Noname manuscript No.
(will be inserted by the editor)

Insights of Object Proposal Evaluation

Yuantian Wang · Lei Huang · Tongwei Ren ✉ · Sheng-Hua Zhong · Han Gu · Yan Liu

Received: date / Accepted: date

Abstract Object proposal aims to locate category-independent objects in a given image with a limited number of object candidates indicated by bounding boxes, which can be served as a fundamental of various multimedia applications. Current evaluation criteria based on recall cannot reveal the real abilities of different object proposal methods in objectness measurement. In this paper, we propose a novel object proposal evaluation criterion instead of recall, named *objectness measurement ability* (OMA). We first analyze the probability to hit an object by non-repetitive random sampling (HPRS), and provide an algorithm for calculating HPRS efficiently. Based on HPRS, we define OMA and extend three commonly used object proposal evaluation criteria by replacing recall with OMA. We evaluated six typical object proposal methods using recall based criteria and OMA based criteria on the test data of PASCAL VOC 2007 and PASCAL VOC 2012. The experimental results show

Yuantian Wang
State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China
E-mail: wangyt@smail.nju.edu.cn

Lei Huang
State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China
E-mail: leihuang@nju.edu.cn

Tongwei Ren
State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China
E-mail: rentw@nju.edu.cn

Sheng-Hua Zhong
College of Computer Science and Software Engineering, Shenzhen University, China
E-mail: csshzhong@szu.edu.cn

Han Gu
State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China
E-mail: 141250039@smail.nju.edu.cn

Yan Liu
Computing Department, The Hong Kong Polytechnic University, Hong Kong, China
E-mail: csyliu@comp.polyu.edu.hk

that OMA based criteria can provide more stable evaluation results than recall based ones in revealing objectness measurement ability.

Keywords Object proposal evaluation · objectness measurement ability · hit probability of random sampling

1 Introduction

Object proposal aims to locate category-independent objects in a given image with a limited number of object candidates indicated by bounding boxes [1]. Since the number of object candidates generated by object proposal is much less than the number of those generated by sliding window [2], object proposal can be served as a fundamental of many multimedia applications, such as object classification [3–6], detection [7–11], segmentation [12,13], retrieval [14–17], tracking [18–20], action recognition [21–23] and image annotation [24–27]. The research on object proposal is based on a consensus that all objects belonging to different categories share some common properties to differ from background, which is named *objectness*. High ability in objectness measurement is crucial to an effective object proposal method. Two paradigms are mainly used in current object proposal methods: window scoring and grouping [28,29]. Window scoring based methods first sample a lot of bounding boxes in a given image, and select the ones with the highest objectness scores as object candidates [30, 31]; while grouping based methods over-segment an image into amounts of segments, group these segments with objectness ranking and use the bounding boxes of the grouping results as object candidates [32,33].

Figure 1 shows an example of object proposal, in which red bounding boxes denote manually labelled objects in ground truth, green and blue bounding boxes denote object candidates generated by object proposal. In object proposal evaluation, intersection over union (IoU) is used to judge whether an object in ground truth is located accurately by an object candidate. If their IoU is larger than a predefined threshold, we consider that the candidate locates the object successfully, named *hit*; otherwise, we consider that the candidate fails in locating the object, named *miss*. As shown in Fig. 1, the candidates denoted by green bounding boxes hit the corresponding objects, while the blue ones miss all the objects.

To compare the performance of different methods, object proposal evaluation is conducted on datasets consisted of numerous images with various objects. Each object proposal method is allowed to provide a certain number of candidates on each given image. *Recall* on an image is calculated as the proportion of hit objects under the predefined IoU in all manually labelled objects in ground truth on the image, and the mean value of recall values on all images is treated as a criterion in object proposal evaluation. Obviously, recall is influenced by the values of the predefined candidate number and IoU. More candidates and lower IoU benefit to obtain higher recall. Hence, recall is usually evaluated versus candidate number or IoU. Figure 2 shows three commonly used criteria in current object proposal evaluation, *i.e.*, recall *vs.*

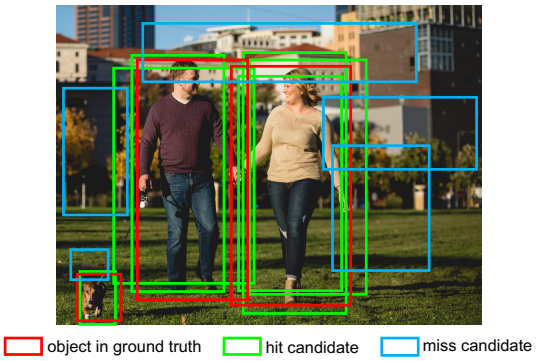
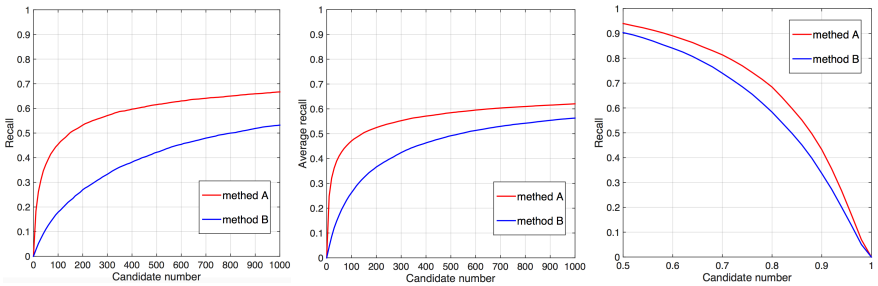


Fig. 1: An example of object proposal. Red, green and blue bounding boxes denote manually labelled objects in ground truth, object candidates hit one object and object candidates miss all the objects, respectively.



(a) (b) (c)

Fig. 2: Examples of commonly used criteria in object proposal evaluation. (a) Recall *vs.* candidate number under a certain IoU. (b) Average recall *vs.* candidate number. (c) Recall *vs.* IoU under a certain candidate number. In all the examples, method A (red) outperforms method B (blue).

candidate number under a certain IoU, average recall *vs.* candidate number, and recall *vs.* IoU under a certain candidate number. The details of these criteria will be presented in Section 2.

Although these recall based criteria can evaluate the effect of object proposal methods in the way that how many objects are hit by the provided candidates, they are easily influenced by some irrelevant factors to objectness, such as candidate number. A simple fact is that every object in a given image can be hit even with non-repetitive random sampling if sufficient candidates are provided, because the number of possible bounding boxes in an image is finite. Besides, other irrelevant factors to objectness may also influence the evaluation results on recall based criteria, including object position and size. Institutively, an object with larger size and near to image center is easier to hit than that with small size and at image corner. It means that an object proposal method may obtain quite unstable performance on recall based criteria, even only the sizes or positions of objects are changed. The evaluation results in simple situations, large object sizes and sufficient

candidate numbers may mislead the readers about the effects of the evaluated object proposal methods. An extreme circumstance is that some methods may obtain acceptable performances on recall based criteria even they are worse than random sampling, *i.e.*, they adopt incorrect objectness measurement strategies. Several examples are shown in Section 5. Hence, we can only compare the relative effects of different methods using recall based criteria, but not reveal their real performance on objectness measurement stably. It works against the exploration of better features of objectness measurement and improves the understanding of common properties of various objects.

To tackle the above problem, we propose a novel object proposal evaluation criterion instead of recall, named *objectness measurement ability* (OMA) . We first analyze the probability to hit an object by non-repetitive random sampling (HPRS), and provide an algorithm of efficient hit candidates calculation for HPRS. Next, we define OMA based on HPRS, and extend three commonly used object proposal evaluation criteria by replacing recall with OMA. Finally, we compare the evaluation results of several typical object proposal methods using current recall based criteria and our OMA based criteria on the test data of PASCAL VOC 2007 and PASCAL VOC 2012. The experimental results show that our proposed criteria can provide more stable evaluation results to reveal objectness measurement abilities of different methods effectively.

Our contributions mainly include:

- We analyze HPRS in object proposal for OMA definition, and present an efficient algorithm for hit candidates counting in HPRS calculation.
- We propose a new OMA criterion instead of recall in object proposal evaluation, and extend current commonly used criteria based on OMA.
- We validate our proposed criteria by evaluating state-of-the-art object proposal methods on two datasets, which are superior to current criteria in evaluating objectness measurement abilities of different methods.

2 Object proposal evaluation criteria based on recall

IoU is a criterion used to determine whether an object candidate hits or misses an object in ground truth in object proposal evaluation. It is calculated as the ratio of the area of their intersection to that of their union:

$$IoU = \frac{S_c \cap S_o}{S_c \cup S_o} = \frac{S_I}{S_c + S_o - S_I}, \tag{1}$$

where S_c and S_o denote the areas of the candidate and the object, respectively; S_I denotes the area of their intersection. If the IoU is not less than a predefined threshold τ, we consider that the candidate hits the object; otherwise, we consider that the candidate misses the object.

By counting the ratio of hit objects to all objects in an image, we can calculate the recall on the image. The mean value of recall values on all the

images in an evaluation dataset is used to represent the performance of object proposal on this dataset:

$$recall = \frac{1}{N_{img}} \sum_{i=1}^{N_{img}} \frac{|H_i|}{|O_i|}, \tag{2}$$

where H_i and O_i are the sets of hit objects and all objects on the ith image, respectively; N_{img} is the number of images in the evaluation dataset; $|.|$ denotes the set cardinality.

To a specific object proposal method, recall usually increases along with candidate number under a certain IoU, *i.e.*, more candidates proposed on each image helps to achieve higher recall. Hence, the relationship between recall and candidate number under a certain IoU is used to illustrate object proposal performance. Figure 2(a) shows an example of recall *vs.* candidate number criterion, in which method A outperforms method B because method A achieves higher recall than method B under the same candidate number and requires less candidates to achieve the same recall.

To evaluate the performance under different IoUs comprehensively, average recall (AR) is proposed by averaging the recall values under the IoUs between 0.5 and 1 [28]:

$$\begin{aligned} AR &= 2 \int_{0.5}^{1} recall(\varphi) \mathrm{d}\varphi \\ &= \frac{1}{N_{IoU}} \sum_{k=1}^{N_{IoU}} recall\left(0.5 + \frac{0.5k}{N_{IoU}}\right), \end{aligned} \tag{3}$$

where $recall(\varphi)$ denotes the recall value under IoU equals φ; N_{IoU} is the number of IoUs, which divides the value range [0.5, 1] into N_{IoU} uniform intervals. Figure 2(b) shows an example of average recall *vs.* candidate number. Similar to Fig. 2(a), method A outperforms method B because method A achieves higher average recall than method B under the same candidate number and requires less candidates to achieve the same average recall.

The relationship between recall and IoU under a certain candidate number is also used to illustrate the performance of different methods in object proposal evaluation. As IoU stands for the accuracy requirement in object localization, recall usually decreases when IoU increases. Figure 2(c) shows an example of recall *vs.* IoU. We can see that method A outperforms method B, because method A achieves higher recall than method B under the same IoU.

The above three criteria are commonly used in object proposal evaluation [28,30,31,34]. Because the curve of one method may be not always above that of another, some additional criteria are used to compare the performance of different methods, such as area under the curve [35]. However, all these criteria are based on recall, which prevents them from evaluating different methods on their real abilities of objectness measurement.

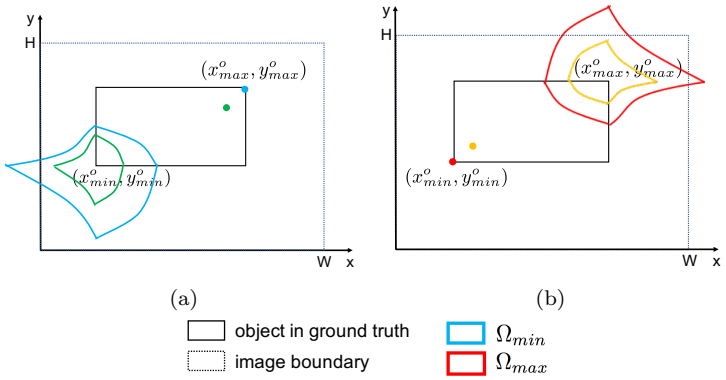


Fig. 3: An example of the coordinate range of bottom-left corner Ω_{min} (denoted by the region with blue boundary) and that of top-right corner Ω_{max} (denoted by the region with red boundary) of all hit candidates to object o. (a) and (b) show that Ω_{min} and Ω_{max} are obtained when the top-right corner and the bottom-left corner of candidate c have the same coordinates to those of object o, respectively. The regions with boundaries in different colors denote the corresponding coordinate sets of all possible bottom-left (or top-right) corners of c when the top-right (or bottom-left) corner of c is in the points denoted in the same colors.

3 Hit probability of non-repetitive random sampling

To evaluate OMA of an object object proposal method, we first calculate HPRS without considering objectness measurement and treat it as the baseline in object proposal evaluation. Assume W and H are the width and the height of a given image, o is an object in the image with the bottom-left corner (x^o_{min}, y^o_{min}) and the top-right corner (x^o_{max}, y^o_{max}), and w and h are the width and the height of object o, which equal to $x^o_{max} - x^o_{min}$ and $y^o_{max} - y^o_{min}$, respectively (see Fig. 3(a)). We count the numbers of possible candidates by non-repetitive random sampling and those hitting o, and calculate the HPRS of o based on them.

3.1 Number of Possible Candidates

Each object candidate can be identified by the coordinates of its bottom-left corner (x^c_{min}, y^c_{min}) and its top-right corner (x^c_{max}, y^c_{max}), satisfying $1 \leq x^c_{min} < x^c_{max} \leq W$ and $1 \leq y^c_{min} < y^c_{max} \leq H$. Hence, the total number of possible candidates N_{tol} can be calculated as:

$$N_{tol} = \mathrm{C}^2_W * \mathrm{C}^2_H = \frac{1}{4} W(W-1)H(H-1), \tag{4}$$

where C^2_W and C^2_H denote the combinations of (x^c_{min}, x^c_{max}) and (y^c_{min}, y^c_{max}), respectively.

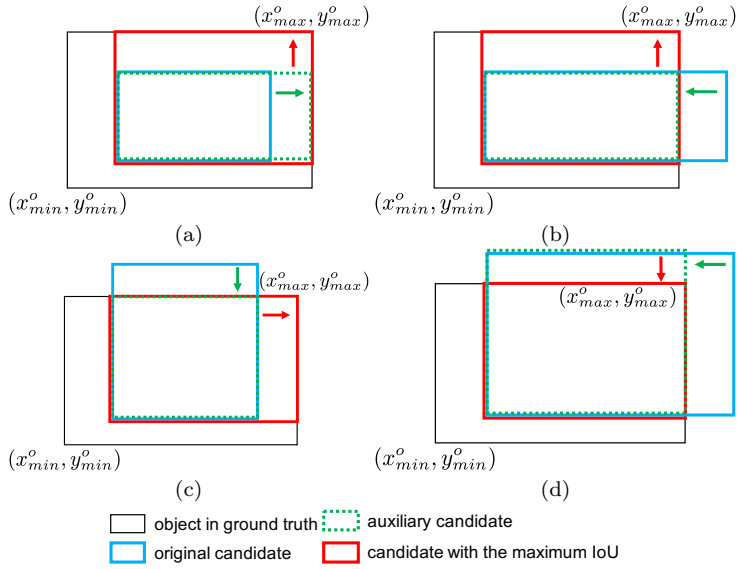


Fig. 4: Examples of enlarging IoU between candidate c with the given bottom-left corner and object o by moving the top-left corner of c to the position of the top-left corner of o. Blue box denotes the original candidate, red box denotes the candidate with the maximum IoU after moving its top-left corner to the position of the top-left corner of o, and green box denotes auxiliary candidate for understanding.

3.2 Number of Hit Candidates

A simple solution for counting the number of candidates hitting object o is to traverse all possible candidates and judge whether their IoU to o are larger than the predefined threshold. However, it is time consuming because the number of possible candidates is much larger than that of the hit ones. Here, we propose a new algorithm for calculating the number of hit candidates efficiently.

Since each candidate is identified by the coordinates of its bottom-left corner and top-right corner, we first estimate the coordinate range of bottom-left corner of all hit candidates. To a given coordinate of its bottom-left corner, candidate c has the largest IoU if its top-right corner has the same coordinate to that of object o. Figure 4 shows the examples when the bottom-left corner of c is inside o and the top-right corner has different positions. We can see that the IoU between c and o is enlarged in all situations when moving the top-left corner of c to that of o, *i.e.*, changing the blue box to the red one (green box denotes a auxiliary one for understanding). The same conclusion can be obtained in the cases of different bottom-left corner's positions of c. In other words, to a bottom-left corner, all candidates identified by it and any top-left corner will miss o if the one identified by it and the top-right corner of o has smaller IoU than the predefined threshold. Hence, we can estimate the coordinate range of bottom-left corner of all hit candidates as the set of all valid bottom-left corner's coordinates when the top-right corner of c has the same coordinate to that of object o. Figure 3(a) shows an example of the

coordinate range of bottom-left corner of all hit candidates (denoted by the region with blue boundary). Based on Eq. (1), we can represent the coordinate range of bottom-left corner of all hit candidates Ω_{min} as follows:

$$\begin{aligned}
\Omega_{min} =& \Omega^1_{min} \cup \Omega^2_{min}, \\
\Omega^1_{min} =& \Big\{ x^o_{min} - w(\frac{1}{\tau} - 1) \leq x^c_{min} \leq x^o_{min}, \\
& y^o_{min} + h - \frac{wh}{\tau(w - x^c_{min})} \leq y^c_{min} \\
& \leq y^o_{min} + h - \frac{wh}{x^c_{min} + \frac{w}{\tau}} \Big\}, \\
\Omega^2_{min} =& \Big\{ x^o_{min} \leq x^c_{min} \leq x^o_{min} + w(1 - \tau), \\
& y^o_{min} + \frac{wh}{w - x^c_{min}} - \frac{h}{\tau} \leq y^c_{min} \\
& \leq y^o_{min} + h - \frac{\tau wh}{w - x^c_{min}} \Big\},
\end{aligned} \tag{5}$$

where w and h are the width and height of object o, respectively; τ is the predefined IoU.

To each coordinate in Ω_{min}, we estimate its corresponding coordinate range of top-right corner of all hit candidates. Similarly, the coordinate range of the top-right corner of all hit candidates Ω_{max} can be estimated as the set of all valid top-right corner's coordinates when the bottom-left corner of c has the same coordinate to that of object o. Figure 3(b) shows an example of the coordinate range of top-right corner of all hit candidates (denoted by the region with red boundary). Based on Eq. (1) and (5), we can calculate Ω_{max} as follows:

$$\begin{aligned}
\Omega_{max} =& \Omega^1_{max} \cup \Omega^2_{max}, \\
\Omega^1_{max} =& \Big\{ x^o_{max} - w(1 - \tau) \leq x^c_{max} \leq x^o_{max}, \\
& y^o_{max} - h - \frac{\tau wh}{x^o_{max} - x^c_{max} - w} \leq y^c_{max} \\
& \leq y^o_{max} + \frac{h}{\tau} + \frac{wh}{x^o_{max} - x^c_{max} - w} \Big\}, \\
\Omega^2_{max} =& \Big\{ x^o_{max} \leq x^c_{max} \leq x^o_{max} + w(\frac{1}{\tau} - 1), \\
& y^o_{max} + \frac{wh}{x^o_{max} - x^c_{max} + \frac{w}{\tau}} - h \leq y^c_{max} \\
& \leq y^o_{max} - \frac{wh}{\tau(x^o_{max} - x^c_{max} - w)} - h \Big\},
\end{aligned} \tag{6}$$

where w, h and τ are defined as Eq. (5).

Algorithm 1 Hit Candidates Counting Algorithm

Input: image width W and height H, predefined IoU τ, object o with bottom-left corner (x^o_{min}, y^o_{min}) and top-right corner (x^o_{max}, y^o_{max})
Output: number of hit candidates N_{hit}
Initialize: $N_{hit} = 0$
Compute Ω_{min} using Eq. (5)
Compute Ω_{max} using Eq. (6)
for $(x^c_{min}, y^c_{min}) \in \Omega_{min}$ AND $x^c_{min} \geq 1$ AND $y^c_{min} \geq 1$ **do**
  for $(x^c_{max}, y^c_{max}) \in \Omega_{max}$ AND $x^c_{max} \leq W$ AND $y^c_{max} \leq H$ **do**
    Generate a candidate c with $(x^c_{min}, y^c_{min}, x^c_{max}, y^c_{max})$
    Compute IoU between c and o using Eq. (1)
    if $IoU \geq \tau$ **then**
      $N_{hit} = N_{hit} + 1$
    end if
  end for
end for

Based on Ω_{min} and Ω_{max}, we can count the number of hit candidates to object o. Algorithm 1 presents the pseudocodes, in which we consider the situations that Ω_{min} and Ω_{max} are partly beyond image boundary.

3.3 HPRS Calculation

Based on the numbers of possible candidates in a given image and hit candidates to object o, we calculate the HPRS of o as follows:

$$HPRS(o,k) = 1 - \frac{\mathrm{C}^k_{N_{tol}-N_{hit}}}{\mathrm{C}^k_{N_{tol}}}, \tag{7}$$

where k is the number of candidates generated by non-repetitive random sampling; $HPRS(o,k)$ denotes the hit probability of o with k randomly sampled candidates, *i.e.*, the probability that o is hit at least once by k randomly sampled candidates; N_{tol} and N_{hit} are the numbers of possible candidates in the image and hit candidates to o, which are calculated by Eq. (4) and Algorithm 1, respectively; C denotes combination operation.

4 Object proposal evaluation criteria based on OMA

We define the OMA of an object proposal method on a given dataset by removing its HPRS from its recall. Based on Eq. (2) and (7), we calculate OMA as follows:

$$OMA = \frac{1}{N_{img}} \sum_{i=1}^{N_{img}} \frac{1}{|O_i|} \left(|H_i| - \sum_{i=1}^{|O_i|} HPRS\left(o^i_j, k\right) \right), \tag{8}$$

where N_{img} is the number of images in the evaluation dataset; H_i and O_i are the sets of hit objects and all objects on the ith image, respectively; o^i_j

is the jth object in O_i; k denotes the number of candidates provided on each image; $|.|$ denotes the set cardinality. Based on OMA defined in Eq. (8), we can extend the criteria of recall *vs.* candidate number, recall *vs.* IoU to OMA *vs.* candidate number and OMA *vs.* IoU, respectively.

To evaluate the OMA values under different IoUs comprehensively, we define *average OMA* (AO) by referring average recall in Eq. (3):

$$\begin{aligned} AO &= 2\int_{0.5}^{1} OMA(\varphi)\mathrm{d}\varphi \\ &= \frac{1}{N_{IoU}}\sum_{k=1}^{N_{IoU}} OMA\left(0.5+\frac{0.5k}{N_{IoU}}\right), \end{aligned} \tag{9}$$

where $OMA(\varphi)$ denotes the OMA value under IoU equals φ; N_{IoU} is the number of IoUs, which divides the value range [0.5, 1] into N_{IoU} uniform intervals. Based on Eq. (9), we can extend the criterion of average recall *vs.* candidate number to average OMA *vs.* candidate number.

5 Experiments

5.1 Datasets and Experiment Settings

We validated the proposed criteria on two datasets: PASCAL VOC 2007 test data (hereinafter referred to as "VOC 2007") [36] and PASCAL VOC 2012 test data (hereinafter referred to as "VOC 2012") [37]. VOC 2007 contains 4,952 images annotated with 20 object classes. The total number of annotated objects is 16,488 and 3.33 objects are on each image in average. VOC 2012 contains 16,135 images, in which 5,138 images are annotated with the same 20 object classes to VOC 2007. The total number of annotated objects is 7,330 and 1.43 objects are on each image in average.

All the experiments were conducted on a computer with 2.9GHz Intel Core i5 CPU and 8GB memory. We apply the default settings of author suggestions for all the object proposal methods in our experiments.

5.2 Efficiency of Hit Candidates Calculation

We validate the efficiency of Algorithm 1 in calculating the number of hit candidates. To illustrate our performance, we use a baseline *Exhaustion* for comparison, which traverses all the N_{tol} possible candidates in Eq. (4). Table 1 shows the time costs of hit candidates calculation using Exhaustion and Algorithm 1 on VOC 2007 and VOC 2012. We can see that Algorithm 1 reduces over 95% time cost of hit candidates calculation than Exhaustion. Moreover, the efficiency improvement of Algorithm 1 is more obvious when requiring higher IoU, because both Ω_{min} and Ω_{max} are smaller when IoU is higher.

Table 1: Efficiency comparison between Algorithm 1 and Exhaustion on VOC 2007 and VOC 2012.

Time per object (s)	Exhaustion	Algorithm 1
VOC 2007, IoU=0.5	48.17	2.36
VOC 2007, IoU=0.8	48.17	1.95
VOC 2012, IoU=0.5	47.53	2.31
VOC 2012, IoU=0.8	47.53	1.94

5.3 Criteria Comparison on VOC 2007

To illustrate the advantages of our proposed criteria, we evaluate seven typical object proposal methods using current recall based criteria and our OMA based criteria on VOC 2007. The evaluated methods include edgeboxes (EB) [30], geodesic object proposals (GOP) [38], multiscale combinatorial grouping (MCG) [34], multi-thresholding straddling expansion of edge boxes (M-EB) and multiscale combinatorial grouping (M-MCG) [39], objectness (OBJ) [1] and selective search (SS) [40].

Recall *vs.* candidate number and OMA *vs.* candidate number. Figure 5 shows the evaluation results of different methods using recall *vs.* candidate number and OMA *vs.* candidate number under IoUs equal 0.5 and 0.8, respectively. We have:

1) Most methods perform more stable in OMA than recall against candidate number increase after the top 100 candidates. It means that OMA eliminates the probability of random hit effectively and reveals the real abilities of different methods in objectness measurement.

2) It shows that OBJ method seems to obtain acceptable recall under IoU equals 0.5 in Fig. 5(a), though it underperforms other methods. However, we can see that its performance is worse than that of random sampling on the top 1,000 candidates, which is denoted with a black dotted line (OMA equals 0) in Fig. 5(b). It means OBJ method adopts incorrect objectness measurement strategy. Note here, we cannot conclude that all the object candidates generated by OBJ methods are incorrect. In fact, we can see that OBJ method slightly outperforms random sampling under IoU equals 0.8 on the top 500 candidates from Fig. 5(d). It means that OBJ method can locate several objects accurately on these candidates, but random sampling hits more objects under IoU equals 0.5.

Average recall *vs.* candidate number and average OMA *vs.* candidate number. Figure 6 shows the evaluation results of different methods using average recall *vs.* candidate number and average OMA *vs.* candidate number, respectively. Similarly, we can see that average OMA performs more stable than average recall. It validate the effectiveness of average OMA in evaluating objectness measurement ability.

Recall *vs.* IoU and OMA *vs.* IoU. Figure 7 shows the evaluation results of different methods using recall *vs.* IoU and OMA *vs.* IoU on the top 1,000 candidates, respectively. Similar to Fig. 5, OBJ method seems to obtain acceptable recall on the top 1,000 candidates, but its real performance

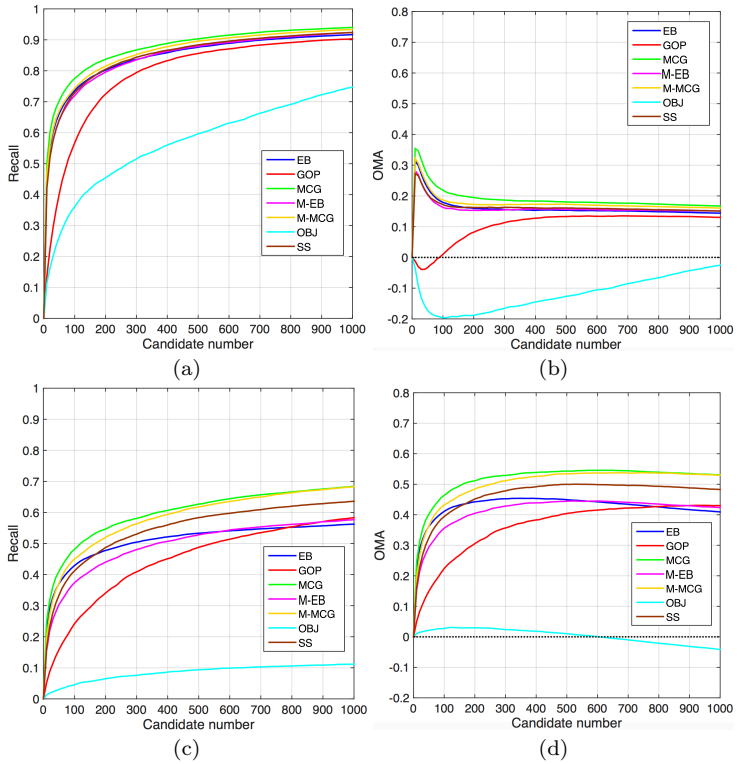


Fig. 5: Evaluation results of different methods using recall *vs.* candidate number and OMA *vs.* candidate number on VOC 2007. (a) Recall *vs.* candidate number (IoU = 0.5). (b) OMA *vs.* candidate number (IoU = 0.5). (c) Recall *vs.* candidate number (IoU = 0.8). (d) OMA *vs.* candidate number (IoU = 0.8).

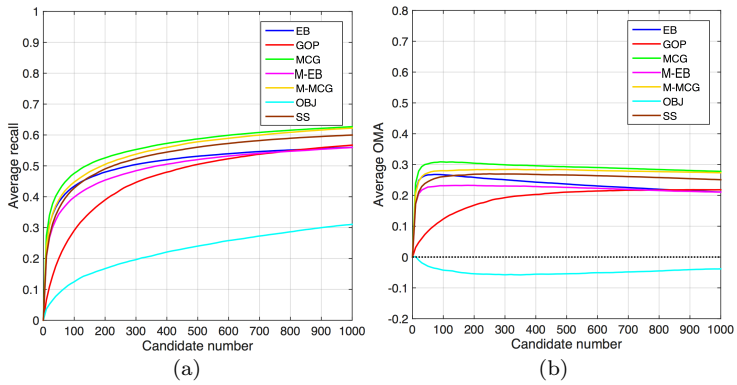
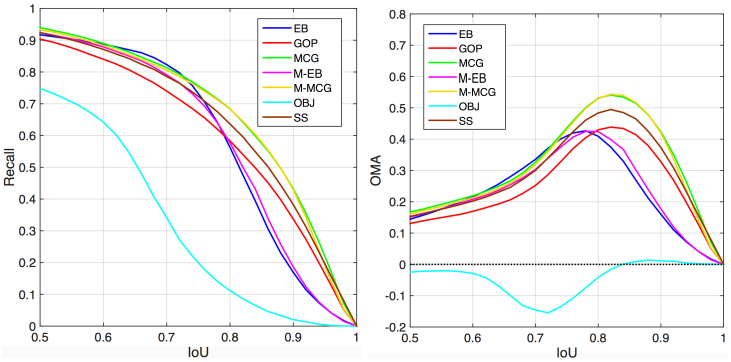


Fig. 6: Evaluation results of different methods using average recall *vs.* candidate number and average OMA *vs.* candidate number on VOC 2007. (a) Average recall *vs.* candidate number. (b) Average OMA *vs.* candidate number.



(a) (b)

Fig. 7: Evaluation results of different methods using recall *vs.* IoU and OMA *vs.* IoU on VOC 2007. (a) Recall *vs.* IoU on the top 1,000 candidates. (b) OMA *vs.* IoU on the top 1,000 candidates.

is worse than that of random sampling under IoUs between 0.5 and 0.8. using OMA *vs.* IoU. It can be easily found when using OMA *vs.* IoU, but it is difficult to realize when only using recall *vs.* IoU. Different to Fig. 5(b) and (d), no method keeps a stable OMA value against IoU. Instead, most methods obtain the best performance around IoU equals 0.8. It is caused by two reasons:

1) Because most object proposal methods adopt appropriate objectness measurement, their performance decreases slowly when IoUs are between 0.5 and 0.8. In contrast, the performance of HPRS drops quickly in this IoU value range. It leads to the increase of OMA values of most methods when IoUs are between 0.5 and 0.8.

2) From Fig. 7(a), we can see that the recall values of most methods drop considerably when IoU increases over 0.8. It means that most methods cannot locate objects correctly under high IoUs. It leads to the degradation of OMA under IoUs larger than 0.8 in Fig. 7(b).

5.4 Criteria Comparison inter Datasets

Object proposal evaluation is usually conducted on multiple datasets to provide comprehensive evaluation results. An interesting question is whether OMA based criteria can generate stable evaluation results to each object proposal method. If so, it means the evaluation on different datasets is redundant.

VOC 2007-few and VOC 2007-multiple. We first decompose VOC 2007 into two datasets according to the object number in each image, named VOC 2007-few and VOC 2007-multiple. The object number in each image in VOC 2007-few is one or two, and that in VOC 2007-multiple is no less than three. The numbers of images in VOC 2007-few and VOC 2007-multiple are 2,874 and 2,078, respectively. We evaluate the seven methods on these two datasets using the criteria in Fig. 5 to 7.

Table 2: Average differences ($\times 10^{-2}$) of evaluation results on VOC 2007-few and VOC 2007-multiple for six object proposal methods. Here, cand. denotes candidate number, AR denotes average recall and AO denotes average OMA, respectively.

average difference	recall *vs.* cand.	OMA *vs.* cand.	AR *vs.* cand.	AO *vs.* cand.	recall *vs.* IoU	OMA *vs.* IoU
EB	23.01	10.20	16.08	2.40	13.90	6.11
GOP	21.67	9.40	15.06	0.68	18.01	7.78
MCG	25.78	11.86	16.88	2.33	15.94	8.88
M-EB	27.18	12.70	17.81	2.63	15.97	6.54
M-MCG	27.07	12.64	17.90	2.58	16.74	8.61
OBJ	6.58	1.02	12.60	0.83	14.53	1.68
SS	28.74	13.64	19.82	3.50	19.25	7.95

Table 3: Average differences ($\times 10^{-2}$) of evaluation results on VOC 2007 and VOC 2012 for six object proposal methods. Here, cand. denotes candidate number, AR denotes average recall and AO denotes average OMA, respectively.

average difference	recall *vs.* cand.	OMA *vs.* cand.	AR *vs.* cand.	AO *vs.* cand.	recall *vs.* IoU	OMA *vs.* IoU
EB	1.87	1.13	3.63	0.43	4.89	1.59
GOP	6.03	3.61	1.46	2.86	3.61	3.23
MCG	1.70	1.01	0.44	2.25	1.34	3.44
M-EB	2.24	1.34	3.16	0.25	4.96	1.74
M-MCG	4.03	2.43	3.87	0.70	3.86	2.58
OBJ	1.55	0.92	1.81	2.73	2.46	1.48
SS	7.62	4.56	1.77	3.04	4.50	3.96

Figure 8 shows the evaluation results of different methods VOC 2007-few and VOC 2007-multiple under the six criteria, namely recall *vs.* candidate number under IoU equals 0.8, OMA *vs.* candidate number under IoU equals 0.8, average recall *vs.* candidate number, average OMA *vs.* candidate number, recall *vs.* IoU on the top 1,000 candidates and OMA *vs.* IoU on the top 1,000 candidates, respectively. In Fig. 8, the solid curves denote the evaluation results on VOC 2007-few and the dotted curves denote the evaluation results on VOC 2007-multiple. Each pair of a solid curve and a dotted curve with the same color in a subfigure denote the evaluation results of an object proposal method on VOC 2007-few and VOC 2007-multiple. We can see that the distance between the evaluation results of the same method on two datasets are smaller under the OMA based criteria than those under the corresponding recall based criteria.

To provide quantitative comparison, we calculate the average distance between the evaluation results of each method on the two datasets under different criteria. Average distance is denoted as the mean difference between the vertical axis distances of all the sampling points of the two curves. Table 2 shows the average differences of each object proposal method on VOC 2007-few and VOC 2007-multiple. We can see that the average differences for all the methods under all the criteria drop significantly when replacing a recall based criterion with the corresponding OMA based one. In particular, the average differences on all the methods decrease over 80% when replacing average recall with average OMA.

VOC 2007 and VOC 2012. We repeat the above experiment on VOC 2007 and VOC 2012. Figure 9 shows the evaluation results of different methods

VOC 2007 and VOC 2012 under the six criteria same to Fig. 8, in which the solid curves denote the evaluation results on VOC 2007 and the dotted curves denote the evaluation results on VOC 2012. Table 3 shows the average differences of each object proposal method on VOC 2007 and VOC 2012.

Unfortunately, the comprehensive degradation on average differences does not appear between VOC 2007 and VOC 2012 when replacing a recall based criterion with the corresponding OMA based one. In contrast, the average differences increase in some cases, such as evaluating GOP with average recall *vs.* candidate number and average OMA *vs.* candidate number. It may be caused by the differences on object appearance diversity or image quality on VOC 2007 and VOC 2012, which are the important and relevant factors of objectness measurement. OMA based criteria perform more stable than recall based criteria on two very similar datasets, such as VOC 2007-few and VOC 2007-multiple, but fail on the datasets with different object appearances, such as VOC 2007 and VOC 2012. Hence, different datasets are still required for comprehensive object proposal evaluation even OMA based criteria are used.

6 Conclusion

We proposed a new object proposal evaluation criterion OMA instead of recall for revealing real abilities of different object proposal methods in objectness measurement. Specially, we defined OMA based on HPRS and extended three commonly used object proposal evaluation criteria by replacing recall with OMA. We compared the evaluation results of six typical object proposal methods on VOC 2007 and VOC 2012 using current recall based criteria and our OMA based criteria. The experimental results illustrated that OMA based criteria are superior to recall based criteria in providing more stable evaluation results, but different datasets are still required for comprehensive evaluation.

Acknowledgements This work is supported by National Science Foundation of China (61321491, 61202320), Undergraduate Innovation Project of Nanjing University (X201610284039), and Collaborative Innovation Center of Novel Software Technology and Industrialization.

References

1. Alexe, B., Deselaers, T., Ferrari, V.: Measuring the objectness of image windows. TPAMI **34**(11), 2189–2202 (2012)
2. Liu, J., Ren, T., Bao, B.K., Bei, J.: Depth-aware layered edge for object proposal. In: ICME, pp. 1–6. IEEE (2016)
3. Bao, B.K., Zhu, G., Shen, J., Yan, S.: Robust image analysis with sparse representation on quantized visual features. TIP **22**(3), 860–871 (2013)
4. Wang, S., Huang, Q., Jiang, S., Tian, Q.: Nearest-neighbor classification using unlabeled data for real world image application. In: MM, pp. 1151–1154 (2010)
5. Chen, Z., Cao, J., Song, Y., Zhang, Y., Li, J.: Web video categorization based on wikipedia categories and content-duplicated open resources. In: MM, pp. 1107–1110 (2010)
6. Song, X., Zhang, J., Han, Y., Jiang, J.: Semi-supervised feature selection via hierarchical regression for web image classification. MMSJ **22**(1), 41–49 (2016)

7. Wang, P., Sun, L., Yang, S., Smeaton, A.F.: Towards training-free refinement for semantic indexing of visual media. In: MMM (2016)
8. Zhang, K., Liu, Q., Song, H., Li, X.: A variational approach to simultaneous image segmentation and bias correction. T CYBERNETICS **45**(8), 1426–1437 (2014)
9. Bai, J., Chen, Z., Feng, B., Xu, B.: Chinese image text recognition on grayscale pixels. In: ICASSP, pp. 1380–1384 (2014)
10. Guo, J., Ren, T., Huang, L., Bei, J.: Saliency detection on sampled images for tag ranking. MMSJ (6), 1–13 (2017)
11. Ren, T., Liu, Y., Ju, R., Wu, G.: How important is location information in saliency detection of natural images. MTAP **75**(5), 2543–2564 (2016)
12. Chen, Z., Sun, L., Yang, S.: Auto-cut for web images. In: MM, pp. 529–532 (2009)
13. Liu, Y., Liu, J., Li, Z., Tang, J., Lu, H.: Weakly-supervised dual clustering for image semantic segmentation. In: CVPR, pp. 2075–2082 (2013)
14. Jiang, F., Hu, H.M., Zheng, J., Li, B.: A hierarchal bow for image retrieval by enhancing feature salience. NEUCOM **175**(PA), 146–154 (2016)
15. Tang, J., Li, H., Qi, G.J., Chua, T.S.: Image annotation by graph-based inference with integrated multiple/single instance representations. TMM **12**(2), 131–141 (2010)
16. Rahman, A.S.M.M., Saddik, A.E.: Mobile based multimodal retrieval and navigation of learning objects using a 3d car metaphor. In: ICIMCS, pp. 103–107 (2011)
17. Zhu, Y., Huang, X., Huang, Q., Tian, Q.: Large-scale video copy retrieval with temporal-concentration sift. NEUCOM **187**(C), 83–91 (2016)
18. Ren, T., Qiu, Z., Liu, Y., Yu, T., Bei, J.: Soft-assigned bag of features for object tracking. MMSJ **21**(2), 189–205 (2015)
19. Sang, J., Xu, C., Lu, D.: Learn to personalized image search from the photo sharing websites. TMM **14**(4), 963–974 (2012)
20. Sang, J., Mei, T., Xu, Y.Q., Zhao, C., Xu, C., Li, S.: Interaction design for mobile visual search. TMM **15**(7), 1665–1676 (2013)
21. Gao, Z., Zhang, H., Xu, G., Xue, Y.: Multi-perspective and multi-modality joint representation and recognition model for 3d action recognition. NEUCOM **151**, 554–564 (2015)
22. Liu, A.A., Su, Y.T., Nie, W.Z., Kankanhalli, M.: Hierarchical clustering multi-task learning for joint human action grouping and recognition. TPAMI **39**(1), 102–114 (2017)
23. Gao, Z., Zhang, Y., Zhang, H., Xue, Y.B., Xu, G.P.: Multi-dimensional human action recognition model based on image set and group sparisty. NEUCOM **215**, 138–149 (2016)
24. Sang, J., Xu, C., Liu, J.: User-aware image tag refinement via ternary semantic analysis. TMM **14**(3), 883–895 (2012)
25. Liu, J., Li, Z., Tang, J., Jiang, Y., Lu, H.: Personalized geo-specific tag recommendation for photos on social websites. TMM **16**(3), 588–600 (2014)
26. Sang, J., Xu, C.: Robust face-name graph matching for movie character identification. TMM **14**(3), 586–596 (2012)
27. Zhu, S., Aloufi, S., El-Saddik, A.: Utilizing image social clues for automated image tagging. In: ICME, pp. 1–6 (2015)
28. Hosang, J., Benenson, R., Dollar, P., Schiele, B.: What makes for effective detection proposals? TPAMI **38**(4), 6644–6665 (2015)
29. Liu, J., Ren, T., Wang, Y., Zhong, S.H., Bei, J., Chen, S.: Object proposal on rgb-d images via elastic edge boxes. NEUCOM **236**, 134–146 (2017)
30. Zitnick, C.L., Dollr, P.: Edge Boxes: Locating Object Proposals from Edges. Springer International Publishing (2014)
31. Cheng, M.M., Zhang, Z., Lin, W.Y., Torr, P.: Bing: Binarized normed gradients for objectness estimation at 300fps. In: CVPR, pp. 3286–3293 (2014)
32. Carreira, J., Sminchisescu, C.: Cpmc: Automatic object segmentation using constrained parametric min-cuts. TPAMI **34**(7), 1312–1328 (2012)
33. Manen, S., Guillaumin, M., Gool, L.V.: Prime object proposals with randomized prim's algorithm. In: ICCV, pp. 2536–2543 (2013)
34. Arbelaez, P., Pont-Tuset, J., Barron, J., Marques, F., Malik, J.: Multiscale combinatorial grouping. In: CVPR, pp. 328–335 (2014)
35. Chavali, N., Agrawal, H., Mahendru, A., Batra, D.: Object-proposal evaluation protocol is 'gameable'. Comp. Sci. (2015)

36. Everingham, M., Gool, L.V., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. IJCV **88**(2), 303–338 (2010)
37. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html
38. Krähenbühl, P., Koltun, V.: Geodesic object proposals. In: ECCV, pp. 725–739 (2014)
39. Chen, X., Ma, H., Wang, X., Zhao, Z.: Improving object proposals with multi-thresholding straddling expansion. In: CVPR, pp. 2587–2595 (2015)
40. Uijlings, J.R.R., Sande, K.E.A.V.D., Gevers, T., Smeulders, A.W.M.: Selective search for object recognition. IJCV **104**(2), 154–171 (2013)

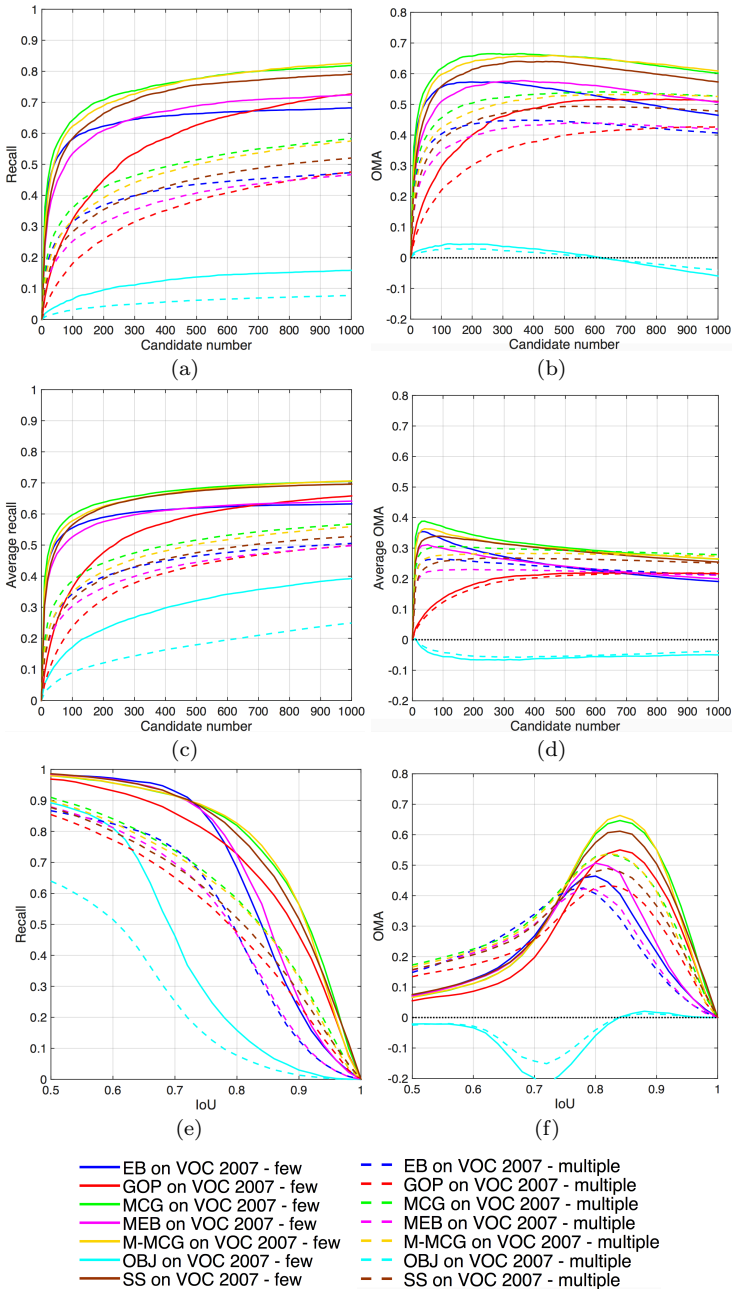


Fig. 8: Evaluation results of different methods on VOC 2007-few and VOC 2007-multiple under six criteria. (a) Recall *vs.* candidate number (IoU = 0.8). (b) OMA *vs.* candidate number (IoU = 0.8). (c) Average recall *vs.* candidate number. (d) Average OMA *vs.* candidate number. (e) Recall *vs.* IoU on the top 1,000 candidates. (f) OMA *vs.* IoU on the top 1,000 candidates. Here, the solid curves denote the evaluation results on VOC 2007-few and the dotted curves denote the evaluation results on VOC 2007-multiple.

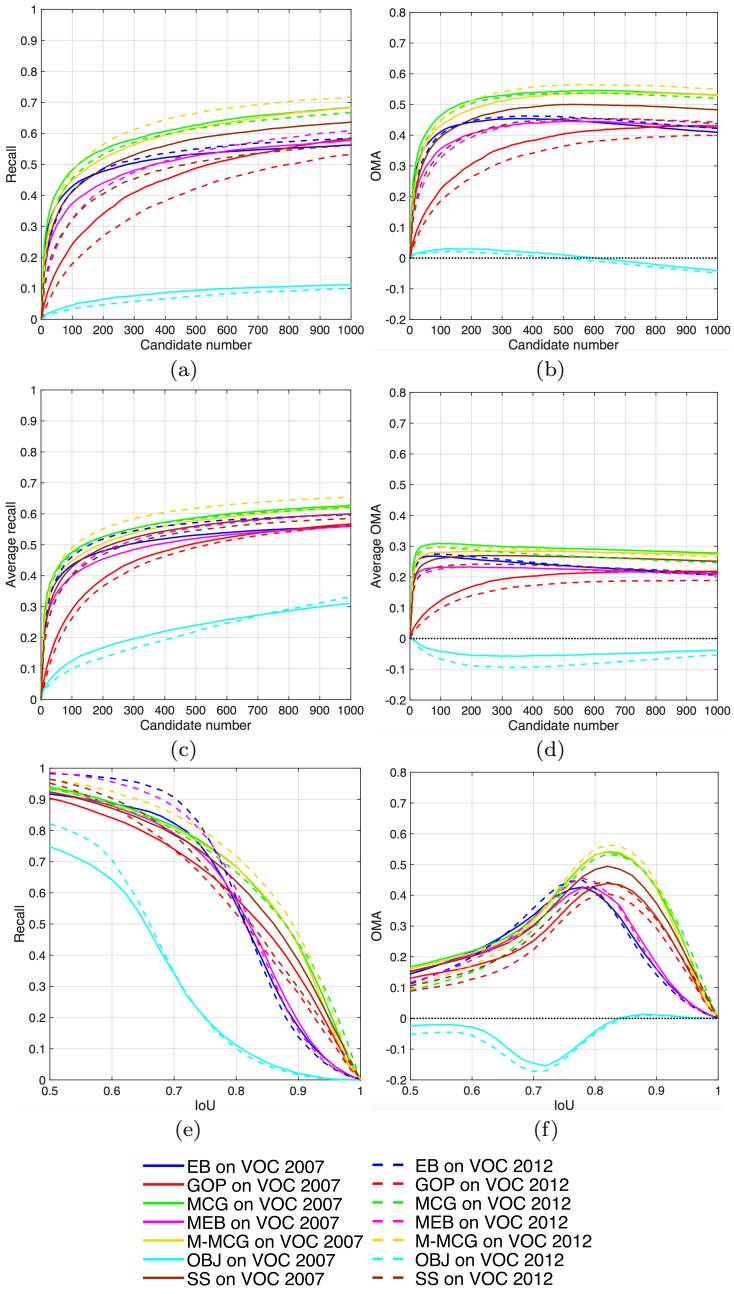


Fig. 9: Evaluation results of different methods on VOC 2007 and VOC 2012 under six criteria. (a) Recall *vs.* candidate number (IoU = 0.8). (b) OMA *vs.* candidate number (IoU = 0.8). (c) Average recall *vs.* candidate number. (d) Average OMA *vs.* candidate number. (e) Recall *vs.* IoU on the top 1,000 candidates. (f) OMA *vs.* IoU on the top 1,000 candidates. Here, the solid curves denote the evaluation results on VOC 2007 and the dotted curves denote the evaluation results on VOC 2012.