are illustrated in Sec. III and IV respectively. Experiments and discussions are conducted in Sec. V. Sec. VI concludes the whole paper.

[1]http://pan.baidu.com/s/1jHABDD8 (baiduPan), https://github.com/fzw310/AdobeBoxes-v1.0-.git (GitHub)

II. Related Work

Generally, the existing methods of object proposals can be categorized into three main groups: patch-based model, superpixel-based model, and part-based model.

A. Patch-Based Model

Under this paradigm, the proposed approaches [7], [8], [11], [12], [23], [24] mainly focus on extracting the discriminative visual features able to distinguish the objects and background well within the local image patches. After calculating the objectness score, the patches of high scores will be pushed up as the object proposals by a ranking procedure. In particular, various of visual cues (e.g., edge density, contrast, etc.) are integrated under the Bayesian framework for objectness estimation in [8]. The gradient information is employed by [7] and [11] as the generic object representative feature for predicting the objectness score, in the supervised manner. That is, a classifier (i.e., SVM) will be trained with the gradient feature for objectness estimation. Rather than gradient, edge [25] is further regarded as a more informative and robust objectness clue by Zitnick and Dollár [12]. Based on the sparse information of edges, the probability of whether a local patch contains the object is consequently calculated [12]. Some other works also investigate the usage of object position information to leverage the performance. For instance, the hottest object positions are mined from the training samples in [24]. Being constrained with this prior information, the detection rate of the first 10 candidate windows can be significantly boosted. In [23], the multi-thresholding straddling expansion (MTSE) is proposed via superpixel-tightness analysis, to alleviate the object localization bias of the patches. Although different kinds of visual features have already been employed for object proposal generation, it is still hard to judge which one is the optimal choice. According to the reported experimental results, they keep the similar performance.

B. Superpixel-Based Model

To generate the object proposals, another main technical stream [8]–[10], [13], [26]–[30] resorts to merging

the superpixels, or executing saliency analysis on the superpixels. In this way, the essential concern is how to define the similarity/dissimilarity measure among the superpixels.

To effectively merge the superpixles, Selective Search [10] proposes various of complementary similarity measures to conduct a hierarchical grouping. Global and local grouping happen simultaneously in [13]. Hierarchical segmentation and multi-scale regions combination are employed in [26]. Based on the randomized version of Prim's algorithm [31], Manen et al. [28] uses the connectivity graph of superpixels to analyze the similarity of the neighboring superpixels. Random Forest is adopted in [29] to learn the grouping rules for the superpixels. After selecting the seeds from all the superpixels, a signed geodesic distance transform is computed for each mask, and the critical level sets are identified to produce object proposals in [30]. The superpixel merging based approaches can achieve high object proposal quality. However, their computational cost is indeed expensive. This somewhat limits their usage for practical applications.

Locating the salient objects via saliency analysis on the superpixels recently draws the researchers' attention. Jiang et al. [27] integrate regional contrast, regional property and regional backgroundness to indicate the salient objects. Feng et al. [9] propose several rules (i.e., appearance proximity, spatial proximity, non-reusability and non-scale-bias) to measure the saliency score of an image region. However, for the non-salient objects these methods fail to work.

These methods generally execute object proposals generation by dividing the whole objects from the background directly. However, without any prior information it is challenging and computationally expensive, especially for the small, non-salient and occluded objects.

C. Part-Based Model

Semantic part plays an important role in object detection, and has achieved great success [2], [32]–[34]. Inspired by [34], Felzenszwalb et al. [2], [33] propose a series of effective frameworks for object detection using part-based model. Furthermore, the sub-part-based description is introduced in [32], and the semantic hierarchy is used to represent the objects. These works indeed demonstrate the importance of the semantic part for object category characterization. However, proposing generic objects using part-based model is still not well investigated. To our knowledge, Cho et al. [35] first pay attention to the part-based object discovery without category information. They tackle the object proposals via part-based matching. Nevertheless, the computational cost of matching is high. And, this approach tends to be confused by the "truncate" objects.

Our work can be categorized into the part-based model. That is, the object adobes are proposed to capture the local salient object parts. And, the adobe compactness is introduced to rank the proposals, from the perspective of part spatial distribution.

III. Adobe Boxes

The Adobe Boxes are proposed as the object proposals that tightly enclose the adobes. The proposition of the object adobes is mainly inspired by the human visual attention mechanism [15]. That is, to catch the object, human will first capture several object parts of significant appearance difference from the background quickly. Then, the whole object can be effectively located according to these perceived parts, without knowing all the object components. Following this principium, the object adobe is generally defined as the superpixel with high local color contrast to the background. The spatial compactness of the adobes can essentially evaluate the objectness of the Adobe Boxes. The higher the compactness is, the more probable that the Adobe Boxes correspond to the objects. In this section, we will introduce how to extract the object adobes and Adobe Boxes respectively, and the calculation of the adobe compactness is also illustrated.

A. Object Adobe Extraction

To extract the object adobes, image segmentation procedure [14] is first executed. The yielded superpixels $V = \{s_n, n \in \{1, 2, \ldots N\}\}$ are described by the normalized HSV color histogram $h_n \in \Re^{75}$ [10]. The histogram intersection distance [10] is used to measure the distance between the superpixels as

$$dist_{\mathrm{HI}}(s_n, s_m) = 1 - \sum_{k=1}^{75} \min\left(h_n^k, h_m^k\right). \tag{1}$$

Let B_w[2] denote an initial window that may contain the object. The object adobes can be extracted as illustrated in Fig. 2. Corresponding to the local perspective of B_w, the background superpixel subset $\mathcal{S}_b$ and the internal superpixel subset $\mathcal{S}_i$ are first extracted. The superpixels in $\mathcal{S}_b$ intersect with B_w, and the superpixels in $\mathcal{S}_i$ locate within B_w. It is worth noting that, $\mathcal{S}_i$ does not involve the superpixels that touch the image boundary, to avoid the object-background ambiguity. Using $\mathcal{S}_b$ and $\mathcal{S}_i$, the superpixels that correspond to the object with high probability can be consequently extracted to form the object seed superpixel subset $\mathcal{S}_s$. To extract the object adobes, we further define a candidate adobe subset $\mathcal{S}_c$ to include the superpixels that indicate the potential object adobes. The $\mathcal{S}_c$ superpixels also locate within B_w, but may touch the image boundary. Compared to $\mathcal{S}_i$, more potential object components can be involved in $\mathcal{S}_c$. Obviously, $\mathcal{S}_i \subseteq \mathcal{S}_c$ as shown in Fig. 4. By executing the local contrast analysis among $\mathcal{S}_b$, $\mathcal{S}_s$ and $\mathcal{S}_c$, the object adobe subset $\mathcal{S}_o$ is finally extracted from $\mathcal{S}_c$ as

$$\mathcal{S}_o = \{s \in \mathcal{S}_c | C(s, \mathcal{S}_b) \geqslant C(s, \mathcal{S}_s)\}, \tag{2}$$

where s represents the superpixel and $C(*, *)$ is the local contrast calculation function given by

$$C(s, \mathcal{S}_t) = \frac{1}{N_{\mathcal{S}_t}} \sum_{s_k \in \mathcal{S}_t} dist_{\mathrm{HI}}(s, s_k), \quad t \in \{b, c, i, s\}, \tag{3}$$

where $N_{\mathcal{S}_t}$ denotes the number of the superpixels in $\mathcal{S}_t$. Generally, the more salient the object is, the more "parts" will

[2] B_w can be obtained via the different approaches, such as the randomly sampled windows, and the object proposal methods in [7], [10], [12], and [18].

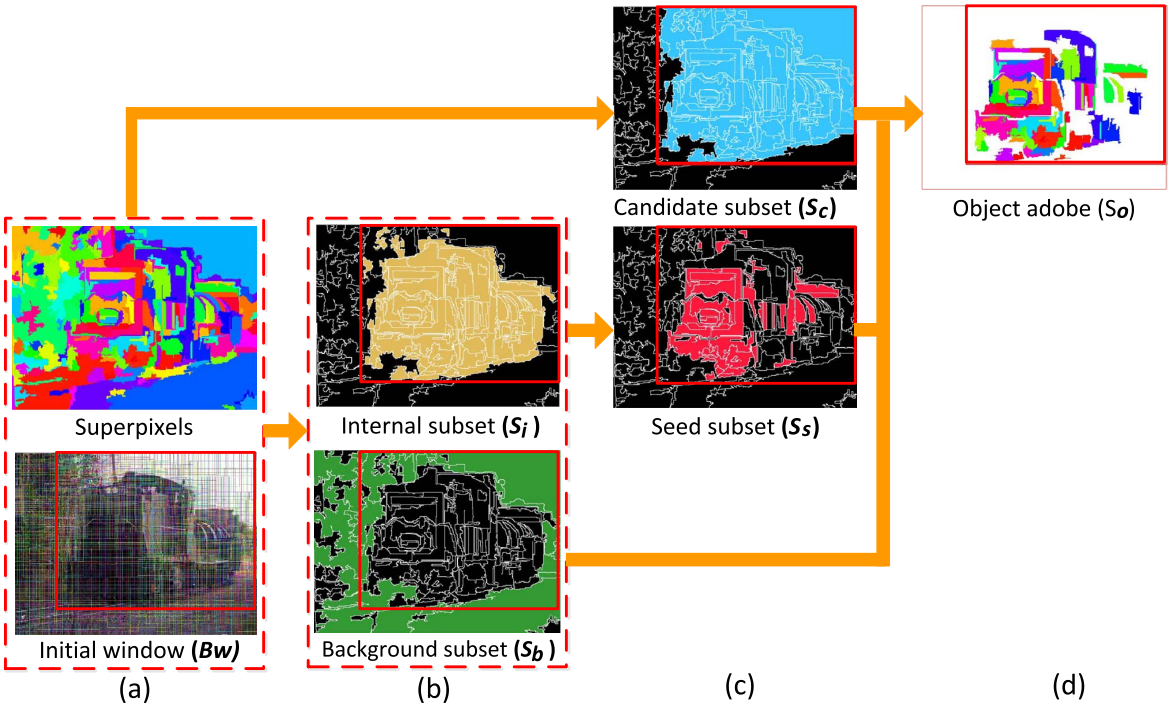


Fig. 2. The detailed object adobe extraction procedure.

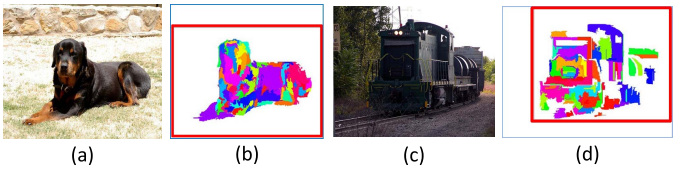


Fig. 3. Examples of the extracted object adobes. (a) Dog. (b) Gog's adobes. (c) Train. (d) Train's adobes.

be extracted from the object. Fig. 3 shows two examples of object adobe extraction. It can be observed that, the "dog" is more salient than the "train". Consequently, more components of the "dog" will be extracted as the object adobes than the "train".

Next, we will illustrate how to extract $\mathcal{S}_b$, $\mathcal{S}_i$, $\mathcal{S}_s$ and $\mathcal{S}_c$ in details, respectively. Fig. 2 shows the detailed adobe extraction procedure for the "train" in Fig. 1.

1) Background Subset $\mathcal{S}_b$: The superpixels in $\mathcal{S}_b$ correspond to the background components. They locate around B_w, and intersect with the boundary of B_w. As a consequence, $\mathcal{S}_b$ can be defined as

$$\mathcal{S}_b = \left\{ s \in V | s \cap \Delta_{B_w} \neq \varnothing \right\}, \tag{4}$$

where Δ_{B_w} represents the boundary of B_w. The green region in Fig. 2 (b) shows the example of $\mathcal{S}_b$.

2) Internal Subset $\mathcal{S}_i$: The superpixels in $\mathcal{S}_i$ locate within B_w. Among them, we intend to find the ones that belong to the object with high probability as the object seed superpixels. Since the objects may be cropped by the image boundary in some cases as shown in Fig. 4, it is difficult to judge whether the superpixels near the image boundary belong to the object or background. As a consequence, the superpixels

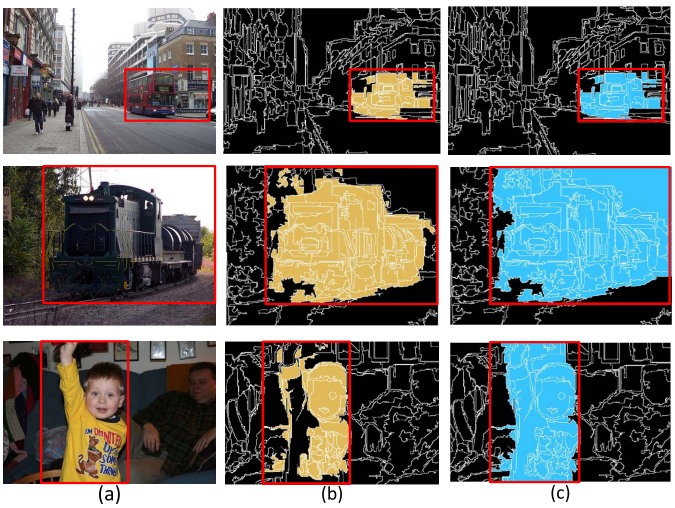


Fig. 4. Illustration of the relationship between $\mathcal{S}_i$ and $\mathcal{S}_c$. It can be observed clearly that $\mathcal{S}_i \subseteq \mathcal{S}_c$. (a) Initial window. (b) Internal subset (S_i). (c) Candidate subset (S_c).

that locate within B_w but touch the image boundary will be discarded from $\mathcal{S}_i$. According to this, $\mathcal{S}_i$ can be obtained by

$$\mathcal{S}_i = \left\{ s \in V | \frac{|s \cap B_w|}{|s|} = 1, s \cap \Delta_{B_w} = \varnothing \right\}, \tag{5}$$

where $|*|$ indicates the cardinality of $*$. The yellow regions in Fig. 2 (b) and Fig. 4 are the internal superpixel examples.

3) Seed Subset $\mathcal{S}_s$: Finding the object seed superpixels from $\mathcal{S}_i$ comprises of two steps. First, for each superpixel $s \in \mathcal{S}_i$, its local contrast to $\mathcal{S}_b$ is calculated to measure its probability of belonging to the object. Then, after ranking all the superpixels in descending order, the top ones will be chosen to form $\mathcal{S}_s$.

Meanwhile, the whole size of the selected $\mathcal{S}_s$ superpixels needs to satisfy

$$\sum_{s\in\mathcal{S}_s} |s| \geqslant \rho \times |B_w|, \tag{6}$$

where $\rho \in (0\ 1)$ is a tunable parameter. The red regions in Fig. 2 (c) show the examples of object seed superpixels.

4) Candidate Subset $\mathcal{S}_c$: In $\mathcal{S}_i$, the superpixels that touch the image boundary are ignored to avoid the object-background ambiguity. However, this may lead to the miss of some essential object components. To address this, the previously discarded superpixels around the image boundary are appended to $\mathcal{S}_i$ to form $\mathcal{S}_c$, such that $\mathcal{S}_i \subseteq \mathcal{S}_c$ as shown in Fig. 4. The blue regions in Fig. 2 (c) and Fig. 4 show the examples of the candidate adobes.

Fig. 4 gives several examples to illustrate the concept of $\mathcal{S}_i$ and $\mathcal{S}_c$. It is worth noting that, the hand and waist of "boy" are truncated by the image boundary. The corresponding superpixels are ignored by $\mathcal{S}_i$. However, they are still remained in $\mathcal{S}_c$ as the candidate adobes.

B. Adobe Box Extraction

After extracting the object adobes, B_w will be refined to B_p that tightly encloses the adobes as the Adobe Box. There remains a question that, whether the Adobe Boxes will miss the objects, especially when not all the object components are captured as the object adobes to localize the objects. To answer this, B_w is first set as the ground truth box B_{gt} of the object. Then, we investigate whether the Adobe Box B_p extracted from B_{gt} possesses high intersection over union (IoU) with B_{gt}. Intuitively, the higher the IoU is, the better that B_p captures the object. IoU between B_p and B_{gt} is given by

$$IoU\left(B_p, B_{gt}\right) = \frac{|B_p \cap B_{gt}|}{|B_p \cup B_{gt}|}. \tag{7}$$

The tests are carried out on VOC2007, VOC2010, VOC2012 and ILSVRC2014 respectively. The normalized IoU probability distribution on the four datasets is shown in Fig. 5. It can be observed that, most of the IoUs are of high values (i.e., $IoU \geq 0.5$). According to the PASCAL-overlap 0.5-criterion [7], [8], [10], [12], we draw the conclusion that Adobe Boxes can generally capture the objects well.

C. Adobe Compactness

How to rank the object proposals is also an important issue. For the extracted Adobe Boxes, the adobe compactness is proposed as the objectness measure for ranking by us. Our proposition is that, for the Adobe Box B_p, the more compactly the object adobes spatially distribute, the more probably that B_p captures the object. Based on this, the adobe compactness for B_p is defined as

$$AC\left(B_p\right) = \frac{\sum_{s\in\mathcal{S}_o} |s|}{|B_p|}. \tag{8}$$

Fig. 6 gives some intuitive examples to demonstrate the effectiveness of $AC(B_p)$. It can be clearly observed that for

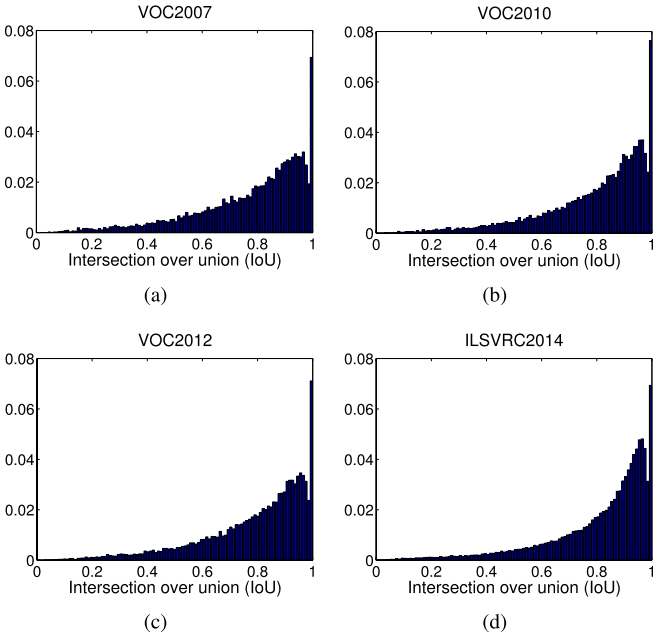


Fig. 5. IoU distribution between B_p and B_{gt} on the four test datasets.

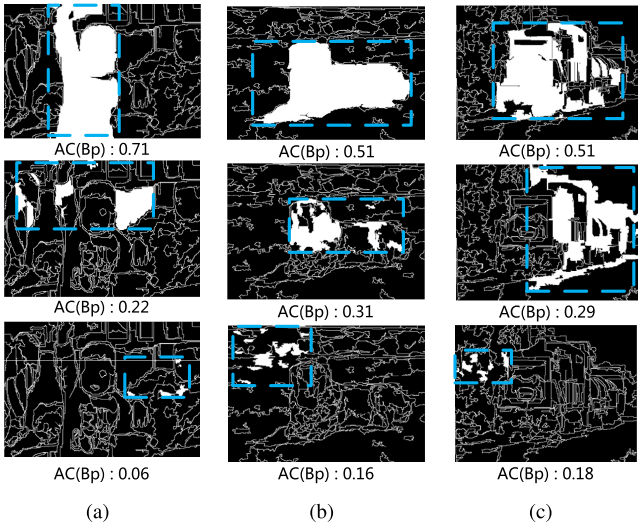


Fig. 6. Adobe compactness examples that correspond to the different kinds of Adobe Boxes. The blue dashed boxes are the Adobe Boxes, and the white regions indicate the adobes.

"boy", "dog" and "train", the object adobes spatially distribute more compactly within B_p, the higher $AC(B_p)$ is and more probably that B_p corresponds to the target objects.

Besides the intuitive examples above, more detailed investigation is further conducted to support our claim on VOC2007, VOC2010, VOC2012 and ILSVRC2014 datesets respectively. That is, 2000 background boxes of IoU less than 0.5 with the object ground truth boxes are randomly sampled from each image. Then, the comparison on the normalized adobe compactness probability distribution between the background and object boxes are executed on each dataset. Fig. 7 shows the comparison results. We can see that, on all the four test datasets the object generally possesses higher adobe compactness than the background. This phenonmenon demonstrates

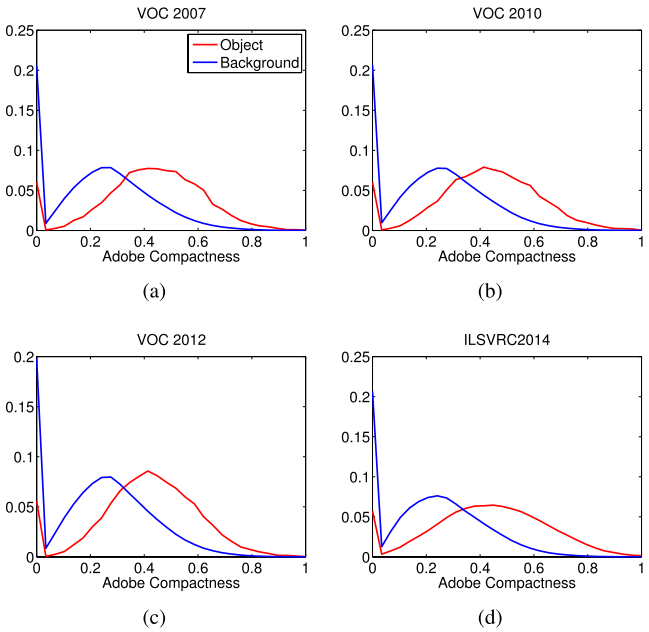


Fig. 7. Adobe compactness distribution comparison between the object and background on the four test datasets.

that, adobe compactness is a feasible objectness measure that can generally push the object forward.

In addition, for the Adobe Boxes of the same adobe compactness, we further propose that the ones extracted from the larger B_w should keep the higher objectness measure. As aforementioned in Fig. 3, generally the more salient the object is, the relatively more object "parts" will be captured as the object adobes. Consequently, the higher adobe compactness tends to be acquired. Hence, for the adobe boxes of the same adobe compactness, larger B_w indicates the salient characteristics from the more global perspective. Thus, the Adobe Boxes with larger B_w should be emphasized. According to this point, the adobe compactness based objectness measure is defined as

$$\begin{aligned} O\left(B_p\right) &= \delta\left(|B_w|\right) \times \left(AC\left(B_p\right)\right)^* \\ &= \frac{1}{1-\left(log_2|B_w|\right)^*} \times \left(AC\left(B_p\right)\right)^*, \end{aligned} \tag{9}$$

where $\delta\left(|B_w|\right)$ is the weighting function that concerns to the size of B_w (i.e., $|B_w|$),

$$\left(log_2|B_w|\right)^* = \frac{log_2|B_w|}{\max\left\{log_2|B_w|\right\}}, \tag{10}$$

and

$$\left(AC\left(B_p\right)\right)^* = \frac{AC\left(B_p\right)}{\max\left\{AC\left(B_p\right)\right\}}, \tag{11}$$

where $\max\left\{log_2|B_w|\right\}$ and $\max\left\{AC\left(B_p\right)\right\}$ indicate the maximum value among all the $log_2|B_w|$ and $AC\left(B_p\right)$ in each image, respectively. The $O(B_p)$ value that corresponds to $\max\left\{log_2|B_w|\right\}$ is set as the highest to avoid the infinitude number computation issue. Indeed, $O(B_p)$ is a monotonically increasing function of $|B_w|$.

Using $O\left(B_p\right)$, the extracted Adobe Boxes will be ranked in descending order. Then, Non-Maximal Suppression (NMS) is executed to reduce the redundant object proposal boxes.

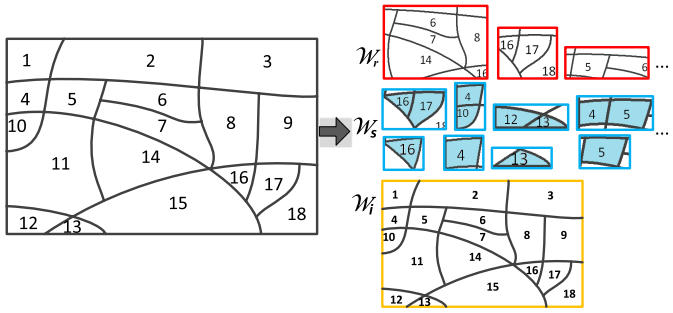


Fig. 8. Examples of $\mathcal{W}_r$, $\mathcal{W}_s$ and $\mathcal{W}_i$.

IV. Initial Object Proposal Window Generation

As mentioned before, to extract the Adobe Box B_p, the initial object proposal window B_w is required to coarsely locate the object. In this section, we will explain how to generate B_w. Actually, B_w can be acquired by different ways, i.e., random sampling or using existing object proposal approaches [7], [10], [12], [18]. In our experiments, we will demonstrate that the yielded Adobe Boxes can improve the performance for all of them.

Without loosing the generality, random sampling is employed to illustrate the proposed B_w generation method. In total, 2000 windows are randomly sampled from the whole image uniformly. The obtained window set is denoted as $\mathcal{W}_r$. Since the randomly sampled windows tends to miss the small objects, after the image segmentation procedure [14], the bounding boxes that tightly surround the individual superpixels or two neighboring superpixels are also considered. The resulting window set is indicated as $\mathcal{W}_s$. Additionally, the whole image window $\mathcal{W}_i$ is appended simultaneously. Fig. 8 shows the examples of $\mathcal{W}_r$, $\mathcal{W}_s$ and $\mathcal{W}_i$. Consequently, the coarse initial object proposal window set comprises of three parts as

$$\mathcal{W}_c = \mathcal{W}_r \cup \mathcal{W}_s \cup \mathcal{W}_i. \tag{12}$$

Let B_w^* denote an arbitrary window in $\mathcal{W}_c$. To help better capture the object, B_w^* will be refined to obtain the initial object proposal window B_w as follows. First, B_w^* is enlarged by the ratio η under the image size limitation. Then, we check the superpixels that partially locate within B_w^*. For the superpixel s, if the condition $\frac{|s \cap B_w^*|}{|s|} \geq 0.9$ is satisfied, it indicates that s should be the internal superpixel of B_w^*. Consequently, B_w^* is further adjusted to fully include s tightly. After doing this for all the concerned superpixles, the resulting B_w^* is finally regarded as B_w.

V. Experiments

In experiments, following [7], [12] we focus on investigating Adobe Boxes' detection rate (DR) and time consumption to demonstrate its effectiveness and efficiency. Specifically, the well known PASCAL-overlap 0.5-criterion [7], [8], [10], [12] is employed to measure DR. Following the evaluation principle in [7], the DR-#WIN curve is used to reveal the relationship between DR and the number of the proposals. For each test dataset, the average processing time consumption per image is also reported.

Four challenging datasets (i.e., PASCAL VOC2007 [19] PASCAL VOC2010 [20], PASCAL VOC2012 [21] and ILSVRC2014 [22]) are employed for test. For VOC2007, 4952 images in the test set are chosen for performance evaluation. Since the ground truth bounding boxes for the test sets are not released by VOC2010, VOC2012 and ILSVRC2014 (classification+localization task with 1000 categories), the corresponding validation sets are used for test instead. In particular, 6323, 11400 and 49032 images are employed respectively. The objects annotated as "difficult" are excluded from the test sets by us. The test images are required to be no larger than 1024×1024, in all the cases.

Since most of the objects in the test datasets contain more than 256 pixels, for superpixel generation [14] 128 is empirically chosen as the minimum superpixel size. There are also two other tunable parameters for extracting Adobe Boxes. They are, the foreground percentage parameter ρ in Sec. III-A, and the amplification parameter η in Sec. IV. They are set to 25% and $\frac{1}{4}$ respectively, during the whole phase of the experiments. Adobe Boxes' performance sensitivity to them will also be investigated.

The experimental results are organized as follows. In Sec. V-A, the performance of Adobe Boxes is tested in the case that the initial proposal windows are generated via random sampling. The other state-of-the-art object proposal approaches: BING [7], EdgeBoxes [12], SEL [10], SEL-Fast [10], OBJ [18] and MTSE [23] will also be included for comparison. Then, we will demonstrate that Adobe Boxes can also leverage the performance of the other object proposal methods as a refinement procedure, in Sec. V-B.

For the practical applications, Sec. V-C gives the recommendation on how to use Adobe Boxes. That is, refining BING with Adobe Boxes can achieve the relatively good balance between the performance and time consumption.

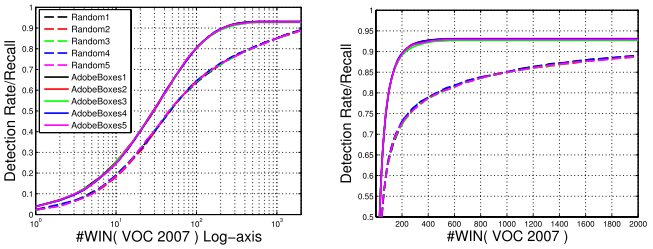
Adobe Boxes are mainly derived from the image segmentation results [14]. GrabCut [36] is another well known image segmentation method that is able to separate the objects from the background well. In Sec. V-D, we will compare the performance of Adobe Boxes with the object proposal approach using GrabCut, from the perspectives of effectiveness and efficiency simultaneously.

The performance of Adobe Boxes using multi-scale superpixel will be analyzed in Sec. V-E. In Sec. V-F, Adobe Boxes will be further compared with the other methods using the DR-overlap criteria. And, the parameter sensitivity is investigated in Sec. V-G.

All the experiments are conducted on the same PC with i7-4770 CPU and 32G RAM.

A. Adobe Boxes With the Randomly Sampled Initial Proposal Windows

In this section, two kinds of experiments are conducted to evaluate the performance of Adobe Boxes. First, Adobe Boxes are extracted solely from the randomly sampled object proposal windows to verify the performance enhancement yielded by them. Next, the additional initial proposal window sets $\mathcal{W}_s$ and $\mathcal{W}_i$ are further appended as aforementioned in Sec. IV. In this case, Adobe Boxes is fully running as an end-to-end object proposal approach, and will be compared with the other state-of-the-art object proposal approaches [7], [10], [12], [18], [23].



(a) (b)

Fig. 9. The DR-#WIN curves of the randomly sampled proposal windows and Adobe Boxes on PASCAL VOC2007. $\text{Random}_\mathbf{i}$ and $\text{AdobeBoxes}_\mathbf{i}$ indicate the DR-#WIN curves of the randomly sampled proposal windows and Adobe Boxes for the $\mathbf{i}^{th}$ round, respectively.

TABLE I

THE AVERAGE DR (%) AND STANDARD DEVIATION (%) OF THE RANDOMLY SAMPLED PROPOSAL WINDOWS AND ADOBE BOXES, ON PASCAL VOC2007. THE RESULTS THAT CORRESPOND TO THE FIRST 1, 10, 200 AND 1000 TOP SCORED PROPOSAL WINDOWS ARE LISTED, RESPECTIVELY. STANDARD DEVIATIONS ARE IN PARENTHESES

	DR (1)	DR (10)	DR (200)	DR ($1k$)
Random	2.27 ($\pm$0.34)	18.77 ($\pm$0.44)	72.67 ($\pm$0.34)	85.02 ($\pm$0.05)
AdobeBoxes	**3.85** ($\pm$0.09)	**25.17** ($\pm$0.43)	**89.37** ($\pm$0.16)	**92.98** ($\pm$0.19)

In the first experimental setting, 2000 initial proposal windows in all are randomly sampled per image for 5 rounds on VOC2007. The DR-#WIN curves of the randomly sampled proposal windows and Adobe Boxes are shown in Fig. 9, corresponding to all the 5 rounds. To clearly depict the DR-#WIN curves of the top scored proposals, the logarithmic axes are used in Fig. 9(a). The average DR and standard deviation of the 5-round test are also reported in Table I. From them, we can observe that:

- Adobe Boxes can leverage the DR of the randomly sampled proposal windows significantly. It is worth noting that, using only 200 randomly sampled initial proposal windows, Adobe Boxes can achieve DR near to 90%. These indeed demonstrate the effectiveness of Adobe Boxes;
- For the 5-round test, the DR stand deviation of Adobe Boxes is small. It means that Adobe Boxes can robustly work, corresponding to the initial proposal windows generated in the different conditions.

The second experimental setting is conducted on VOC2007, VOC2010, VOC2012 and ILSVRC2014 simultaneously. The DR-#WIN curves of Adobe Boxes and other methods are shown in Fig. 10. And, the detailed results and average time consumption per image are listed in Table II, Table III, Table IV and Table V. It can be seen that:

- Even only using the randomly sampled initial proposal windows, Adobe Boxes generally can achieve the comparable or even better performance than the other methods;

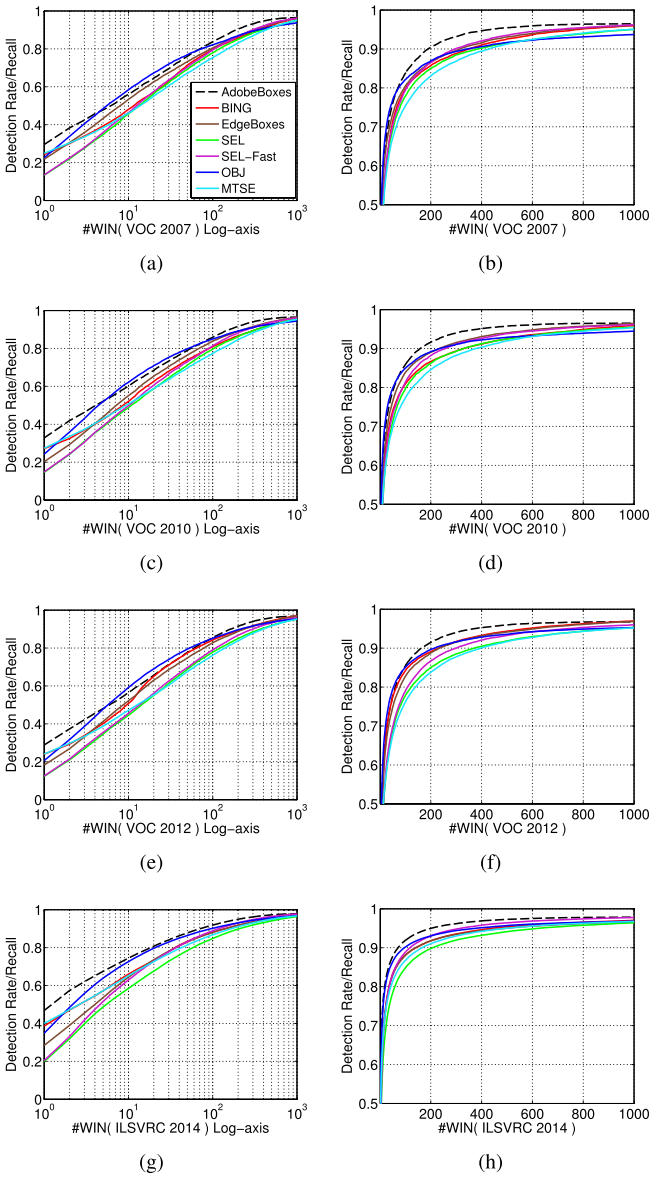


Fig. 10. Comparison between Adobe Boxes with the randomly sampled initial proposal windows and the other state-of-the-art methods on the four test datasets.

• Adobe Boxes outperform most of the other methods when the relatively small number (e.g., 200) of proposals are used. This is an important property that implies that the objects can be efficiently detected within a smaller searching space, using Adobe Boxes;

• The average time consumption per image of Adobe Boxes is around 48 ms, which nearly meets the real-time processing requirement. BING is faster than our method. However, it is inferior to Adobe Boxes on DR, in most of the cases.

B. Refining the Other Methods Using Adobe Boxes

Here, we will demonstrate that Adobe Boxes can also refine the other methods [7], [10], [12], [18] by using their object proposal results as the initial proposal windows. The experiment is executed on PASCAL VOC2007. And, the results are shown in Fig. 11 and Table VI. In particular, the refined BING [7], EdgeBoxes [12], SEL [10] and OBJ [18] are denoted as AdobeBoxes$_B$, AdobeBoxes$_E$, AdobeBoxes$_S$ and AdobeBoxes$_O$ respectively. Another recently proposed refinement approach named MTSE [23] is also included

TABLE II

THE DR (%) AND TIME CONSUMPTION (S) COMPARISON BETWEEN ADOBE BOXES WITH THE RANDOMLY SAMPLED INITIAL PROPOSAL WINDOWS AND THE OTHER METHODS, ON VOC2007. THE APPROACHES WITH PARALLEL PROCESSING ARE MARKED WITH *

Method	DR (1)	DR (10)	DR (200)	DR ($1k$)	Time
AdobeBoxes*	**29.54**	55.89	**90.24**	95.90	0.047
BING*	24.63	48.45	85.98	96.02	**0.003**
EdgeBoxes*	21.79	53.36	86.60	96.09	0.196
SEL	13.39	45.69	85.05	95.00	8
SEL-Fast	13.47	46.36	86.51	**96.19**	1.7
OBJ	22.59	**58.41**	86.97	93.75	3
MTSE	24.92	46.09	83.12	95.05	0.125

TABLE III

THE DR (%) AND TIME CONSUMPTION (S) COMPARISON BETWEEN ADOBE BOXES WITH THE RANDOMLY SAMPLED INITIAL PROPOSAL WINDOWS AND THE OTHER METHODS, ON VOC2010. THE APPROACHES WITH PARALLEL PROCESSING ARE MARKED WITH *

Method	DR (1)	DR (10)	DR (200)	DR ($1k$)	Time
AdobeBoxes*	**32.84**	60.03	**91.44**	96.17	0.045
BING*	27.18	52.06	86.65	95.88	**0.003**
EdgeBoxes*	20.29	55.10	89.06	**96.45**	0.185
SEL	14.63	48.76	86.18	95.48	7.9
SEL-Fast	14.90	49.80	88.17	96.11	1.7
OBJ	24.25	**62.13**	89.22	94.46	2.8
MTSE	27.38	50.78	84.57	95.40	0.125

TABLE IV

THE DR (%) AND TIME CONSUMPTION (S) COMPARISON BETWEEN ADOBE BOXES WITH THE RANDOMLY SAMPLED INITIAL PROPOSAL WINDOWS AND THE OTHER METHODS, ON VOC2012. THE APPROACHES WITH PARALLEL PROCESSING ARE MARKED WITH *

Method	DR (1)	DR (10)	DR (200)	DR ($1k$)	Time
AdobeBoxes*	**28.86**	55.89	**91.47**	96.41	0.047
BING*	24.03	50.75	89.09	**96.97**	**0.003**
EdgeBoxes*	18.29	52.17	88.77	96.87	0.198
SEL	12.32	44.53	85.01	95.33	8.1
SEL-Fast	12.54	45.54	86.70	96.01	1.8
OBJ	20.66	**59.19**	89.64	95.27	2.9
MTSE	24.10	47.13	83.77	95.23	0.122

TABLE V

THE DR (%) AND TIME CONSUMPTION (S) COMPARISON BETWEEN ADOBE BOXES WITH THE RANDOMLY SAMPLED INITIAL PROPOSAL WINDOWS AND THE OTHER METHODS, ON ILSVRC2014. THE APPROACHES WITH PARALLEL PROCESSING ARE MARKED WITH *

Method	DR (1)	DR (10)	DR (200)	DR ($1k$)	Time
AdobeBoxes*	**46.85**	**74.02**	**94.79**	97.61	0.051
BING*	38.64	66.00	91.77	97.01	**0.003**
EdgeBoxes*	28.43	64.42	91.75	96.64	0.209
SEL	19.92	58.45	89.79	96.40	8.6
SEL-Fast	20.40	62.97	93.02	**97.75**	2.33
OBJ	34.88	72.60	93.03	96.81	2.5
MTSE	40.03	65.18	91.02	96.75	0.147

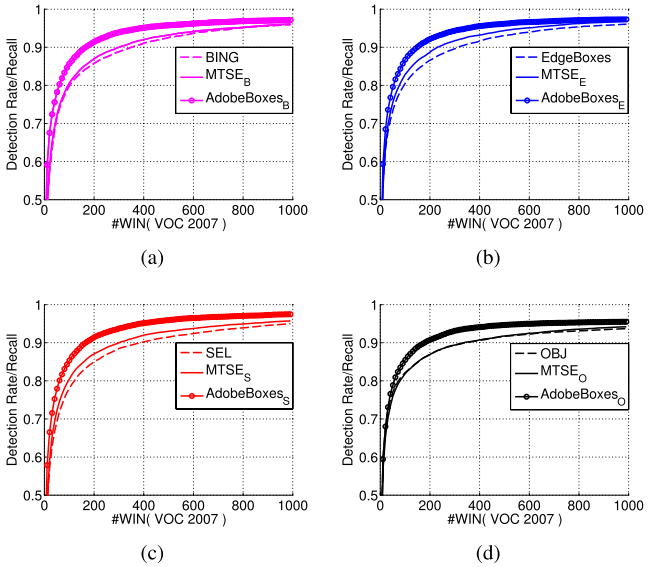


Fig. 11. DR-#WIN curve comparison between the object proposal methods [7], [10], [12], [18] and the corresponding counterparts refined by Adobe Boxes and MTSE on PASCAL VOC2007.

TABLE VI

DR (%) COMPARISON BETWEEN THE OBJECT PROPOSAL METHODS [7], [10], [12], [18] AND THE CORRESPONDING COUNTERPARTS REFINED BY ADOBE BOXES AND MTSE ON PASCAL VOC2007

	DR (1)	DR (10)	DR (200)	DR ($1k$)
BING	24.63	48.45	85.98	96.02
MTSE_B	20.51	46.73	86.99	96.14
$\text{AdobeBoxes}_\text{B}$	**29.54**	**57.21**	**91.70**	**97.16**
EdgeBoxes	21.79	53.36	86.60	96.09
MTSE_E	18.14	51.30	88.72	96.90
$\text{AdobeBoxes}_\text{E}$	**29.79**	**57.60**	**91.92**	**97.33**
SEL	13.39	45.69	85.05	95.00
MTSE_S	13.14	46.03	87.04	95.75
$\text{AdobeBoxes}_\text{S}$	**29.56**	**56.46**	**90.95**	**97.52**
OBJ	22.59	**58.41**	86.97	93.75
MTSE_O	23.78	55.39	86.93	94.22
$\text{AdobeBoxes}_\text{O}$	**29.95**	57.89	**90.53**	**95.42**

for comparison. Its results on BING, EdgeBoxes, SEL and OBJ are respectively denoted as MTSE_B, MTSE_E, MTSE_S and MTSE_O on counterpart. We can see that:

- Adobe Boxes can improve the performance of the other approaches, almost in all the cases;
- With 200 object proposals, the DR of all the methods are enhanced beyond 90% by Adobe Boxes.
- Adobe Boxes is consistently superior to MTSE. The reason seems that, MTSE ranks the object proposals by imposing randomness to the objectness score like SEL. Nevertheless, this may not reduce the proposal redundancy effectively. Thus, DR increases relatively slowly.

The results above indeed verify that, besides being an end-to-end object proposal method, Adobe Boxes can also be regarded as an effective refinement tool to leverage the other methods' performance.

C. Practical Application Recommendation

BING [7] possesses the ultra fast computational speed (i.e. 300fps) for object proposal. Towards the practical

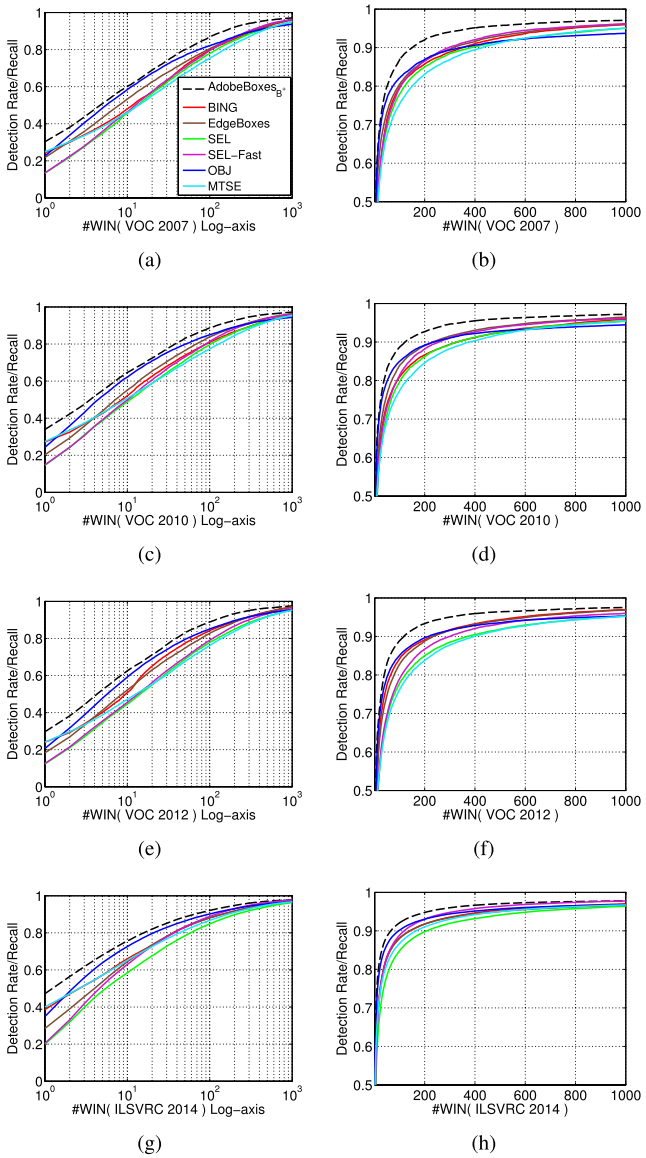


Fig. 12. DR-#WIN curve comparison between the proposed AdobeBoxes_{B^+} and the other methods on the four test datasets.

applications, we recommend that refining BING with Adobe Boxes is the feasible choice that can achieve good balance between effectiveness and efficiency. Besides replacing the initial proposal windows with BING's results, we also modify the objectness score computation method accordingly. That is, since the objectness score yielded by BING already takes the proposal window size into consideration, we substitute the weighting factor $\delta(|B_w|)$ in Eqn. 9 with BING's score for the adobe compactness based objectness measure calculation. The newly refined BING via Adobe Boxes is denoted as AdobeBoxes_{B^+}. And, it is compared with the other methods on PASCAL VOC2007, VOC2010, VOC2012 and ILSVRC2014. The experimental results are shown in Fig. 12, Table VII, Table VIII, Table IX and Table X respectively. We can see that:

- In almost all the cases, AdobeBoxes_{B^+} consistently outperforms the other methods, especially with the relatively small number of proposals (i.e., less than 200);

TABLE VII

THE DR (%) AND TIME CONSUMPTION (S) COMPARISON BETWEEN ADOBEBOXES$_{B+}$ AND THE OTHER METHODS ON VOC2007. THE APPROACHES IMPLEMENTED WITH PARALLEL PROCESSING ARE MARKED WITH *

Method	DR (1)	DR (10)	DR (200)	DR ($1k$)	Time
AdobeBoxes$^*_{B+}$	**30.30**	**60.14**	**92.11**	**97.10**	0.045
AdobeBoxes*	29.54	55.89	90.24	95.90	0.047
BING*	24.63	48.45	85.98	96.02	**0.003**
EdgeBoxes*	21.79	53.36	86.60	96.09	0.196
SEL	13.39	45.69	85.05	95.00	8
SEL-Fast	13.47	46.36	86.51	96.19	1.7
OBJ	22.59	58.41	86.97	93.75	3
MTSE	24.92	46.09	83.12	95.05	0.125

TABLE VIII

THE DR (%) AND TIME CONSUMPTION (S) COMPARISON BETWEEN ADOBEBOXES$_{B+}$ AND THE OTHER METHODS ON VOC2010. THE APPROACHES IMPLEMENTED WITH PARALLEL PROCESSING ARE MARKED WITH *

Method	DR (1)	DR (10)	DR (200)	DR ($1k$)	Time
AdobeBoxes$^*_{B+}$	**33.83**	**64.19**	**92.84**	**97.31**	0.048
AdobeBoxes*	32.84	60.03	91.44	96.17	0.045
BING*	27.18	52.06	86.65	95.88	**0.003**
EdgeBoxes*	20.29	55.10	89.06	96.45	0.185
SEL	14.63	48.76	86.18	95.48	7.9
SEL-Fast	14.90	49.80	88.17	96.11	1.7
OBJ	24.25	62.13	89.22	94.46	2.8
MTSE	27.38	50.78	84.57	95.40	0.125

TABLE IX

THE DR (%) AND TIME CONSUMPTION (S) COMPARISON BETWEEN ADOBEBOXES$_{B+}$ AND THE OTHER METHODS ON VOC2012. THE APPROACHES IMPLEMENTED WITH PARALLEL PROCESSING ARE MARKED WITH *

Method	DR (1)	DR (10)	DR (200)	DR ($1k$)	Time
AdobeBoxes$^*_{B+}$	**29.63**	**61.93**	**93.14**	**97.61**	0.048
AdobeBoxes*	28.86	55.89	91.47	96.41	0.047
BING*	24.03	50.75	89.09	96.97	**0.003**
EdgeBoxes*	18.29	52.17	88.77	96.87	0.198
SEL	12.32	44.53	85.01	95.33	8.1
SEL-Fast	12.54	45.54	86.70	96.01	1.8
OBJ	20.66	59.19	89.64	95.27	2.9
MTSE	24.10	47.13	83.77	95.23	0.122

TABLE X

THE DR (%) AND TIME CONSUMPTION (S) COMPARISON BETWEEN ADOBEBOXES$_{B+}$ AND THE OTHER METHODS ON ILSVRC2014. THE APPROACHES IMPLEMENTED WITH PARALLEL PROCESSING ARE MARKED WITH *

Method	DR (1)	DR (10)	DR (200)	DR ($1k$)	Time
AdobeBoxes$^*_{B+}$	**47.18**	**75.62**	94.75	**97.78**	0.052
AdobeBoxes*	46.85	74.02	**94.79**	97.61	0.051
BING*	38.64	66.00	91.77	97.01	**0.003**
EdgeBoxes*	28.43	64.42	91.75	96.64	0.209
SEL	19.92	58.45	89.79	96.40	8.6
SEL-Fast	20.40	62.97	93.02	97.75	2.33
OBJ	34.88	72.60	93.03	96.81	2.5
MTSE	40.03	65.18	91.02	96.75	0.147

• Besides the DR advantage, AdobeBoxes$_{B+}$ is also efficient. The average time consumption per image around 48ms nearly meets the real-time processing requirement.

According to the experimental results above, BING and EdgeBoxes are the two main competitors of AdobeBoxes$_{B+}$, when considering effectiveness and efficiency simultaneously. Fig. 13 gives some intuitive examples to further compare them.

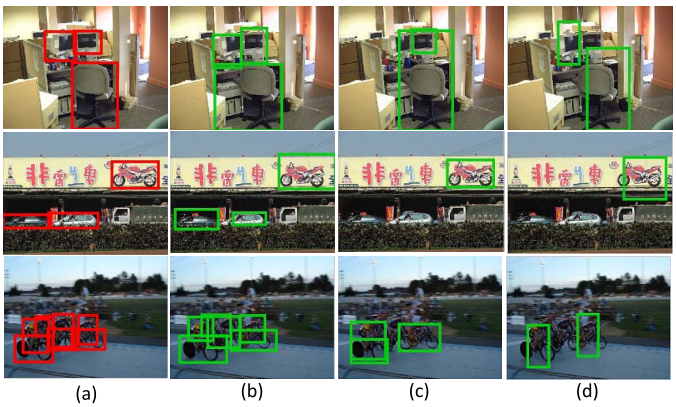


Fig. 13. Intuitive comparison among AdobeBoxes$_{B+}$, BING and EdgeBoxes. For each method, 100 proposals are used. The ground truth windows are labelled as red, and the best proposals of $IoU \geq 0.5$ with the ground truth windows are labelled as green to indicate the available object proposals. (a) GT. (b) Ours. (c) Edge Boxes. (d) Bing.

It can be observed that, AdobeBoxes$_{B+}$ can capture the objects better within the cluttered scenes. The reason seems that, BING and EdgeBoxes heavily rely on the object's holistic edge information. However, parts of the object's edge may be suppressed by the other strong edges that do not belong to the object. It indeed leads to detection loss with the top windows, such as the cars in the second row and the riders in the third row. While, AdobeBoxes$_{B+}$ captures the objects by using the local salient parts, without requiring knowing the holistic object characteristics. Hence, it can alleviate the object edge loss issue above to some degree.

D. Comparison With Object Proposal Approach via GrabCut

To further demonstrate the effectiveness and efficiency of Adobe Boxes, we compare it with another image segmentation based object proposal approach via the well known GrabCut method [36]. In particular, given the initial proposal windows, GrabCut can separate the objects from the background, instead of object adobes. Then, the bounding boxes that tightly surround the extracted proto-objects are regarded as the object proposals. Concerning to efficiency and consistency, BING is employed to yield the initial proposal windows. Since GrabCut is actually time consuming, the more efficient GrabCut in one cut [37] is adopted here. Unfortunately, it still costs about 77 seconds per image to yield the object proposals. The corresponding GrabCut based object proposal approach is denoted as GrabCut$_{\mathrm{B}}$. The comparison between GrabCut$_{\mathrm{B}}$ and our recommended AdobeBoxes$_{B+}$ on PASCAL VOC2007 is shown in Fig. 14. It is obviously that, AdobeBoxes$_{B+}$ consistently outperforms GrabCut$_{\mathrm{B}}$, running much faster as well. The reason for why GrabCut$_{\mathrm{B}}$ is inferior to AdobeBoxes$_{B+}$ seems that the objects may be truncated by the initial proposal windows, which leads to the unsatisfactory GrabCut segmentation results.

E. Adobe Boxes With Multi-Scale Superpixel

In the previous experiments, for Adobe Boxes the minimum superpixel size is only set as 128. This may lead to

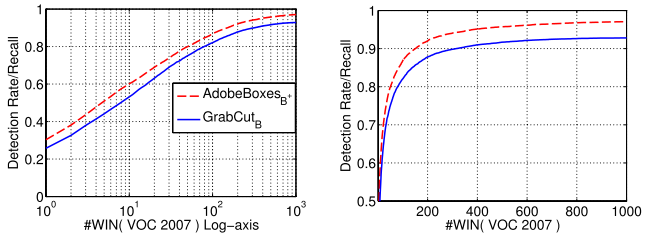


Fig. 14. DR-#WIN curve comparison between AdobeBoxes$_{B+}$ and GrabCut$_B$ on VOC2007.

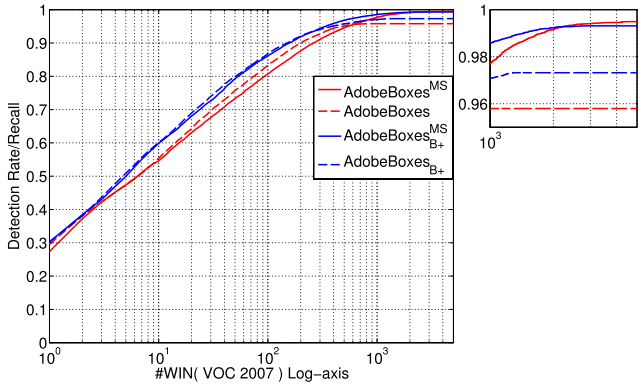


Fig. 15. Illustration of trade-off between DR and #WIN with the enhance strategy. This figure is best viewed in color.

some information loss. In this subsection, we will investigate whether Adobe Boxes's performance can be further leveraged by the using multi-scale superpixel. That is, the minimum superpixel size will be set as 32, 64, 128 and 256 to generate the proposals respectively. All the yielded proposals are considered simultaneously to generate Adobe Boxes using non-maximal suppression. Here, AdobeBoxes and AdobeBoxes$_{B^+}$ using the multi-scale superpixel are termed as AdobeBoxes^{MS} and AdobeBoxes$^{MS}_{B^+}$. The comparison results on PASCAL VOC2007 are shown in Fig. 15 and Table XI. It can be observed that:

- With the increment of proposal amount, Adobe Boxes using the multi-scale superpixel tend to outperform the single-scale superpixel version. In particular, with 5000 proposals AdobeBoxes^{MS} and AdobeBoxes$^{MS}_{B^+}$ achieve the DR of 99.49% and 99.31%. The reason seems that, more small objects can be captured by the multi-scale superpixel;
- Using the multi-scale superpixel, the time consumption of AdobeBoxes is obviously increased. Thus, concerning the balance between effectiveness and efficiency, single-scale superpixel is preferred.

F. DR-Overlap Evaluation

In this section, we will estimate Adobe Boxes from the perspective of DR-overlap curve. In particular, "overlap" indicates the IoU threshold with the ground truth that judges whether an object proposal is available. Fig. 16 shows the DR-overlap curves of the different methods, corresponding to

TABLE XI

THE DR (%) AND TIME CONSUMPTION (S) COMPARISON BETWEEN ADOBE BOXES WITH MULTI-SCALE SUPERPIXEL AND SINGLE-SCALE SUPERPIXEL ON VOC2007. THE APPROACHES IMPLEMENTED WITH PARALLEL PROCESSING ARE MARKED WITH *

Method	DR(2K)	DR(3K)	DR(4K)	DR(5K)	Time
AdobeBoxes*	95.80 %	95.80 %	95.80 %	95.80 %	0.047
AdobeBoxes^{MS*}	99.16 %	**99.41** %	**99.46** %	**99.49** %	1.033
AdobeBoxes$^*_{B+}$	97.32 %	97.32 %	97.32 %	97.32 %	0.048
AdobeBoxes$^{MS*}_{B+}$	**99.23** %	99.31 %	99.31 %	99.31 %	0.338

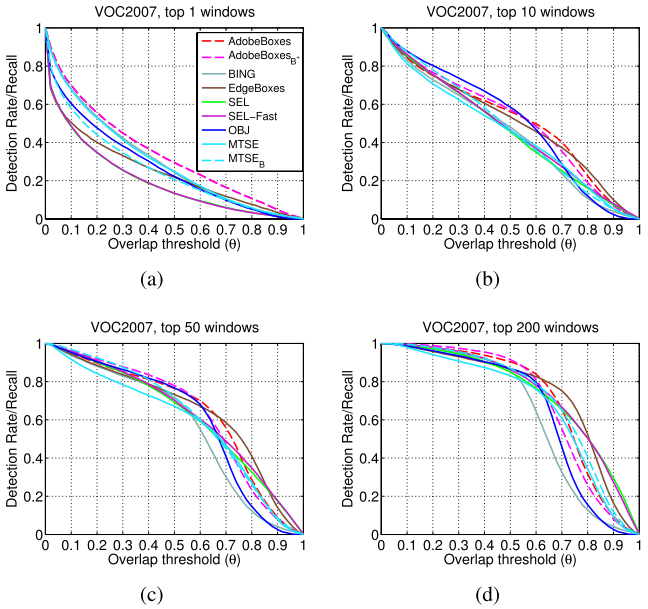


Fig. 16. Comparison of recall-overlap curves with different methods on VOC2007.

the top 1, 10, 50 and 200 scored proposals respectively, on PASCAL VOC2007. It can be observed that:

- Using the standard PASCAL-overlap 0.5-criterion, Adobe Boxes outperforms the other methods in most cases;
- With the increment of overlap threshold, SEL [10] and EdgeBoxs [12] are generally performing better than our method. The reason may be that, they employ several additional procedures to refine the proposal results, such as the complex superpixel merging [10] and elaborate integral image [12]. On the other hand, much higher time consumption is required by SEL and EdgeBoxs.
- MTSE$_B$ performs better than our method with 200 proposals, since it sets five expansion thresholds to reduce the localization bias. Nevertheless, multi-thresholding expansion also yields proposal redundancy. This leads DR to increase relatively slowly as analyzed in Sec. V-B.

The phenomenons above indicate that the performance of Adobe Boxes still need to be enhanced, if the high overlap is required by the applications.

G. Parameter Sensitivity Investigation

As aforementioned, the foreground percentage ρ in Sec. III-A, and the amplification parameter η in Sec. IV are two tunable parameters within Adobe Boxes. Here, the

TABLE XII

PARAMETER SENSITIVITY INVESTIGATION ON η. STANDARD DEVIATIONS (%) ARE LISTED IN PARENTHESES

η	DR (1)	DR (10)	DR (200)	DR ($1k$)
$\frac{1}{2}$	$29.42(\pm 0.63)$	$55.32(\pm 1.58)$	$90.03(\pm 0.26)$	$95.61(\pm 0.31)$
$\frac{1}{3}$	$29.42(\pm 0.63)$	$55.30(\pm 1.54)$	$90.40(\pm 0.31)$	$95.83(\pm 0.29)$
$\frac{1}{4}$	$29.42(\pm 0.63)$	$55.39(\pm 1.48)$	$90.25(\pm 0.22)$	$95.88(\pm 0.25)$
$\frac{1}{5}$	$29.42(\pm 0.63)$	$55.33(\pm 1.52)$	$90.19(\pm 0.23)$	$95.88(\pm 0.29)$
$\frac{1}{6}$	$29.42(\pm 0.63)$	$55.46(\pm 1.57)$	$90.12(\pm 0.31)$	$95.86(\pm 0.25)$

TABLE XIII

PARAMETER SENSITIVITY INVESTIGATION ON ρ. STANDARD DEVIATIONS (%) ARE LISTED IN PARENTHESES

ρ	DR (1)	DR (10)	DR (200)	DR ($1k$)
5%	$30.50(\pm 0)$	$57.56(\pm 0.15)$	$89.84(\pm 0.11)$	$95.10(\pm 0.13)$
10%	$30.10(\pm 0)$	$57.13(\pm 0.15)$	$90.48(\pm 0.14)$	$95.71(\pm 0.12)$
15%	$29.78(\pm 0)$	$56.79(\pm 0.21)$	$90.64(\pm 0.17)$	$95.80(\pm 0.07)$
20%	$29.47(\pm 0)$	$56.17(\pm 0.20)$	$90.61(\pm 0.15)$	$95.88(\pm 0.09)$
25%	$29.54(\pm 0)$	$55.61(\pm 0.18)$	$90.37(\pm 0.13)$	$95.90(\pm 0.12)$
30%	$29.47(\pm 0)$	$55.01(\pm 0.19)$	$90.26(\pm 0.15)$	$95.92(\pm 0.14)$
35%	$29.22(\pm 0)$	$54.49(\pm 0.06)$	$90.22(\pm 0.21)$	$95.97(\pm 0.15)$
40%	$28.94(\pm 0)$	$54.08(\pm 0.09)$	$90.10(\pm 0.13)$	$95.96(\pm 0.11)$
45%	$28.73(\pm 0)$	$53.61(\pm 0.15)$	$90.04(\pm 0.08)$	$95.96(\pm 0.12)$
50%	$28.41(\pm 0)$	$53.17(\pm 0.08)$	$90.03(\pm 0.07)$	$95.93(\pm 0.15)$

performance sensitivity of Adobe Boxes to ρ and η will be investigated. We set $\rho \in \{5\%, 10\%, \ldots 50\%\}$ with stride size 5%, and $\eta \in \{\frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \frac{1}{5}, \frac{1}{6}\}$. The cross-over experimental results are listed in Table XII and Table XIII. For each row in Table XII, the value of η is fixed, and the mean and standard deviation of DR that correspond the different values of ρ are listed, vice versa for Table XIII. It is obviously that, for both ρ and η, the standard deviation on DR is small in all the cases. This demonstrates that Adobe Boxes are not sensitive to them. And, ρ and η are empirically set to 25% and $\frac{1}{4}$.

VI. CONCLUSION

In this paper, we propose a new object proposal method termed Adobe Boxes. Based on the local contrast analysis, the object adobes are first extracted to capture the generic object components. Then, adobe compactness is proposed as a new objectness measurement to rank the object proposals. Adobe Boxes can not only work as an end-to-end object proposal approach using the randomly sampled initial proposal windows, but also refine the proposals generated by existing methods. The experimental results on four challenging datasets demonstrate the effectiveness and efficiency of Adobe Boxes. For example, Adobe Boxes generally achieves 90% DR with nearly the half number of proposals required by other state-of-the-arts approaches, being close to the real-time processing.

REFERENCES

[1] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1. Jun. 2005, pp. 886–893.

[2] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2008, pp. 1–8.

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 580–587.

[4] S. Hare, A. Saffari, and P. H. S. Torr, "Struck: Structured output tracking with kernels," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Nov. 2011, pp. 263–270.

[5] Y. Zhou, X. Bai, W. Liu, and L. J. Latecki, "Similarity fusion for visual tracking," *Int. J. Comput. Vis.*, vol. 118, no. 3, pp. 337–363, Jul. 2016.

[6] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1409–1422, Jul. 2012.

[7] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. Torr, "BING: Binarized normed gradients for objectness estimation at 300fps," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 3286–3293.

[8] B. Alexe, T. Deselaers, and V. Ferrari, "What is an object?" in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 73–80.

[9] J. Feng, Y. Wei, L. Tao, C. Zhang, and J. Sun, "Salient object detection by composition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Nov. 2011, pp. 1028–1035.

[10] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *Int. J. Comput. Vis.*, vol. 104, no. 2, pp. 154–171, Apr. 2013.

[11] Z. Zhang, J. Warrell, and P. H. S. Torr, "Proposal generation for object detection using cascaded ranking SVMs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 1497–1504.

[12] C. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, vol. 8693. 2014, pp. 391–405.

[13] P. Rantalankila, J. Kannala, and E. Rahtu, "Generating object segmentation proposals using global and local search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 2417–2424.

[14] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *Int. J. Comput. Vis.*, vol. 59, no. 2, pp. 167–181, Sep. 2004.

[15] V. Cantoni, S. Levialdi, and B. Zavidovique, *3C Vision: Cues, Context and Channels*. Amsterdam, The Netherlands: Elsevier, 2011.

[16] D. Walther, L. Itti, M. Riesenhuber, T. Poggio, and C. Koch, "Attentional selection for object recognitiona gentle way," in *Proc. Biol. Motivated Comput. Vis.*, vol. 2525. 2002, pp. 472–479.

[17] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 10, pp. 1915–1926, Oct. 2012.

[18] B. Alexe, T. Deselaers, and V. Ferrari, "Measuring the objectness of image windows," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2189–2202, Nov. 2012.

[19] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. *The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results*, accessed on Jul. 2014. [Online]. Available: http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html

[20] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. *The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Results*, acessed on Sep. 2014. [Online]. Available: http://www.pascal-network.org/challenges/VOC/voc2010/workshop/index.html

[21] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. *The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results*, acessed on Sep. 2014. [Online]. Available: http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html

[22] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[23] X. Chen, H. Ma, X. Wang, and Z. Zhao, "Improving object proposals with multi-thresholding straddling expansion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 2587–2595.

[24] Q. Zhao, Z. Liu, and B. Yin, "Cracking BING and beyond," in *Proc. Brit. Mach. Vis. Conf.*, 2014, pp. 1–10.

[25] P. Dollár and C. L. Zitnick, "Structured forests for fast edge detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2013, pp. 1841–1848.

[26] P. Arbeláez, J. Pont-Tuset, J. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 328–335.

[27] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li, "Salient object detection: A discriminative regional feature integration approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 2083–2090.

[28] S. Manen, M. Guillaumin, and L. Van Gool, "Prime object proposals with randomized Prim's algorithm," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2013, pp. 2536–2543.
[29] V. Yanulevskaya, J. Uijlings, and N. Sebe, "Learning to group objects," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 3134–3141.
[30] P. Krähenbühl and V. Koltun, "Geodesic object proposals," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 725–739.
[31] R. C. Prim, "Shortest connection networks and some generalizations," *Bell Syst. Tech. J.*, vol. 36, no. 6, pp. 1389–1401, 1957.
[32] B. Epshtein and S. Ullman, "Semantic hierarchies for recognizing objects and parts," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2007, pp. 1–8.
[33] P. F. Felzenszwalb and D. P. Huttenlocher, "Pictorial structures for object recognition," *Int. J. Comput. Vis.*, vol. 61, no. 1, pp. 55–79, Jan. 2005.
[34] M. A. Fischler and R. A. Elschlager, "The representation and matching of pictorial structures," *IEEE Trans. Comput.*, vol. 22, no. 1, pp. 67–92, Jan. 1973.
[35] M. Cho, S. Kwak, C. Schmid, and J. Ponce, "Unsupervised object discovery and localization in the wild: Part-based matching with bottom-up region proposals," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1201–1210.
[36] C. Rother, V. Kolmogorov, and A. Blake, "'GrabCut': Interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, Aug. 2004.
[37] M. Tang, L. Gorelick, O. Veksler, and Y. Boykov, "GrabCut in one cut," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2013, pp. 1769–1776.

Zhiwen Fang received the B.S. and M.S. degrees from the Automation School, Beihang University. He is currently pursuing the Ph.D. degree with the Huazhong University of Science and Technology. Since 2008, he has been a Researcher with the Hunan University of Humanities, Science, and Technology. His research interests include object detection, object tracking, and machine learning.

Zhiguo Cao received the B.S. and M.S. degrees in communication and information systems from the University of Electronic Science and Technology of China, and the Ph.D. degree in pattern recognition and intelligent systems from the Huazhong University of Science and Technology. He is a Professor with the School of Automation. His research interests spread across image understanding and analysis, depth information extraction, 3D video processing, motion detection, and human action analysis. His research results, which have published dozens of papers at international journals and prominent conferences, have been applied to an automatic observation system for crop growth in agricultural, for weather phenomenon in meteorology, and for object recognition in video surveillance system based on computer vision.

Yang Xiao received the B.S., M.S., and Ph.D. degrees from the Huazhong University of Science and Technology, China. He was the Research Fellow with the School of Computer Engineering and the Institute of Media Innovation, Nanyang Technological University, Singapore. He is currently an Associate Professor with the Automation School, Huazhong University of Science and Technology, China. His research interests involve computer vision, image processing, and machine learning.

Lei Zhu received the Ph.D. degree in control science and engineering from the Huazhong University of Science and Technology. Since 2007, he has been a Researcher with the Wuhan University of Science and Technology. His main areas of interest include pattern recognition and biologically inspired vision systems.

Junsong Yuan (M'08–SM'14) received the B.Eng. degree from the Special Class for the Gifted Young, Huazhong University of Science and Technology, the M.Eng. degree from the National University of Singapore, and the Ph.D. degree from Northwestern University. He is currently an Associate Professor and the Program Director of Video Analytics with the School of Electrical and Electronics Engineering, Nanyang Technological University, Singapore.

His research interests include computer vision, video analytics, gesture and action analysis, and large-scale visual search and mining. He was the Program Co-Chair of the IEEE Conference on Visual Communications and Image Processing (2015) and the Organizing Co-Chair of the Asian Conference on Computer Vision (ACCV14), and the Area Chair of the IEEE Winter Conference on Computer Vision (2014), the IEEE Conference on Multimedia Expo (2014 and 2015), and ACCV14. He was a Guest Editor of the *International Journal of Computer Vision* and an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGIES, and *The Visual Computer journal*.

He received the Nanyang Assistant Professorship and the Tan Chin Tuan Exchange Fellowship from Nanyang Technological University, the Outstanding EECS Ph.D., the Thesis Award from Northwestern University, and the Doctoral Spotlight Award from CVPR09.