Object Proposal on RGB-D Images via Elastic Edge Boxes

Jing Liu[a,b], Tongwei Ren[a,b,*], Yuantian Wang[a,b], Sheng-Hua Zhong[c], Jia Bei[a,b], Shengchao Chen[b]

[a] *State Key Laboratory for Novel Software Technology, Nanjing University, China*
[b] *Software Institute, Nanjing University, China*
[c] *College of Computer Science and Software Engineering, Shenzhen University, China*

Abstract

As a fundamental preprocessing of various multimedia applications, object proposal aims to detect the candidate windows possibly containing arbitrary objects in images with two typical strategies, window scoring and grouping. In this paper, we first analyze the feasibility of improving object proposal performance by integrating window scoring and grouping strategies. Then, we propose a novel object proposal method for RGB-D images, named elastic edge boxes. The initial bounding boxes of candidate object regions are efficiently generated by edge boxes, and further adjusted by grouping the super-pixels within elastic range to obtain more accurate candidate windows. To validate the proposed method, we construct the largest RGB-D image data set NJU1800 for object proposal with balanced object number distribution. The experimental results show that our method can effectively and efficiently generate the candidate windows of object regions and it outperforms the state-of-the-art methods considering both accuracy and efficiency.

Keywords: elastic edge boxes, object proposal, RGB-D image
2010 MSC: 00-01, 99-00

*Corresponding author
Email address: rentw@nju.edu.cn (Tongwei Ren)

1. Introduction

Object proposal aims to detect candidate image windows possibly containing category-independent objects in a given image [1]. Compared to the renowned "dense sampling" paradigm [2], object proposal can provide content-aware candidate windows, i.e., the number of candidate windows generated by object proposal will not linearly increase with the growth of image size while retaining high coverage of image content. Obviously, it is beneficial to reduce the computational cost and difficulty of the subsequent processing. Hence, object proposal is widely used as a fundamental preprocessing of various multimedia applications, such as object detection [3, 4] and classification [5, 6], target tracking [7, 8], saliency analysis [9, 10], object recognition [11], social media mining [12, 13] and information retrieval [14].

Typically served as a preprocessing procedure, effective object proposal technique needs to satisfy the following requirements: Firstly, the proposed candidate windows should cover all or most objects in images, in order to avoid serious image content loss. Secondly, the number of candidate windows should be controlled in a limited range to reduce the computational cost of the subsequent processing. Thirdly, the candidate windows should cover the objects with high accuracy, which is usually measured by the intersection over union (IoU) of the candidate windows and the bounding boxes of objects, for high IoU is important to object detection and other applications [15]. Finally, the generation of candidate windows should be efficient, which will benefit the usage of object proposal in realtime or large-scale applications.

Though object proposal has been extensively studied in the recent years [16, 17], current methods still suffer two problems. One problem is that current object proposal methods mainly focus on the effect of color cue, which is not sufficient to a task as challenging as object proposal since it aims to extract the common properties of the objects of all categories to distinguish them from background [15]. It requires to fully explore the potentialities of different cues besides color, such as depth. In fact, depth has been used as the complement

to color in numerous object-level applications, including object segmentation [18, 19], salient object detection [20, 21], object retrieval [22, 23] and object recognition [24, 25]. Xu *et al.* firstly combined color and depth cues in object proposal [26], but they ignored the quality difference between the acquired color and depth information, i.e., depth cue usually has lower quality than color for the limitation of the existing capture devices and estimation algorithms. It means that depth cue should be dealt with in a different way to color cue when combining them together [27].

The other problem is that the existing object proposal methods usually only satisfy partial requirements of object proposal instead of all, which limits their usage in multimedia applications. Generally speaking, the typical strategies for addressing object proposal problem can be classified into two categories: window scoring [1, 28] and grouping [16, 29]. Both these two strategies focus on the former two requirements, i.e., covering as many objects in images as possible with a limited number of candidate windows, but they have distinct performance on the latter two requirements. Window scoring based methods usually have high efficiency for only requiring once scoring for each sampled box, but they are easy to fail in providing the candidate windows with high accuracy for the quantization error in sampling. In contrast, grouping based methods can generate the candidate windows with high accuracy, but they are usually time consuming for image segment merging. An interesting idea is to integrate these two strategies to obtain both high accuracy and efficiency, but the related research is still in embryonic stage and limited in RGB images [30].

In this paper, we propose a novel object proposal method, named *elastic edge boxes*, by integrating window scoring and grouping strategies and exploring both color and depth cues in RGB-D images. Figure 1 shows an overview of the proposed method. For each RGB-D image (Figure 1(a)), we first utilize window scoring strategy to identify the potential object locations with boxes according to edge cue (Figure 1(b)). Then, we represent the RGB-D image with super-pixels and select the undetermined super-pixels for each box (Figure 1(c)). Finally, we adjust the boundary of each box by applying grouping strategy on

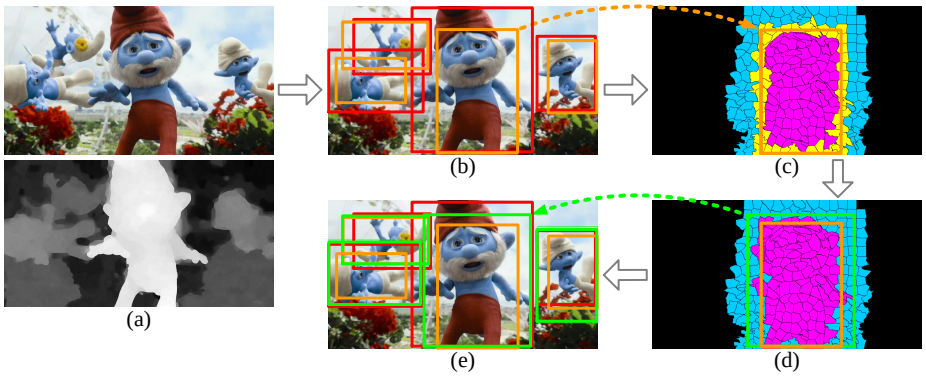

Figure 1: An overview of the proposed method. (a) RGB-D image. (b) Initial bounding boxes (orange boxes) by window scoring strategy and ground truths (red boxes). (c) Super-pixel representation. (d) Box boundary adjustment by grouping strategy. (e) Final bounding boxes (green and orange boxes) and ground truths (red boxes).

the undetermined super-pixels (Figure 1(d)) and generate the final candidate windows (Figure 1(e)). To the best of our knowledge, it is the first object proposal method integrating window scoring and grouping strategies for RGB-D images. To validate the performance of the proposed method, we construct the largest RGB-D image data set for object proposal, named *NJU1800*, on the base of stereo objectness data set [26], which provides a comprehensive and challenging benchmark for object proposal evaluation. The experimental results show that our method can generate the candidate windows with accurate locations under different accuracy, and it outperforms the existing methods considering both accuracy and efficiency. Some preliminary results of our method were presented in [31]. In this paper, we additionally analyze the feasibility of integrating window scoring and grouping strategies, and extend box boundary adjustment from unidirectional adjustment in single layer super-pixels to bidirectional adjustment in multiple layer super-pixels. Moreover, we further extend the previously proposed data set *NJU1500* to *NJU1800*, in which 300 images with one object are supplemented, and utilize it to validate the performance of our method and compare it with the state-of-the-art methods.

Our major contribution can be summarized as follows:

- We propose a novel object proposal method for RGB-D images by fully exploring the potentialities of both color and depth cues in different ways, which can outperforms the state-of-the-art methods considering both accuracy and efficiency.

- We first analyze the feasibility of integrating window scoring and grouping strategies in object proposal, and provide the upper-bound of the integrated strategies. It shows that it is possible to obtain a trade-off between accuracy and efficiency in object proposal by integrating window scoring and grouping strategies.

- We construct the largest RGB-D image data set *NJU1800* for object proposal, with balanced object number distribution and high average object number per image to provide comprehensive and challenging evaluation. It can be used as a benchmark for the future research.

The rest of the paper is organized as follows. Section 2 provides a brief review of the related work. Section 3 analyzes the feasibility of integrating window scoring and grouping strategies in object proposal. The details of the proposed method is presented in Section 4, and its performance evaluation is shown in Section 5. Finally, the paper is concluded in Section 6.

2. Related Work

The strategies of the existing object proposal methods can be roughly classified into two categories: window scoring and grouping.

Window scoring. Window scoring based methods sample a quantity of boxes in each image, score these sampled boxes based on the pre-defined features to measure the likelihood of each box containing an object, and treat the sampled boxes with high scores as candidate windows. Alexe *et al.* [1] first propose an objectness measurement based on a variety of appearance and geometry properties. Rahtu *et al.* [32] use the improved scoring algorithm of [1] on the randomly sampled boxes and the bounding boxes of single, two and

three adjacent super-pixels. Zhang *et al.* [33] apply classifiers on each scale and aspect ratio and rank the classification results to generate proposals. Cheng *et al.* [17] utilize binarized normed gradient by training a linear classifier over edge features. Zitnick *et al.* [28] use edge cue to guide window refinement, which can be specially optimized for different IoU thresholds. Xu *et al.* [26] explore the effectiveness of depth cue in handling complex scenes. Liu *et al.* [27] extend EdgeBoxes method by using depth-aware layered edges to avoid mixing the edges from the objects and background. Overall, window scoring based methods can efficiently generate the bounding boxes as proposal results, but their performance under high IoU is usually limited.

Grouping. Grouping based methods generate a number of image segments, merge the similar segments, and produce the bounding boxes of the merged segments as candidate windows. Carreira *et al.* [16] use constrained parametric mincuts in merging by several different seeds and multiple features. Humayun *et al.* [34] improve it by applying multiple graph cut segmentations and using edge detectors. Uijlings *et al.* [35] propose a typical grouping based method, selective search, which greedily merges super-pixels to generate proposals with feature similarity instead of learning. Rantalankila *et al.* [36] propose a similar merging strategy to selective search with different features in similarity measurement. Xiao *et al.* [37] extend selective search by specializing merging in high-complexity scenarios, and Wang *et al.* [38] improve it with multi-branch hierarchical segmentation. Manen *et al.* [39] use randomised super-pixel connectivity graph during merging with the learned probabilities. Long *et al.* [40] utilize bottom-up merging to generate initial object candidates, and train a supervised descent model to greedily adjust the boxes. Arbelaez *et al.* [29] perform hierarchical segmentation and multiscale combinatorial grouping with a speed-up algorithm. Krähenbühl *et al.* [41] judiciously place object-like seeds and identify critical level sets in geodesic distance transforms as object proposal results. Overall, grouping based methods can generate accurate bounding boxes as well as object boundaries, especially under high IoU, but they are usually inefficient due to bottom-up merging.

Integration of window scoring and grouping. It is interesting to integrate window scoring and grouping strategies together, for example, generating the initial candidate windows based on window scoring and further adjusting the boundaries of these candidate windows by grouping. Chen *et al.* [30] first propose this strategy for object proposal to achieve accurate bounding boxes while retaining high efficiency, but their method only focuses on RGB images and completely ignores depth cue. Our previous work [31] first applies this strategy on RGB-D images, but it is limited in one layer super-pixel extension in boundary adjustment.

3. Feasibility Analysis

A critical question is whether it is feasible to improve the performance of object proposal by integrating window scoring and grouping strategies, i.e., whether the IoUs of the candidate windows generated based on window scoring and the bounding boxes of objects will increase while adjusting the boundaries of these candidate windows by adapting to the boundaries of the related super-pixels. If so, what is the potentiality of performance improvement under the best case? For various window scoring and grouping based methods may be utilized, it is not practicable to observe the results of different window scoring based methods and further apply different boundary adjustment algorithms in the grouping based methods. Instead, we analyze the upper bounds of window scoring based methods and their boundary adjustment results to provide a general answer of the above question, and treat the difference between these two upper bounds as the potential performance improvement.

Upper bound of window scoring based methods. In different windows scoring based methods, various scoring algorithms are used to sort the sampled boxes, and the number of the retained candidate windows may be changed under different evaluation. Nevertheless, no matter which scoring algorithm is used and how many candidate windows are retained, the performance of the retained candidate windows cannot exceed the one of all the sampled boxes when ignoring

resampling of candidate windows, which are equal when the scoring algorithm is optimal and the number of the retained candidate windows is sufficient. It means the performance of all the sampled boxes can be treated as the upper bound of the performance of all the window scoring based methods initialized with the same number of sampled boxes.

Upper bound of boundary adjustment. Similarly, we use the performance of the boundary adjustment results of all the sampled boxes to represent the upper bound of the boundary adjustment results of candidate windows generated by window scoring based methods. Theoretically, to a given object, any sampled box can be adjusted by continuously adding and removing super-pixels until match the bounding box of the object as much as possible. Hence, the upper bound of the performance of boundary adjustment can be measured by comparing the bounding boxes of objects and the most similar bounding boxes of super-pixel sets to them, i.e., these bounding boxes of super-pixel sets have the highest IoU to their corresponding bounding boxes of objects.

To the bounding box b_o of a given object and the bounding box of super-pixel set b_o^* with the highest IoU to b_o, it is intuitive that their boundaries should be as close as possible. We prove that the IoU of b_o and b_o^* increases when a side of b_o^* gets close to the corresponding side of b_o^*. Yet the increasing rates from inside and outside of b_o may be not same, i.e., the IoUs of b_o and b_o^* may be not same when a side of b_o^* inside or outside b_o even with the same distance to the corresponding side of b_o. Meanwhile, the influences to IoU of four sides of b_o^* are not independent. More details can be found in Appendix. The optimal positions of b_o^*'s sides inside b_o is the boundary of the bounding box of all the super-pixels inside b_o. And the optimal position of b_o^*'s sides outside b_o is determined by the retainment of all the super-pixels on the boundary of b_o. Note here, to the super-pixels only on one side of b_o, we can simply retain the super-pixels which lead to the closest boundary of b_o^* to b_o on each side. But to the super-pixels on more than one sides of b_o, the retainment of each super-pixel will influence several sides of b_o^*, i.e., one side of b_o^* may be closer to b_o but another side of b_o^* may be farther when retaining a super-pixel. We should verify all the cases to

obtain b_o^*.

Unlimited boundary adjustment is too difficult to obtain optimal results and it completely ignores the effect of window scoring strategy. A more practicable solution is slightly adjusting the boundaries of the initial boxes generated based on window scoring, for example, only considering the retainment of super-pixels on the boundaries of these initial boxes. The performance upper bound of such "single layer" boundary adjustment can be measured by applying a similar approach as above on each initial box to each object and calculating the highest IoU among the adjusted initial boxes for each object.

Figure 2 shows the results of feasibility analysis of integrating window scoring and grouping strategies on PASCAL VOC 2007 test set, which includes 4,952 images. During the analysis, we utilize three sampling approaches, including uniform sampling, gaussian sampling and sliding window sampling. Uniform sampling and gaussian sampling sample the box center position, log aspect ratio and square root area uniformly and with Gaussian distribution, respectively. And sliding window sampling uniformly samples the boxes with different box sizes. By observing the existing object proposal methods using window scoring strategy, we choose three numbers of sampled boxes in our validation, which are 5,000, 10,000 and 20,000. In Figure 2, "sampling" and "adjustment" denote the performance before boundary adjustment and after single layer adjustment, and "unlimited adjustment" denotes the performance of unlimited adjustment. We can find that window scoring based methods cannot achieve satisfactory performance under high IoU, even assuming the scoring approach is optimal. And simply increasing the number of sampled boxes cannot obviously improve the performance of object proposal. In contrast, integrating boundary adjustment can significantly increase the recall under high IoU, even only using single layer boundary adjustment.

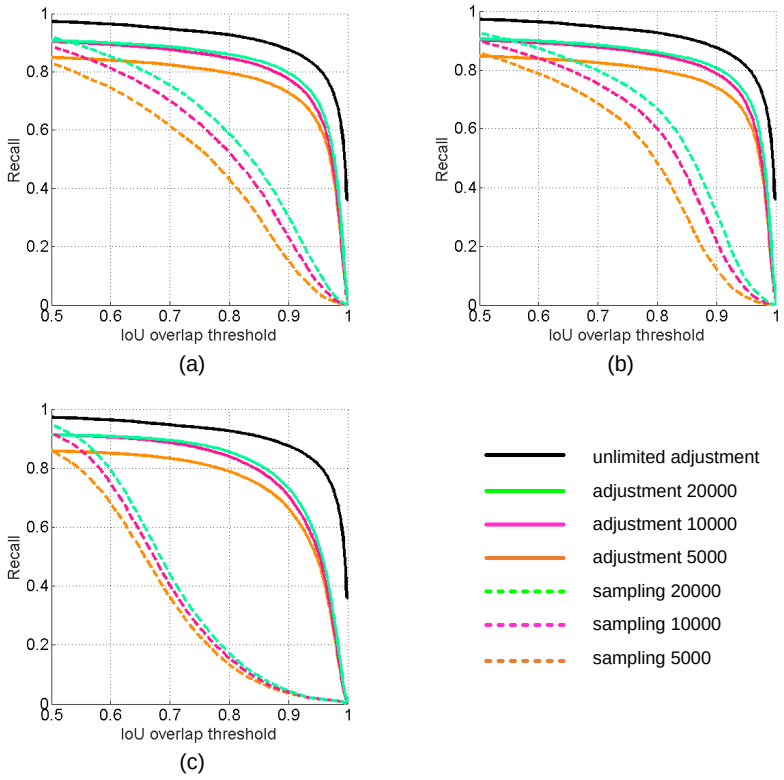

Figure 2: Feasibility analysis of integrating window scoring and grouping strategies on PASCAL VOC 2007. (a) Uniform sampling. (b) Gaussian sampling. (c) Sliding window sampling.

4. Elastic Edge Boxes

4.1. Initial box generation

We first generate the initial boxes using window scoring strategy. In the proposed approach, we specifically utilize edge boxes method [28], which can efficiently detect the approximate locations of most objects by exploiting edge cue. Edge boxes method first obtains sparse edge map through structured edge detector [42] and generates the sampled boxes with sliding window approach. Then, it scores these sample boxes according to the number of contours completely inside each box, which is highly indicative of the possibility of a

sampled box including an object. The score of a sampled box b_k^s is defined as:

$$score(b_k^s) = \frac{\sum_i \rho_k(e_i)\hat{m}_i}{2(w_k + h_k)^\eta} - \frac{\sum_{p \in b_k^{ct}} m_p}{2(w_k^{ct} + h_k^{ct})^\eta}, \quad (1)$$

where w_k and h_k are width and height of the sampled box b_k^s; b_k^{ct} is a box centered in b_k^s with the size of $w_k^{ct} \times h_k^{ct}$, which equal $w_k/2$ and $h_k/2$, respectively; $\eta = 1.5$ is a parameter to offset the bias of larger windows generally containing more edges; m_p represents the edge magnitude of each pixel and $\hat{m}_i$ is obtained by summing up edge magnitude of each pixel in the ith edge group e_i enclosed by box b_k^s; ρ_k equals zero if e_i overlaps the boundary of b_k^s. Finally, non-maximal suppression is performed to decrease the number of sampled boxes, and the pre-defined number of sample boxes with the highest scores will be selected as initial boxes.

Though its performance under high IoU is barely satisfactory, edge boxes method can achieve high recall under low IoU. It means that the initial boxes generated by edge boxes method provide the approximate locations of objects. We can adjust these initial boxes to provide more accurate candidate windows for object proposal.

4.2. Elastic range extension

Based on the initial boxes generated by edge boxes method, we further adjust their boundaries to generate the candidate windows with high accuracy. Inspired by grouping strategy, we represent the images with super-pixels [43] and utilize super-pixel as the basic operation unit in boundary adjustment, for the advantages of super-pixel in describing object boundaries, handling depth map inaccuracy, and reducing computational cost.

A key problem in boundary adjustment is to determine the *elastic range* for each initial box, i.e., the valid range for boundary adjustment. Obviously, too small elastic range will limit the adjustment and prevent from providing accurate candidate windows, while too large elastic range may cause high computational cost and reduce the effect of initial boxes.

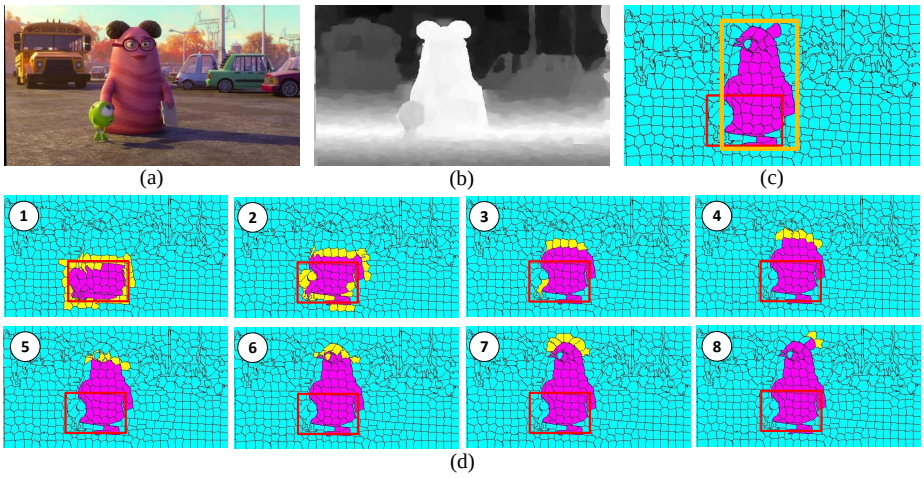

Figure 3: Example of boundary adjustment. (a) and (b) Color channel and depth channel of RGB-D image. (c) Initial box (red box) and candidate window generated after boundary adjustment (yellow box). (d) Detailed procedure of boundary adjustment, including inner super-pixels (magenta), outer super-pixels (cyan) and elastic range (yellow super-pixels).

Assume an image is represented as a set of super-pixels $S = \{s_1, \ldots, s_N\}$. Given an initial box b_k, we define $S_{in}^{b_k}$ as a set of super-pixels which are completely inside b_k (magenta ones in Figure 3(d-1)), $S_{out}^{b_k}$ as a set of super-pixels which are completely outside b_k (cyan ones in Figure 3(d-1)), and $S_e^{b_k}$ as a set of the rest super-pixels which are crossed by b_k (yellow ones in Figure 3(d-1)). In our method, $S_e^{b_k}$ is used as the initial elastic range.

As shown in Figure 3(d-2) to Figure 3(d-8), to each super-pixel newly grouped into $S_{in}^{b_k}$, we add its adjacent super-pixels in $S_{out}^{b_k}$ into the elastic range in next iteration. Note here, each super-pixel is only added into elastic range once at most. It means a super-pixel will not be added into elastic range again if it was determined to be in $S_{out}^{b_k}$.

4.3. Iterative boundary adjustment

For the number of super-pixels in $S_{in}^{b_k}$ and $S_{out}^{b_k}$ are usually unbalanced, to each super-pixel s_i in elastic range $S_e^{b_k}$, we calculate its similarities to the same number m of its nearest super-pixels in $S_{in}^{b_k}$ and $S_{out}^{b_k}$, respectively, and determine

whether it should be grouped into $S_{in}^{b_k}$. The number of the nearest super-pixels m equals 10 in our experiments to obtain a trade-off between determination accuracy and computational cost. Here, we utilize both color cue and depth cue of an RGB-D image, and define four decision parameters φ_{in}^{c}, φ_{in}^{d}, φ_{out}^{c} and φ_{out}^{d} as follows:

$$\varphi_{in}^{c} = \sum_{s_j \in \hat{S}_{in,m}^{b_k}} sim^{c}(s_i, s_j), \tag{2}$$

$$\varphi_{in}^{d} = \sum_{s_j \in \hat{S}_{in,m}^{b_k}} sim^{d}(s_i, s_j), \tag{3}$$

$$\varphi_{out}^{c} = \sum_{s_j \in \hat{S}_{out,m}^{b_k}} sim^{c}(s_i, s_j), \tag{4}$$

$$\varphi_{out}^{d} = \sum_{s_j \in \hat{S}_{out,m}^{b_k}} sim^{d}(s_i, s_j), \tag{5}$$

where $\hat{S}_{in,m}^{b_k}$ and $\hat{S}_{out,m}^{b_k}$ denote the super-pixel sets with the nearest m super-pixels to s_i in $S_{in}^{b_k}$ and $S_{out}^{b_k}$, respectively; $sim^{c}(s_i, s_j)$ and $sim^{d}(s_i, s_j)$ are calculated as follows:

$$sim^{c}(s_i, s_j) = (1 - dis^{c}(s_i, s_j))exp(-dis^{s}(s_i, s_j)), \tag{6}$$

$$sim^{d}(s_i, s_j) = (1 - dis^{d}(s_i, s_j))exp(-dis^{s}(s_i, s_j)), \tag{7}$$

where $dis^{c}(,)$ denotes the Euclidean distance of the average colors of two super-pixels in HSV space; $dis^{d}(,)$ denotes the distance of the average depth of two super-pixels; $dis^{s}(,)$ denotes the spatial distance between the centers of two super-pixels.

Based on the four parameters, we extend $S_{in}^{b_k}$ by grouping the super-pixels satisfying the following requirement:

$$S_{in}^{b_k}{}^{*} = S_{in}^{b_k} \cup \left\{ s_i \in S_{e}^{b_i} \mid (\varphi_{in}^{c} - \varphi_{out}^{c}) > t \ and \ (\varphi_{in}^{d} - \varphi_{out}^{d}) > t \right\}, \tag{8}$$

where t equals to 0.01 in our experiments.

Once $S_{in}^{b_k}$ is unchanged, we generate a bounding box $\tilde{b}_k$ of $S_{in}^{b_k}$ (yellow box in Figure 3(c)). We score the boundary adjustment results of all the initial boxes

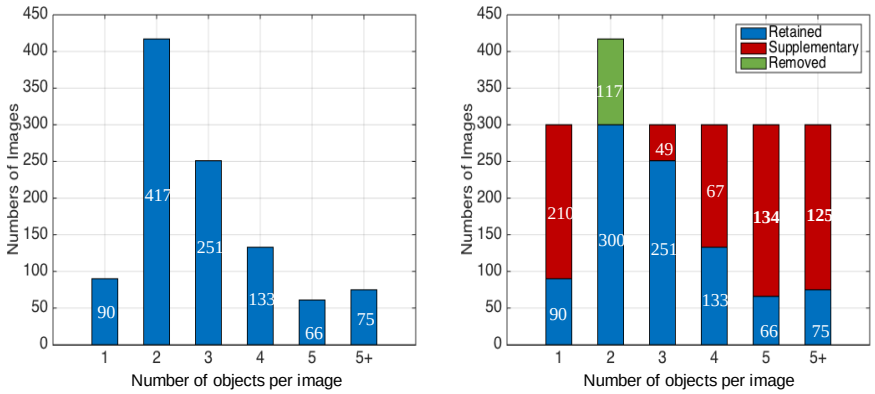

Figure 4: Distribution of object number per image. (a) Stereo objectness data set. (b) *NJU1800* data set.

with Equation (1), and sort them with the initial boxes together. Finally, we treat the boxes with the highest scores as object proposal results.

5. Experiments

5.1. Dataset construction

To validate the performance of our method, we extend the previously constructed RGB-D image data set *NJU1500* to *NJU1800*, in which 300 RGB-D images with one object are added.

Similar to *NJU1500*, *NJU1800* is constructed based on the stereo objectness data set [26]. Stereo objectness data set includes 1,032 stereo images with the content of indoor and outdoor objects in real world and virtual objects in movies. However, with the analysis of stereo objectness data set, we find that it has obvious unbalance in object number distribution. We divide the images in stereo objectness data set into six groups according to their object numbers per image, including 1, 2, 3, 4, 5, and 5+ (more than five), and count the number of images belong to each group. Figure 4(a) shows the object number distribution in stereo objectness data set, in which nearly half of the images contain no more than two objects and the average number of objects per image is only 2.98. For

the object number per image influences the performance of object proposal, i.e., more objects per image leads to higher challenge in object proposal [44], such object distribution makes the object proposal task on it less challenging.

A primary idea to solve this problem is to construct a data set by randomly sampling the real-world images. But it is difficult to construct a RGB-D image data set for object proposal in such a way. For RGB-D images are not as popular as RGB images, which are mainly from 3D movies and user capture, it is hard to ensure the high coverage of the possible image sources. Meanwhile, to avoid high repetition in image content and extremely low quality of depth information, the constructed data set usually has a limited size. It causes a data set constructed by random sampling cannot represent the real distribution of object number.

Hence, in the construction of *NJU1800* data set, we keep balance in object number distribution among images, which can provide a comprehensive benchmark for the robustness of object proposal methods to different object number per image and more challenging evaluation with high average object number per image. We remove a part of images with two objects, and supplement the images containing one object or more than two objects. The selection of the images with two objects only depends on their identifier in stereo objectness data set, and the images of large identifiers are removed. As shown in Figure 4(b), 915 images are retained from the 1,032 images of stereo objectness data set, and 885 images are supplemented. The supplemented images are collected from several 3D movies and videos, and the depth maps are calculated with Sun's optical flow method [45]. Similar to stereo objectness data set construction, we annotate the ground truths of object locations according to PASCAL VOC 2007 annotation guidelines. Five participants, including three males and two females, are invited to annotate the object bounding boxes for each supplementary image. The final constructed data set includes six groups with 300 images per group, and the average number of objects per image increases from 2.98 to 3.68. Though the average object number is slightly lower than the one on *NJU1500*, which equals 4.22, *NJU1800* provides a more

Figure 5: Examples of object proposal results generated by our method.

comprehensive benchmark for the RGB-D images with one object are popular.

5.2. Performance evaluation

We validate the performance of our method on *NJU1800* data set. All the experiments are carried out on a computer with Intel i5 2.8GHz CPU and 8GB memory. Figure 5 shows some examples of object proposal results generated by our method. The best candidate windows to each ground truth within top 2,000 ones of each image are marked with green bounding boxes. We can find that our method can effectively handle the complex situations in object proposal, such as small objects and occluded objects.

We also compare our method with the state-of-the-art object proposal methods. We firstly compare our method with the typical methods for RGB images, including binarized normed gradients (BING) [17], edge boxes (EB) [28], objectness (OBJ) [1], geodesic object proposal (GOP) [41], multiscale combi-

natorial grouping (MCG) [29], selective search (SS) [45], multi-thresholding straddling expansion of edge boxes (M-EB) and multiscale combinatorial grouping (M-MCG) [30], to validate the effectiveness of depth cue in object proposal. These compared methods with window scoring strategy, grouping strategy or integration of them can obtain excellent performance in object proposal [15]. Then, we compare our method with the object proposal methods for RGB-D images to illuminate the performance of our integration strategy. For only one typical object proposal method is proposed for RGB-D images, adaptive integration of depth and color (AIDC) [26], we extend some typical open-source object proposal methods for RGB images by combining color and depth cues, including EB [28], OBJ [1], M-EB and M-MCG [30]. We also treat AIDC [26] as an extension of BING [17] in comparison. Though attempting to individually extend each method is beyond the scope of this paper, we try to avoid too simple extension of these methods to cause unbias comparison, e.g., simply average the scores on color cue and depth cue. Without loss of generality, we select the key step for each method, such as the structured edge detection in EB, and integrate color cue and depth cue with equal weights in it. We represent the extended methods with the superscript of *, e.g., EB* denotes the extension of EB.

Accuracy. We firstly validate the recall of all the methods under different IoUs. Figure 6 and 7 show the comparison results with the methods for RGB images and RGB-D images, respectively. The comparison is carried out on the whole data set with three criteria, including the recall vs. proposal number curves when IoU equals 0.8 (Figure 6(a) and 7(a)), the average recall (AR) vs. proposal number curve [15] (Figure 6(b) and 7(b)), and the recall vs. IoU curve (Figure 6(c) and 7(c)). And Table 1 and 2 provide more details of the comparison result, in which "#prop" denotes the proposal number (i.e., the number of candidate windows), "0.8-DR" denotes the recall when IoU equals 0.8, and "AR" denotes the average recall.

In comparison with the methods for RGB images, we can find that our method outperforms all the existing methods when IoU is in range of [0.7,

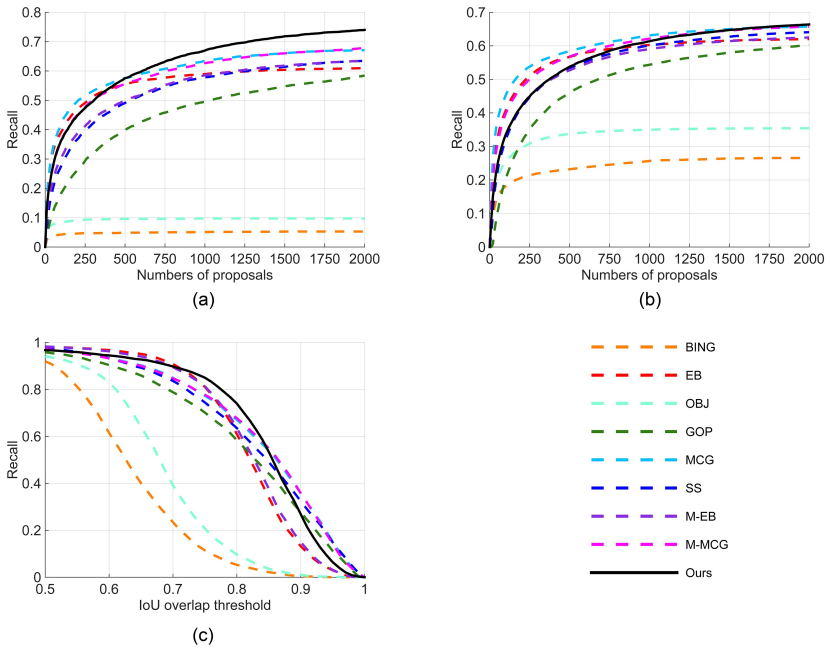

Figure 6: Comparison of our method with the methods for RGB images under different IoUs. (a) Recall vs. proposal number curves when IoU equals 0.8. (b) Average recall vs. proposal number curve. (c) Recall vs. IoU curve with 2,000 candidate windows.

Table 1: Comparison of our method and the methods for RGB images with different proposal numbers under IoU=0.8 and average IoU.

Method	Type	#prop=1000		#prop=1500		#prop=2000	
		0.8-DR	AR	0.8-DR	AR	0.8-DR	AR
BING	scoring	0.06	0.26	0.06	0.27	0.06	0.27
EB	scoring	0.57	0.60	0.59	0.61	0.60	0.62
OBJ	scoring	0.09	0.35	0.09	0.35	0.09	0.35
GOP	grouping	0.44	0.53	0.50	0.57	0.54	0.59
MCG	grouping	0.62	**0.62**	0.65	**0.64**	0.66	0.65
SS	grouping	0.56	0.59	0.60	0.62	0.62	0.63
M-MCG	grouping	0.63	0.61	0.66	0.63	0.68	0.65
M-EB	integration	0.57	0.58	0.60	0.61	0.62	0.62
Ours	integration	**0.67**	0.61	**0.72**	**0.64**	**0.74**	**0.66**

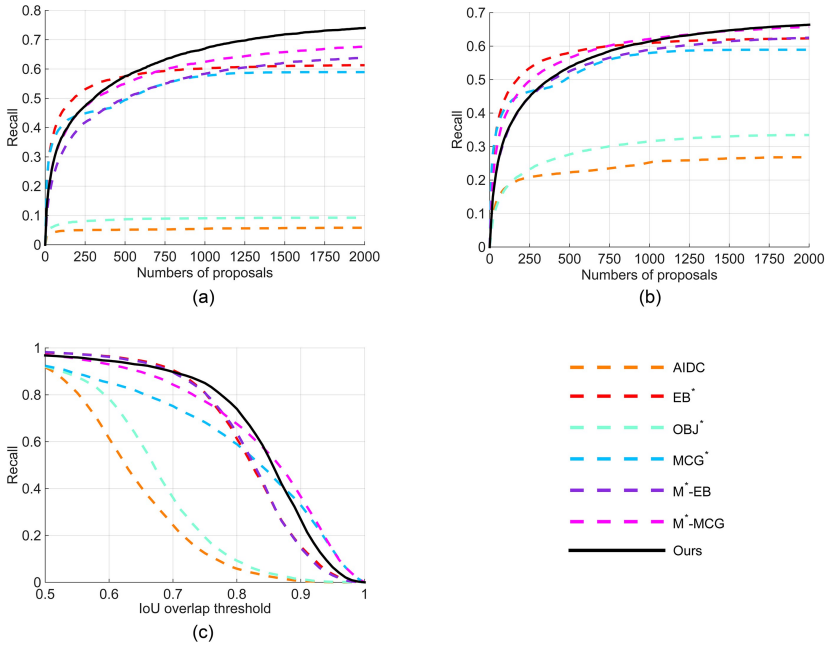

Figure 7: Comparison of our method with the methods for RGB-D images under different IoUs. (a) Recall vs. proposal number curves when IoU equals 0.8. (b) Average recall vs. proposal number curve. (c) Recall vs. IoU curve with 2,000 candidate windows.

Table 2: Comparison of our method and the methods for RGB-D images with different proposal numbers under IoU=0.8 and average IoU.

Method	Type	#prop=1000		#prop=1500		#prop=2000	
		0.8-DR	AR	0.8-DR	AR	0.8-DR	AR
AIDC	scoring	0.06	0.26	0.07	0.27	0.07	0.27
EB*	scoring	0.60	**0.61**	0.61	0.62	0.61	0.62
OBJ*	scoring	0.09	0.32	0.09	0.33	0.09	0.23
MCG*	grouping	0.58	0.58	0.59	0.59	0.59	0.59
M*-MCG	grouping	0.62	**0.61**	0.66	**0.64**	0.68	**0.66**
M*-EB	integration	0.58	0.59	0.62	0.61	0.64	0.63
Ours	integration	**0.67**	**0.61**	**0.72**	**0.64**	**0.74**	**0.66**

0.85] and slightly worse than GOP, MCG, SS, and M-MCG when IoU is larger than 0.85, which all use grouping strategy and have low efficiency. Meanwhile,

our method obtains similar performance on average recall to the best existing methods, such as MCG and M-MCG, whose running time is more than two times of ours as shown in Table 5. It shows that integrating color and depth cues is helpful to efficiently obtain high recall under high IoU requirement.

Similarly, in comparison with the methods for RGB-D images, our method outperforms all the other methods under high IoU and most other methods on average recall. By further comparing the performance of the corresponding methods for RGB images and RGB-D images in Figure 6 and 7 and Table 1 and 2, we can find that directly combining color cue and depth cue cannot obviously improve proposal performance, such as EB vs. EB*, and even cause slight decrease, such as MCG vs. MCG*, though we have tried to avoid the unbias caused by simple combination of color and depth cues in extension. It shows that the integration strategy for color and depth cues in our method is effective.

Figure 8 and 9 show some examples of object proposal results generated by our method and other methods for RGB images and RGB-D images, respectively. In these examples, red boxes indicate the ground truths, green boxes indicate the candidate windows generated by different methods, and blue boxes indicate the missed ground truths under IoU=0.8 with 2,000 candidate windows. It shows that our method can handle the images with complex structures and inconspicuous objects, while other methods may miss partial or all the objects in object proposal.

Robustness to object number. We further analyze the performance of our method and other methods on the images with different object numbers. Though the influence of object number per image to object proposal performance was mentioned in [44], it is not quantitatively evaluated. Table 3 and 4 show the comparison results of the methods for RGB images and RGB-D images on the recall under IoU=0.8 with 2,000 candidate windows, respectively. We can find that the performance of all the methods generally declines when the object number per image increases, which validates the influence of object number per image to object proposal performance. The slight volatility in the declining

Table 3: Comparison of our method and the methods for RGB images on different object numbers under IoU=0.8 with #prop=2000.

Method	#obj=1	#obj=2	#obj=3	#obj=4	#obj=5	#obj=5+
BING	0.07	0.07	0.05	0.06	0.05	0.03
EB	0.77	0.68	0.68	0.58	0.50	0.45
OBJ	0.18	0.14	0.07	0.09	0.06	0.05
GOP	0.91	0.61	0.51	0.53	0.51	0.43
MCG	0.95	0.72	0.58	0.60	0.63	**0.55**
SS	**0.97**	0.67	0.58	0.57	0.53	0.48
M-MCG	**0.97**	0.73	0.60	0.60	0.63	0.54
M-EB	0.87	0.73	0.69	0.59	0.50	0.43
Ours	0.89	**0.84**	**0.79**	**0.73**	**0.64**	**0.55**

trend may be caused by other factors influencing object proposal performance, such as the contrast of object and background. Compared to other methods, our method obtains the highest recall when the object number per image is more than one. To the methods obtain higher recall on the images with one object, including SS, M-MCG and M*-MCG, we can find their performance obviously declines 25% to 31% when the object number per image increases from one to two, which is caused by their over-emphasis of the most salient object in proposal. In contrast, the performance of our method only declines 6%, which is much lower than the ones of the above methods. It shows that our method is more robust in handling the images with different object numbers.

Speed. We also compare the efficiency of all the methods. Table 5 and 6 present the running time of our method and the methods for RGB images and RGB-D images, respectively. Though some methods require much less time than our method in processing an image, such as BING and AIDC, they obtain the worse performance. To the methods obtaining similar performance to our method, such as M-MCG and M*-MCG, they require more than double running time than our method.

The above experimental results validate the effectiveness of depth cue in

Table 4: Comparison of our method and the methods for RGB-D images on different object numbers under IoU=0.8 with #prop=2000.

Method	#obj=1	#obj=2	#obj=3	#obj=4	#obj=5	#obj=5+
AIDC	0.08	0.08	0.06	0.06	0.05	0.04
EB*	0.80	0.72	0.66	0.58	0.49	0.43
OBJ*	0.21	0.10	0.08	0.06	0.06	0.04
MCG*	0.86	0.63	0.55	0.56	0.52	0.42
M*-MCG	**0.97**	0.72	0.59	0.61	0.63	0.53
M*-EB	0.90	0.74	0.69	0.59	0.49	0.43
Ours	0.89	**0.84**	**0.79**	**0.73**	**0.64**	**0.55**

Table 5: Comparison of our method and the methods for RGB images in running time.

Method	Type	Language	Time (s)
BING	window	C++	0.06
EB	window	C++ & Matlab	0.67
OBJ	window	C++ & Matlab	4.12
GOP	grouping	C++ & Matlab	7.24
MCG	grouping	C++ & Matlab	60.09
SS	grouping	C++ & Matlab	6.42
M-MCG	grouping	C++ & Matlab	60.40
M-EB	integration	C++ & Matlab	0.98
Ours	integration	C++ & Matlab	22.34

efficiently obtaining accurate candidate windows and illustrate the superiority of our integration strategy of color and depth cues. It shows that our method can obtain balance between accuracy and efficiency in object proposal, which satisfies all the requirements of object proposal better.

6. Conclusions

In this paper, we propose an object proposal method for RGB-D images by integrating window scoring and grouping strategies. The method generates the initial boxes by an efficient edge-based window scoring method, and adjusts

Table 6: Comparison of our method and the methods for RGB-D images in running time.

Method	Type	Language	Time (s)
AIDC	window	C++	0.08
EB*	window	C++ & Matlab	0.68
OBJ*	window	C++ & Matlab	4.19
MCG*	grouping	C++ & Matlab	60.13
M*-MCG	grouping	C++ & Matlab	60.41
M*-EB	integration	C++ & Matlab	0.99
Ours	integration	C++ & Matlab	22.34

the boundaries of the initial boxes by grouping the super-pixels in elastic range, which improves proposal accuracy while retaining high efficiency. Moreover, the effectiveness of depth cue is explored as well as color cue, which is beneficial for handling the images with complex situations. The experiments show that our method can effectively and efficiently generate the candidate windows with high IoU, which outperforms state-of-the-art object proposal methods considering both accuracy and efficiency, and it is more robust to different object numbers per image.

In the future, we will focus on improving the integration of window scoring and grouping strategies in object proposal, for example, clustering the initial boxes by super-pixel representation to reduce the number of candidate windows. We will also attempt to extend our work to object proposal on video, in order to provide high accuracy proposals with acceptable efficiency.

Appendix

As shown in Figure 10, b_o (black box) is the bounding box of an object, whose four vertexes are A, B, C, and D, and b'_o (red box) is a box attempting to match b_o, whose four vertexes are A', B', C', and D'.

The IoU of b_o and b'_o can be calculated as follows:

$$IoU_{b_o b'_o} = \frac{S_{AB^*C'D^*}}{S_{ABCD} + S_{A'B'C'D'} - S_{AB^*C'D^*}}, \tag{9}$$

Figure 8: Examples of object proposal results with different methods for RGB images. (a) Original images with ground truths. (b)-(j) Object proposal results of BING [17], EB [28], OBJ [1], GOP [41], MCG [29], SS [45], M-EB [30], M-MCG [30] and our method.

where S denotes the area of the box with the corresponding four vertexes.

Considering a side of b'_o outside b_o, e.g. $A'B'$, we move it towards the corresponding side of b_o. b''_o (blue box in Figure 10(a)) is the box after the movement of side $A'B'$. The IoU of b_o and b''_o can be calculated as follows:

$$IoU_{b_o b''_o} = \frac{S_{AB^*C''D^*}}{S_{ABCD} + S_{A''B''C''D''} - S_{AB^*C''D^*}}. \tag{10}$$

For $S_{AB^*C'D^*} = S_{AB^*C''D^*}$ and $S_{A'B'C'D'} > S_{A''B''C''D''}$, we can find that $IoU_{b_o b''_o} > IoU_{b_o b'_o}$. It means the closer an outside side of b'_o to its corresponding

Figure 9: Examples of object proposal results with different methods for RGB-D images. (a) Original images with ground truths. (b)-(h) Object proposal results of AIDC [26], EB*, OBJ*, MCG*, M*-EB, M*-MCG, and our method.

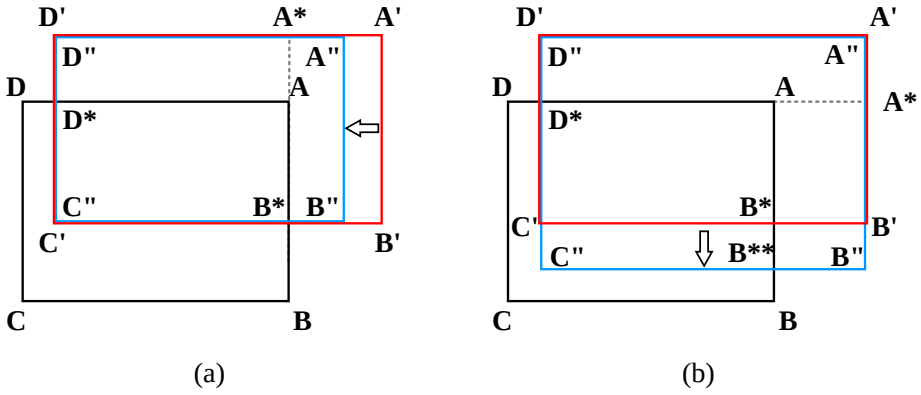

Figure 10: Influence of box side position to IoU. (a) Outside side. (b) Inside side.

side of b_o, the larger IoU is.

Similarly, considering a side of b'_o inside b_o, e.g. $B'C'$, we move it towards the corresponding side of b_o. b''_o (blue box 10(b)) is the box after the movement of side $B'C'$. The IoU of b_o and b''_o can be calculated as follows:

$$\begin{aligned} IoU_{b_o b''_o} &= \frac{S_{AB^{**}C''D^*}}{S_{ABCD} + S_{A''B''C''D''} - S_{AB^{**}C''D^*}} \\ &= \frac{S_{AB^{**}C''D^*}}{S_{ABCD} + S_{A''A^*D^*D''} + S_{A^*B''C''D^*} - S_{AB^{**}C''D^*}} \\ &= \frac{1}{\frac{S_{ABCD}+S_{A''A^*D^*D''}}{S_{AB^{**}C''D^*}} + \frac{S_{A^*B''C''D^*}}{S_{AB^{**}C''D^*}} - 1}, \end{aligned} \tag{11}$$

and we rewrite Equation (9) as follows:

$$\begin{aligned} IoU_{b_o b'_o} &= \frac{S_{AB^*C'D^*}}{S_{ABCD} + S_{A'B'C'D'} - S_{AB^*C'D^*}} \\ &= \frac{S_{AB^*C'D^*}}{S_{ABCD} + S_{A'A^*D^*D'} + S_{A^*B'C'D^*} - S_{AB^*C'D^*}} \\ &= \frac{1}{\frac{S_{ABCD}+S_{A'A^*D^*D'}}{S_{AB^*C'D^*}} + \frac{S_{A^*B'C'D^*}}{S_{AB^*C'D^*}} - 1}. \end{aligned} \tag{12}$$

For $S_{A'A^*D^*D'} = S_{A''A^*D^*D''}$, $S_{AB^*C'D^*} < S_{AB^{**}C''D^*}$ and $\frac{S_{A^*B'C'D^*}}{S_{AB^*C'D^*}} = \frac{S_{A^*B''C''D^*}}{S_{AB^{**}C''D^*}}$, we can find that $IoU_{b_o b''_o} > IoU_{b_o b'_o}$. Note here, if side $A'D'$ is inside b_o, the above proof keeps correct by simply changing the signs of the items $S_{A'A^*D^*D'}$ and $S_{A''A^*D^*D''}$ from "$+$" to "$-$". It means the closer an inside side of b'_o is to its corresponding side of b_o, the larger IoU is.

With the further analysis of Equation (9)-(12), we can find that the IoUs may be different when a side of b'_o inside and outside b_o, even its distances to the corresponding side of b_o are same. Moreover, we can find that the positions of other sides may influence IoU when considering the position of one side, i.e., different positions of other sides may cause different IoUs even the position of one side is determined. Therefore, if we attempt to obtain a box b'_o with the highest IoU with b_o, we should consider all the four sides of b'_o under both the cases of inside and outside b_o for IoU calculation.

Acknowledgments

The authors would like to thank the anonymous reviewers and the associate editor for their valuable comments, which have greatly helped us to make

improvements. This work is supported by National Science Foundation of China (61321491, 61502311, 61202320), Research Project of Excellent State Key Laboratory (61223003), Research Fund of the State Key Laboratory for Novel Software Technology at Nanjing University (ZZKT2016B09), and Collaborative Innovation Center of Novel Software Technology and Industrialization.

References

[1] B. Alexe, T. Deselaers, V. Ferrari, Measuring the objectness of image windows, IEEE Trans. Pattern Anal. Mach. Intell. 34 (2012) 2189–2202.

[2] T. Tuytelaars, Dense interest points, in: IEEE Conf. Computer Vis. Pattern Recognit., 2010, pp. 2281–2288.

[3] R. Wu, Q. Jia, H. Li, A novel stap method for the detection of fast air moving targets from high speed platform, Science China Inf. Sciences 55 (2012) 1259–1269.

[4] T. Li, H. Chang, M. Wang, B. Ni, R. Hong, S. Yan, Crowded scene analysis: A survey, IEEE Trans. Circuits Syst. Video Techn. 25 (3) (2015) 367–386.

[5] B.-K. Bao, G. Liu, C. Xu, S. Yan, Inductive robust principal component analysis, IEEE Trans. Image Proc. 21 (8) (2012) 3794–3800.

[6] Y. Lu, Z. Lai, Z. Fan, J. Cui, Q. Zhu, Manifold discriminant regression learning for image classification, Neurocomputing 166 (2015) 475–486.

[7] T. Yang, B. Li, M. Q.-H. Meng, Robust object tracking with reacquisition ability using online learned detector, IEEE Trans. Cybernetics 44 (2014) 2134–2142.

[8] T. Ren, Z. Qiu, Y. Liu, T. Yu, J. Bei, Soft-assigned bag of features for object tracking, Multimedia Syst. 21 (2) (2015) 189–205.

[9] Z. Liu, X. Zhang, S. Luo, O. L. Meur, Superpixel-based spatiotemporal saliency detection, IEEE Trans. Circuits Syst. Video Techn. 24 (9) (2014) 1522–1540.

[10] S.-H. Zhong, Y. Liu, T.-Y. Ng, Y. Liu, Perception-oriented video saliency detection via spatio-temporal attention analysis, Neurocomputing 42 (2016) 5658–5667.

[11] F. Pan, J. Wang, X. Lin, Local margin based semi-supervised discriminant embedding for visual recognition, Neurocomputing 74 (2011) 812–819.

[12] J. Tang, D. Tao, G.-J. Qi, B. Huet, Social media mining and knowledge discovery, Multimedia Syst. 20 (6) (2014) 633–634.

[13] J. Sang, C. Xu, J. Liu, User-aware image tag refinement via ternary semantic analysis, IEEE Trans. Multimedia 14 (3-2) (2012) 883–895.

[14] Y. Yang, Z.-J. Zha, Y. Gao, X. Zhu, T.-S. Chua, Exploiting web images for semantic video indexing via robust sample-specific loss, IEEE Trans. Multimedia 16 (6) (2014) 1677–1689.

[15] J. H. Hosang, R. Benenson, P. Dollár, B. Schiele, What makes for effective detection proposals?, IEEE Trans. Pattern Anal. Mach. Intell. 38 (4) (2016) 814–830.

[16] J. Carreira, C. Sminchisescu, Constrained parametric min-cuts for automatic object segmentation, in: IEEE Conf. Computer Vis. Pattern Recognit., 2010, pp. 3241–3248.

[17] M.-M. Cheng, Z. Zhang, W.-Y. Lin, P. H. S. Torr, Bing: Binarized normed gradients for objectness estimation at 300fps, in: IEEE Conf. Computer Vis. Pattern Recognit., 2014, pp. 3286–3293.

[18] R. Ju, X. Xu, Y. Yang, G. Wu, Stereo grabcut: Interactive and consistent object extraction for stereo images, in: Pac. Rim Conf. Multimedia, 2013, pp. 418–429.

[19] L. Ge, R. Ju, T. Ren, G. Wu, Interactive rgb-d image segmentation using hierarchical graph cut and geodesic distance, in: Pac. Rim Conf. Multimedia, 2015, pp. 114–124.

[20] R. Ju, Y. Liu, T. Ren, L. Ge, G. Wu, Depth-aware salient object detection using anisotropic center-surround difference, Sig. Proc.: Image Comm. 38 (2015) 115–126.

[21] J. Guo, T. Ren, J. Bei, Salient object detection for rgb-d image via saliency evolution.

[22] X. Xu, W. Geng, R. Ju, Y. Yang, T. Ren, G. Wu, Obsir: Object-based stereo image retrieval, in: IEEE Int. Conf. Multimedia Expo, 2014, pp. 1–6.

[23] A. Petrelli, D. Pau, L. Di Stefano, Analysis of compact features for rgb-d visual search, in: Int. Conf. Image Anal. Proc., 2015, pp. 14–24.

[24] M. Blum, J. T. Springenberg, J. Wülfing, M. Riedmiller, A learned feature descriptor for object recognition in rgb-d data, in: IEEE Int. Conf. Robotics Automation, 2012, pp. 1298–1303.

[25] L. Bo, X. Ren, D. Fox, Unsupervised feature learning for rgb-d based object recognition, in: Int. Symp. Experimental Robotics, 2013, pp. 387–402.

[26] X. Xu, L. Ge, T. Ren, G. Wu, Adaptive integration of depth and color for objectnes estimation, in: IEEE Int. Conf. Multimedia Expo, 2015, pp. 1–6.

[27] J. Liu, T. Ren, B.-K. Bao, J. Bei, Depth-aware layered edge for object proposal, in: IEEE Int. Conf. Multimedia Expo, 2016, pp. 1–6.

[28] C. L. Zitnick, P. Dollár, Edge boxes: Locating object proposals from edges, in: European Computer Vis., 2014, pp. 391–405.

[29] P. Arbeláez, J. Pont-Tuset, J. T. Barron, F. Marques, J. Malik, Multiscale combinatorial grouping, in: IEEE Conf. Computer Vis. Pattern Recognit., 2014, pp. 328–335.

[30] X. Chen, H. Ma, X. Wang, Z. Zhao, Improving object proposals with multi-thresholding straddling expansion, in: IEEE Conf. Computer Vis. Pattern Recognit., 2015, pp. 2587–2595.

[31] J. Liu, T. Ren, J. Bei, Elastic edge boxes for object proposal on rgb-d images, in: Int. Conf. Multimedia Modeling, 2016, pp. 199–211.

[32] E. Rahtu, J. Kannala, M. B. Blaschko, Learning a category independent object detection cascade, in: IEEE Int. Conf. Computer Vis., 2011, pp. 1052–1059.

[33] Z. Zhang, J. Warrell, P. H. S. Torr, Proposal generation for object detection using cascaded ranking svms, in: IEEE Conf. Computer Vis. Pattern Recognit., 2011, pp. 1497–1504.

[34] A. Humayun, F. Li, J. M. Rehg, Rigor: Reusing inference in graph cuts for generating object regions, in: IEEE Conf. Computer Vis. Pattern Recognit., 2014, pp. 336–343.

[35] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, A. W. M. Smeulders, Selective search for object recognit., Int. J. Computer Vis. 104 (2) (2013) 154–171.

[36] P. Rantalankila, J. Kannala, E. Rahtu, Generating object segmentation proposals using global and local search, in: IEEE Conf. Computer Vis. Pattern Recognit., 2014, pp. 2417–2424.

[37] Y. Xiao, C. Lu, E. Tsougenis, Y. Lu, C.-K. Tang, Complexity-adaptive distance metric for object proposals generation, in: IEEE Conf. Computer Vis. Pattern Recognit., 2015, pp. 778–786.

[38] C. Wang, L. Zhao, S. Liang, L. Zhang, J. Jia, Y. Wei, Object proposal by multi-branch hierarchical segmentation, in: IEEE Conf. Computer Vis. Pattern Recognit., 2015, pp. 3873–3881.

[39] S. Manen, M. Guillaumin, L. J. V. Gool, Prime object proposals with randomized prim's algorithm, in: IEEE Int. Conf. Computer Vis., 2013, pp. 2536–2543.

[40] C. Long, X. Wang, G. Hua, M. Yang, Y. Lin, Accurate object detection with location relaxation and regionlets re-localization, in: Asian Conf. Computer Vis., 2014, pp. 260–275.

[41] P. Krähenbühl, V. Koltun, Geodesic object proposals, in: European Computer Vis., 2014, pp. 725–739.

[42] P. Dollár, C. L. Zitnick, Structured forests for fast edge detection, in: IEEE Int. Conf. Computer Vis., 2013, pp. 1841–1848.

[43] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, S. Süsstrunk, Slic superpixels compared to state-of-the-art superpixel methods, IEEE Trans. Pattern Anal. Mach. Intell. 34 (11) (2012) 2274–2282.

[44] J. Zhang, S. Ma, M. Sameki, S. Sclaroff, M. Betke, Z. Lin, X. Shen, B. Price, R. Mech, Salient object subitizing, in: IEEE Conf. Computer Vis. Pattern Recognit., 2015, pp. 4045–4054.

[45] D. Sun, S. Roth, M. J. Black, Secrets of optical flow estimation and their principles, in: IEEE Conf. Computer Vis. Pattern Recognit., 2010, pp. 2432–2439.