Contents lists available at ScienceDirect

Journal of Visual Communication and Image Representation

journal homepage: www.elsevier.com/locate/jvci

Weakly supervised instance segmentation using multi-stage erasing refinement and saliency-guided proposals ordering☆

Zheng Hu [a,b], Zhi Liu [a,b,*], Gongyang Li [a,b], Linwei Ye [c], Lei Zhou [d], Yang Wang [c]

[a] Shanghai Institute for Advanced Communication and Data Science, Shanghai University, Shanghai 200444, China
[b] School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China
[c] Department of Computer Science, University of Manitoba, Winnipeg, MB R3T 2N2, Canada
[d] School of Medical Instrument and Food Engineering, University of Shanghai for Science and Technology, Shanghai, China

ARTICLE INFO

Keywords:
Weakly supervised instance segmentation
Image-level annotations
Multi-stage erasing refinement
Saliency-guided proposals ordering

ABSTRACT

Weakly supervised instance segmentation is a new research topic in the field of computer vision. Compared with fully supervised instance segmentation, weakly supervised methods use weaker data annotations such as points, scribbles or class labels which are easy to obtain. Among these annotations, image-level instance segmentation using only class labels as supervision is the most challenging task. In this paper, we propose a novel weakly supervised instance segmentation framework using a multi-stage erasing refinement method and a saliency-guided proposals ordering method. Firstly, the multi-stage erasing refinement method is exploited to enhance the instance representation by iteratively discovering separate object-related regions, so as to obtain more complete discriminative regions. Then, the saliency-guided proposals ordering method utilizes the saliency map to alleviate the background noise and better select the object proposals for generating the instance segmentation result. Experimental results on the PASCAL VOC 2012 dataset and the COCO dataset demonstrate that our framework achieves superior performance compared with the state-of-the-art weakly supervised instance segmentation models and the ablation study shows the effectiveness of the proposed two methods.

1. Introduction

Instance segmentation is a challenging task in the field of computer vision. The goal is to segment a number of instances present in an image and to predict both class label and instance label for each instance. With the development of deep learning which has achieved remarkable progresses in many fields such as image classification [1,2], the recently proposed instance segmentation models [3–8] also achieve a significant improvement by using convolutional neural networks (CNNs). However, these instance segmentation models heavily rely on a large amount of training data with annotations of pixel-level masks and class labels to achieve the high performance. The annotation work especially the pixel-level mask annotation is expensive and time-consuming. Weakly supervised methods can alleviate the tedious annotations. Compared with pixel-level mask annotations, some weak annotations, *e.g.*, points, scribbles and bounding boxes are much cheaper to collect but still require a bit of human efforts. To further reduce the reliance on aforementioned weak annotations, some work [9–12] tries to use only image-level class labels to design the instance segmentation networks since the image-level class labels require minimal efforts.

Compared with pixel-level mask annotations, image-level class labels discard the location and shape information of objects. The key issue of image-level weakly supervised methods is how to localize objects with image-level class labels. In [14], Zhou *et al.* gave a popular choice for object localization by Class Activation Map (CAM), which builds the relationship between each pixel's localization and classification result, and indicates the class-related regions in the image. But CAM cannot tell from different object instances in the same class. To resolve this issue, Zhou *et al.* [9] proposed to boost the local maximum in the CAM to localize objects. The local maximum points are propagated back to compute the gradient map called peak response map (PRM). PRM provides shape and location information of objects. It is used to generate the instance masks with the prior object information, *i.e.* object proposals. However, both CAM and PRM have a common drawback that the regions highlighted in them only contain the discriminative parts of objects but miss the other object-related regions.

☆ This paper has been recommended for acceptance by Dr. Zicheng Liu.

* Corresponding author at: School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China.
E-mail address: liuzhisjtu@163.com (Z. Liu).

https://doi.org/10.1016/j.jvcir.2020.102957
Received 20 January 2020; Received in revised form 28 September 2020; Accepted 24 October 2020
Available online 29 October 2020

In this paper, we propose a novel weakly supervised instance segmentation framework, which is able to detect the foreground object-related regions to represent the objects with image-level labels and adopt saliency map to suppress object-irrelevant regions, and generates the instance masks with the prior object proposals. We propose two methods, *i.e.* Multi-Stage Erasing Refinement (MSER) method and Saliency-Guided Proposals Ordering (SGPO) method, to overcome the shortcomings of PRM [9].

Since CNNs tend to capture the most discriminative features to predict the classification results, the object-related regions directly discovered by PRM might be incomplete. We propose the MSER method to enhance the initial PRM. Concretely, by training multiple classification networks with different concentrations, the MSER method can discover complementary object-related regions to form the enhanced peak response map (EPRM), which is a more fine-detailed instance representation. Besides, PRM is generated through the back propagation from the local maximum points in the feature map. However, the pixels of some background regions in the image may be falsely activated since these background regions also contain strong textural features and are close to the local maximum points. To solve this problem, the proposed SGPO method introduces the saliency map to better distinguish the objects from background regions and to alleviate the background noise in the PRM. The instance segmentation masks are retrieved from a pool of candidate object proposals by the SGPO method, which chooses the desired proposal with the maximum overlap, the similar shape with the EPRM and the maximum overlap with the saliency-guided instance representation. Notably, during the process of proposals ordering, the saliency map is helpful to select the desired proposals to constitute different object instances.

To sum up, our main contributions are three-fold:

1) We propose a novel weakly supervised instance segmentation framework, which is equipped with the MSER and the SGPO. Our framework is able to detect more object-related regions to better represent object instances, and introduces the saliency map to better select object proposals for instance segmentation.
2) We propose the Multi-Stage Erasing Refinement (MSER) method to generate the enhanced PRM, which can discover the object-related regions and form a more fine-detailed instance representation to better provide the location and boundary information of objects.
3) We propose the Saliency-Guided Proposals Ordering (SGPO) method, which introduces the saliency map to alleviate the adverse effect of background regions for better selecting the desired object proposals.

The remainder of this paper is organized as follows. We introduce the related work in Section 2. In Section 3, we describe the proposed framework in detail. Experimental results are presented in Section 4, and the conclusion is drawn in Section 5.

2. Related work

In this section, we briefly introduce several groups of previous work related to this work.

Fully supervised instance segmentation. With the development of deep learning, some researchers have developed various CNNs-based instance segmentation models [3–8]. For example, Mask R-CNN [5] is a simple and effective instance segmentation model with three branches to predict the class labels, bounding boxes and masks of objects. In [7], Liu *et al.* followed the architecture of Mask R-CNN and proposed several modules to improve the performance. In [8], Chen *et al.* proposed a hybrid task cascade which interweaves detection and segmentation features to improve the performance. The frameworks of these instance segmentation models contain multiple tasks, and each of them is trained by one type of annotations such as class labels, bounding boxes and pixel-level masks. The CNNs-based methods heavily rely on a large amount of training data with various types of annotations to achieve a high performance. However, the work of annotating these labels especially the pixel-level masks costs a lot of human efforts and time. As a result, instance segmentation is confined to a limited range of datasets and object categories. The commonly used segmentation datasets (*e.g.* PASCAL VOC 2012 [15] and MS COCO [16]) are restricted to a few dozen of object classes, far away from the number of categories in the image classification datasets (*e.g.* ImageNet [17]).

Weakly supervised instance segmentation. In the past few years, some researchers have proposed several weakly supervised instance segmentation models. In [18], an end-to-end weakly supervised instance segmentation network is trained with point-level annotations and noise samples. In [19], Khoreva *et al.* used a recursive training strategy to train the semantic segmentation network iteratively with the bounding box annotations. The network progressively achieves the better segmentation result after each stage's training. Finally, the network can accurately segment the objects in the bounding boxes detected by Fast R-CNN [20]. In [21], Lin *et al.* proposed a weakly supervised method based on scribble annotations and optimized a graphical model for propagating information from scribbles. In [9], Zhou *et al.* leveraged class labels to predict instance masks by boosting the local maximum points in the feature map output by the last convolution layer of the CNN, which is originally for image classification. These local maximum points, which can localize the object instances, are back propagated to compute for each point the gradient map, which is called the peak response map (PRM). PRM provides shape and location information of objects, and is further used to segment instances with object proposals. In [10], Zhu *et al.* designed an Instance Extend Filling module, which leverages PRM to generate accurate Instance Activation Maps (IAM) to represent instances. In [11], Cholakkal *et al.* proposed an image-level supervised density map estimation approach to provide both object count and spatial distribution of object instances. In [12], Ge *et al.* proposed a Sequential Label Propagation and Enhancement Network, which progressively transforms image-level labels to pixel-wise labels in a coarse-to-fine manner.

Weakly supervised semantic segmentation with image-level annotation. Compared with pixel-level mask annotations, image-level class annotations are easy to obtain, but they lose location and shape information in the weakly supervised semantic segmentation. Therefore, many approaches [22–28,30] focus on simulating the absent information. For example, in [24], saliency maps are utilized to replace the ground truths for training a fully supervised network. In [13], Meng *et al.* proposed a new cosegmentation and fusion-based strategy for weakly supervised semantic segmentation, which can sufficiently use the labels of images. In [14], Zhou *et al.* gave a popular choice for object localization by Class Activation Map (CAM), which builds the relationship between each pixel's localization and the classification result to localize the objects. Based on CAM, some weakly supervised semantic segmentation methods [26,27,30] were proposed. In [26], the AffinityNet trained with CAM is designed to predict semantic affinities of pairs of adjacent image coordinates and to diffuse the discriminative regions in CAM according to the affinities of pairs. In [27], a deep seeded region growing network iteratively generates the class-related regions to update the CAM. In [30], a three-stage adversarial erasing process is designed to discover class-related regions, which finally constitute the entire objects in the image. In [31], the Guided Attention Inference Network is proposed to provide a direct guidance on attention maps, so as to generate more accurate and more complete attention maps.

Erasing strategy. The erasing strategy has been used in weakly supervised semantic segmentation. For example, in [29], Zhang *et al.* proposed a two-branch network. The class-related regions are detected in the feature map of the first branch and erased in the feature map of the second branch. The network generates two separate CAMs and combines them to segment the image. In [30], class-related regions are detected in CAMs and erased in the image for three times. The class-related regions are combined to represent the object of specific category. Li *et al.* [31] proposed the Guided Attention Inference Network to predict the class

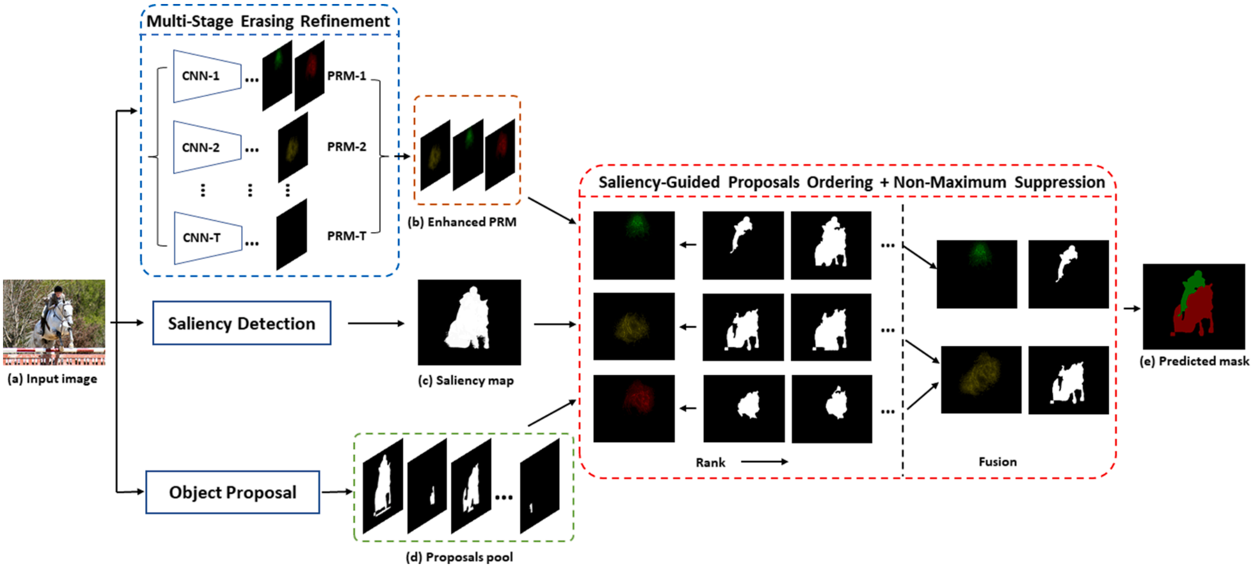

Fig. 1. Overview of the proposed weakly supervised instance segmentation framework. Given an input image in (a), we first adopt the multi-stage erasing refinement method to iteratively discover different object-related regions to produce several PRMs, and combine them to generate the enhanced PRM in (b). Then, we generate the saliency map in (c) and the object proposals pool in (d). We rank candidate object proposals with the saliency-guided proposals ordering method according to four measures and integrate the overlapped object proposals. Finally, we select the most matching proposal of each object as the predicted mask in (e).

labels of the original image and the erased image, and designed the attention mining loss to guide the network focus on the whole object of interest. Compared with the above erasing strategies used in weakly supervised semantic segmentation [29–31], we adapt the erasing strategy to instance segmentation. We detect the local maximum points in the feature map to locate the instance and backward propagate the points to extract the object-related regions of the detected instance in the image. As the CNN tends to capture the most discriminative regions and ignore the other object-related regions, the instance representation discovered in PRM [9] is less complete. Therefore, we erase the most discriminative parts of the discovered instances in the original image and slightly change the appearances of the instances in the image. Compared with the erasing strategies used in weakly supervised semantic segmentation [29–31], we erase much fewer regions in the image to maintain instances, so that the newly trained network can still capture the instances correctly and discover new object-related regions.

3. Proposed framework

In this section, we first give a brief description of our framework in Section 3.1. Then, we describe the multi-stage erasing refinement method in detail in Section 3.2. In Section 3.3, we present the saliency-guided proposals ordering method. In the end, we introduce the implementation details in Section 3.4.

3.1. Framework overview

As shown in Fig. 1, given an input image, our goal is to extract object-related regions in the image and to exploit the object-related regions to obtain the object mask from a set of object proposals. We first adopt the multi-stage erasing refinement method to iteratively discover object-related regions and combine them to represent the whole objects in detail. The initial PRMs will be iteratively enhanced towards a more complete instance representation named as the Enhanced PRM (EPRM), which is more informative than the initial PRMs and can indicate more accurate locations and complete boundaries of objects in the image. Then, a saliency detection method, *i.e.* R3Net [32], is applied to the input image, generating the saliency map. After that, we use the off-the-shelf object proposal method, *i.e.* COB [33], to generate a set of candidate object proposals in the proposals pool. In the end, the EPRM, the saliency map and the candidate object proposals are processed with the saliency-guided proposals ordering method to generate the instance segmentation mask. Specifically, the instance segmentation mask is retrieved from a set of desired candidate object proposals which have the maximum overlap and the similar shape with the EPRM and the minimum overlap with background. And we apply the non-maximum suppression scheme to integrate the overlapped object proposals.

3.2. Multi-stage erasing refinement method

The PRM method [9] constructs fully convolution layers followed by a convolutional layer with 1 × 1 kernel size to generate the feature map, which indicates the responses of the image to different object classes. In the feature map, the local maximum points are exploited to roughly localize the objects. The local maximum points are back propagated to compute the PRMs, which measure the contribution from pixels to the local maximum points. To generate the PRMs, the forward propagation is first performed to detect local maximum points in the feature map. Then by setting each local maximum point to 1 and the other positions to 0 in the feature map, the backward propagation is performed to calculate the gradient map, which is termed as PRM. Each PRM can reflect the regions contributing to each local maximum point. A group of PRMs are generated after processing all the local maximum points in turn. However, the object-related regions directly discovered by the PRMs might be incomplete, due that the CNN tends to capture the most discriminative features to predict the classification results. To address the problem of less activated object-related regions, we propose a Multi-Stage Erasing Refinement (MSER) method to discover complementary object-related regions for PRMs enhancement.

The fundamental idea of the MSER method is to construct an updated training image set by erasing the detected discriminative regions to slightly change the appearances of instances in the previous training image set. The updated training image set is used to train a CNN for image classification in each stage. Since the highly discriminative object-related regions have been erased from images and they no longer contribute to the classification result, the newly trained CNN in the current stage has to shift the concentration to the other object-related regions for classification. In this way, the less discriminative object-related regions will be enhanced iteratively during the refinement process. The MSER method iteratively trains CNN with the updated image

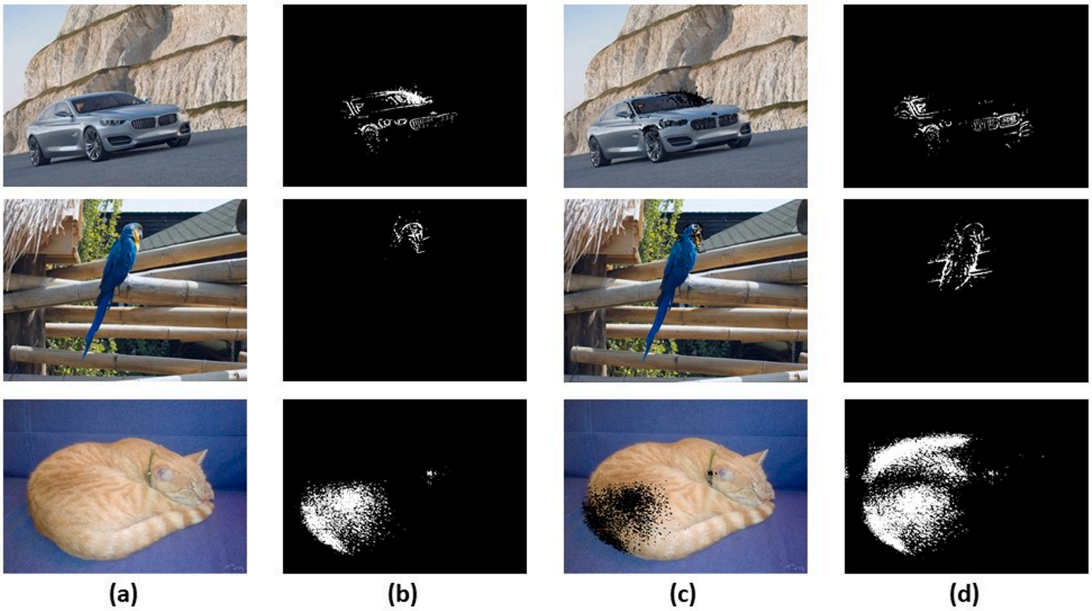

Fig. 2. Visualization of the process of erasing foreground regions in MSER. (a) Input image, (b) Discriminative regions, (c) Image after the erasing process, and (d) New discriminative regions detected by the model trained with the erased images.

set for discovering the object-related regions, and updates the training image set by erasing the detected object-related regions in the images. After several stages, we can obtain a group of CNNs which have different concentrations on different object-related regions. Then, we use these CNNs to generate separate and complementary regions belonging to the same object. Finally, these regions comprise a more fine-detailed instance representation, *i.e.* Enhanced PRM (EPRM). In the following, we will provide a detailed description of the MSER method.

The MSER method consists of two phases including training phase and inference phase. During the training phase, we iteratively perform two steps, *i.e.* CNNs training and training image set updating, to obtain a group of CNNs. We describe the training phase in detail as follows:

CNNs training. Firstly, the original training image set $I^0 = \{I_i^0, L_i\}_{i=1}^N$, with each image I_i^0 and the corresponding class label L_i, is used to train the initial image classification network $\aleph_0$. In particular, we use ResNet-50 [2] as the backbone and remove the fully connected layers to obtain a fully convolutional network (FCN) so as to maintain the spatial information for segmentation. Besides, we apply another convolutional layer with 1 × 1 kernel size to reduce the channel dimension of the output feature map to the number of classes. The output feature map of each image I_i^0 is denoted as F_i^0, with a size of $C\times H\times W$, where C is the number of channels and $H\times W$ is the spatial size. Each channel of F_i^0 reflects the response to each object class. Then, a set of local maximum points of each image I_i^0, denoted as $K_i^0 = \{K_{i,j}^0\}_{j=1}^M = \{\left(c_{i,j}^0, x_{i,j}^0, y_{i,j}^0\right)\}_{j=1}^M$, are searched on the feature map F_i^0. Here, M is the number of the local maximum points, $\left(x_{i,j}^0, y_{i,j}^0\right)$ indicates the spatial coordinates of the j^{th} point and $c_{i,j}^0$ indicates the class. To generate the set of local maximum points K_i^0, we first slide the maximum filter with a size of 3*3 over the feature map F_i^0 to select each point whose response value is greater than its eight neighboring points, and then we remove those local maximum points whose response values are lower than the global median value of the feature map F_i^0. The class confidence score of the c^{th} class, s_c, is calculated by averaging the response values of all local maximum points belonging to the c^{th} class as $s_{i,c}^0 = \frac{1}{\sum c_j\in c}\sum_{c_j\in c}K_{i,j}^0$. Then the multi-label soft margin loss is defined as follows:

$$\mathscr{L}(s,l) = -\sum_{c=1}^{C}[l_c log\frac{e^{s_c}}{1+e^{s_c}} + (1-l_c)log\frac{1}{1+e^{s_c}}] \tag{1}$$

where s is the predicted class confidence vector and l is the vector representing the corresponding class label.

Training image set updating. After finishing training the network, we use the trained network $\aleph_0$ to update the training image set I^0 to I^1. First, the local maximum points $\{K_{i,j}^0\}_{j=1}^M$ in each image I_i^0 are separately backward propagated to compute a set of PRMs, *i.e.* $P_i^0 = \{P_{i,j}^0\}_{j=1}^M$, for the image I_i^0. As the local maximum points can roughly localize objects, the generated PRMs can roughly indicate parts of objects. The regions high-lighted in the PRMs contain most foreground objects and few background noise. In order to attenuate the effect of background noise, we then apply a threshold δ on each PRM, *i.e.* $P_{i,j}^0$, to generate the binary map $R_{i,j}^0$, in which the discriminative object-related points with higher response values in PRM are labeled with "0″ and the remaining points are labeled with "1" as follows:

$$R_{i,j}^0(x,y) = \begin{cases} 0, P_{i,j}^0(x,y) \geq \delta \\ 1, otherwise \end{cases} \tag{2}$$

Using Eq. (2) as an indicator, we can erase the most discriminative parts of the discovered instances in the original image and slightly change the appearances of the instances in the image. Thus, the newly trained CNN will shift the concentration to the other object-related regions for classification. In order to train a network with different concentrations to discover new complementary object-related regions for enhancing the initial PRMs, we erase the object-related regions indicated by $\{R_{i,j}^0\}_{j=1}^M$ from the original training images, by replacing the pixel values of these object-related regions with zero, and obtain the updated training image set $I^1 = \{I_i^1, L_i\}_{i=1}^N$ as follows:

$$I_i^1(x,y) = I_i^0(x,y) - I_i^0(x,y)\cdot[1 - \prod\nolimits_{j=1}^M R_{i,j}^0(x,y)] \tag{3}$$

where Π is the multiplication operation. The updated training image set I^1 is used to train a new network, *i.e.* $\aleph_1$. Since the most discriminative regions have been removed from the training images and do not

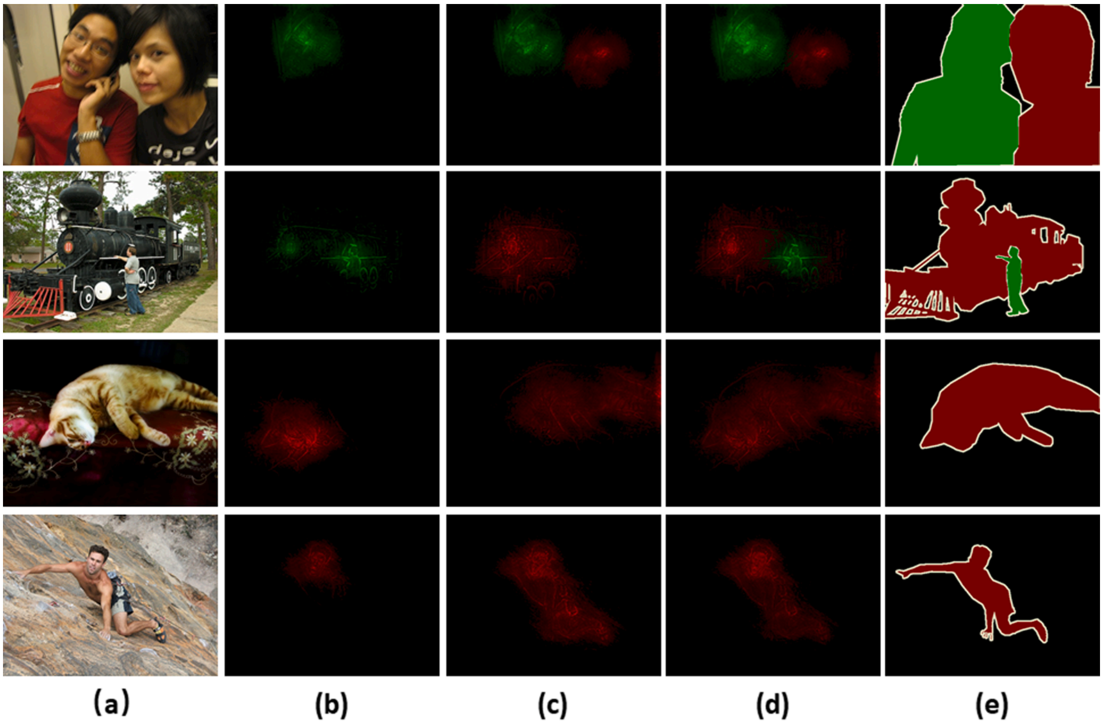

Fig. 3. Visualization of the process of MSER with two stages. (a) Input image, (b) initial PRM, (c) complementary PRM from the newly trained CNN, (d) EPRM and (e) ground truth. By fusing the initial PRM and complementary PRM, the EPRM in (d) is more complete and more accurate than the initial PRM in (b).

contribute to the classification prediction, $\aleph_1$ has to activate other object-related regions to maintain the classification results. Similarly, a set of PRMs, *i.e.* P_i^1, are generated via the backward propagation process of $\aleph_1$. We visualize the process of erasing discriminative regions in Fig. 2. Fig. 2(b) shows the discriminative regions which can roughly indicate parts of objects, and Fig. 2(c) shows the images in which the discriminative regions have been erased. Besides, Fig. 2(d) shows the new discriminative regions detected by the newly trained CNN with the erased images.

We repeat CNNs training and training image set updating for several times, until the number of iteration process reaches the maximum erasing stage number, T. By this way, we can obtain a group of image classification networks, *i.e.* $\{\aleph_t\}_{t=1}^T$, and the T networks are used to generate multiple object-related regions. The whole process of training phase is summarized in Algorithm 1.

During the inference phase, given a test image, we send the image into the trained networks, $\{\aleph_t\}_{t=1}^T$, to generate a group of PRMs $\{P_t\}_{t=1}^T$ for object-related regions. These PRMs comprise a more fine-detailed instance representation, *i.e.* EPRM. We set the maximum erasing stage number, T, to 2 and show visualization of the process of MSER with two stages in Fig. 3. It can be observed that the initial PRMs capture the discriminative parts of objects and only provide the incomplete location and boundary information. Concretely, the first and second rows in Fig. 3(b) indicate that the initial PRMs fail to detect the objects, while the third and fourth rows in Fig. 3(b) indicate that the initial PRMs fail to capture the entire object. The complementary PRMs in Fig. 3(c) are produced from the newly trained CNNs, which are trained with the training images after erasing. After generating the complementary PRMs, we feed the complementary PRMs and the initial PRMs, *i.e.* EPRM in Fig. 3(d), to SGPO to predict the instance mask. Obviously, the EPRMs can provide more fine-detailed location and boundary information of object instances than the initial PRMs.

Algorithm 1. Training phase of MSER method

```
Input: Training data $I = \{I_i, L_i\}_{i=1}^N$.
```

(continued on next column)

(continued)

Algorithm 1. Training phase of MSER method

```
Output: A group of image classification networks$\aleph = \{\aleph_t\}_{t=1}^T$.
Initialize: $\aleph = \varnothing$, iteration times $t = 1$, flag $=$ true, training image set $I^0 = I$.
while (flag) do
    Train the image classification network $\aleph_t$ using the training image set $I^t$ with the loss function defined in Eq. (1).
    Add $\aleph_t$ into the network group $\aleph$.
    for $I_i^t$ in $I^t$ do
        Use $\aleph_t$ to classify $I_i^t$, and extract the final output feature map $F_i^t$.
        Detect the local maximum points $K_i^t = \{K_{i,j}^t\}_{j=1}^M$ in the feature map $F_i^t$.
        for $K_{i,j}^t$ in $K_i^t$ do
            Set the pixel value of $K_{i,j}^t$to 1 and the rest pixels in $F_i^t$ as 0.
            Generate the PRM $P_{i,j}^t$ through the backward propagation with $F_i^t$.
            Generate $R_{i,j}^t$ by thresholding $P_{i,j}^t$ to represent discriminative object-related regions similarly as Eq. (2),
                $R_{i,j}^t(x,y) = \begin{cases} 0, P_{i,j}^t(x,y) \geq \delta \\ 1, otherwise \end{cases}.$
        end for
        Erase the discriminative regions based on $R_i^t = \{R_{i,j}^t\}_{j=1}^M$ from $I_i^t$, and generate $I_i^{t+1}$ similarly as Eq. (3),
            $I_i^{t+1}(x,y) = I_i^t(x,y) - I_i^t(x,y) \cdot \prod_{j=1}^M R_{i,j}^t(x,y)$.
    end for
    Update the training image set as $I_i^{t+1}$.
    if $t = T$
        Set flag $=$ false.
        Set the length of $\aleph$ as $T = t$.
    end if
    $t = t + 1$.
end while
Output: $\aleph = \{\aleph_t\}_{t=1}^T$.
```

3.3. Saliency-guided proposals ordering method

After generating the EPRM, we propose the Saliency-Guided Proposals Ordering (SGPO) method to generate the instance segmentation mask by retrieving the candidate object proposals. We employ a scoring

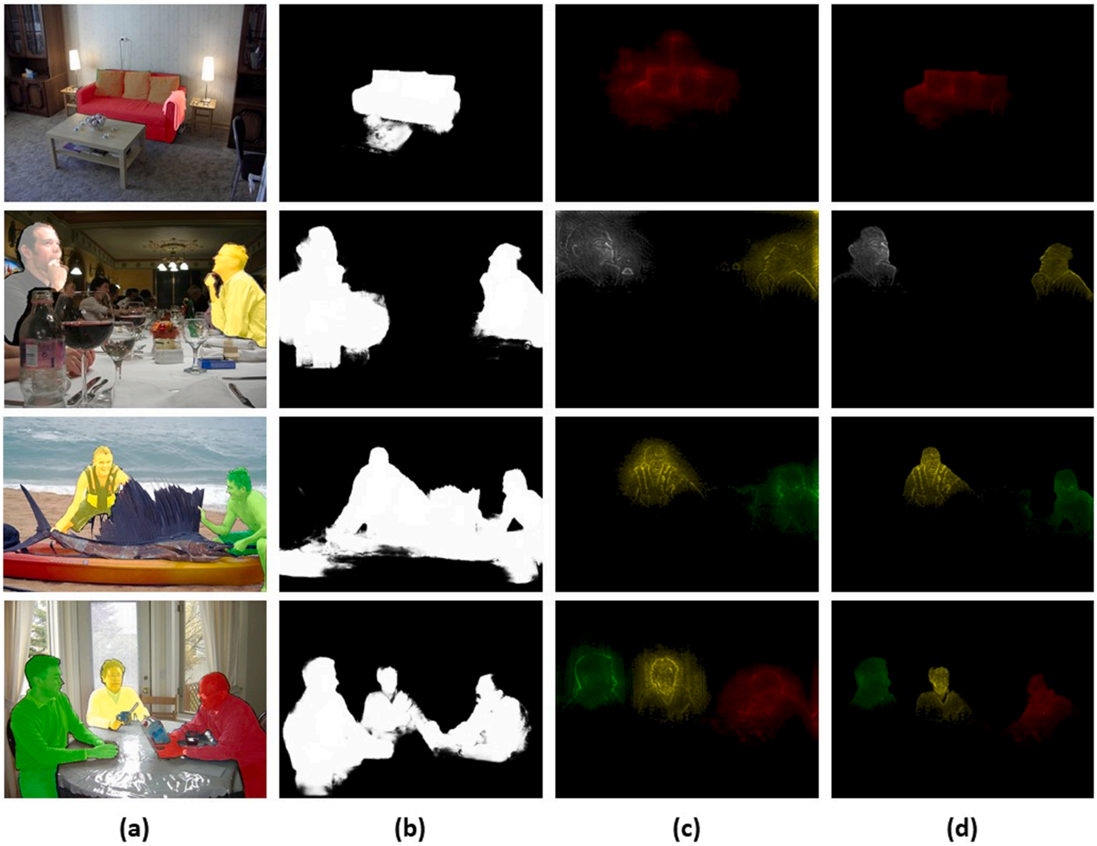

Fig. 4. Visualization of the process of saliency-guided refinement. (a) Input image with the annotated object instances, (b) saliency map generated using R3Net, (c) EPRM, and (d) the refined EPRM, which is generated by multiplying the saliency map and EPRM.

metric to rank off-the-shelf object proposals based on the EPRM for generating the instance segmentation mask. The scoring metric consists of four terms including instance-aware information and class-aware information from EPRM, boundary-aware term from object proposals, and saliency-aware term from saliency map. For each candidate object proposal OP_i, which is a binary mask with the same size of image, the scoring metric is defined as follows:

$$Score(OP_i) = \alpha \cdot S_i^{ins} + \beta \cdot S_i^{bou} + \gamma \cdot S_i^{cla} + \omega \cdot S_i^{sal} \quad (4)$$

Following [9], the first three terms are used to encourage object proposals that have the maximum overlap and the similar shape with the EPRM.

In our SGPO method, we introduce the saliency-aware term, S_i^{sal}, which calculates the maximum overlap with the instance representation according to the saliency map, to encourage object proposals in the salient regions. Since the EPRM may contain some background noises, we introduce saliency map to object proposals ordering as the saliency detection models are effective in capturing salient regions. The saliency map can distinguish between the foreground objects and background regions. Motivated by this, we propose the saliency-aware term to better rank object proposals. We use R3Net [32] to generate the saliency map, which is exploited to encourage the salient regions in the EPRM when ordering candidate object proposals.

Specifically, for each candidate object proposal OP_i, the instance-aware term is used to encourage OP_i to have the maximum overlap with the EPRM E, and is defined as follows:

$$S_i^{ins} = \sum_{x,y} E(x,y) \cdot OP_i(x,y) \quad (5)$$

The boundary-aware term calculates the response of the EPRM on the boundary of the object proposal to encourage the object proposal to have the similar shape with the EPRM, and is defined as follows:

$$S_i^{bou} = \sum_{x,y} E(x,y) \cdot CP_i(x,y) \quad (6)$$

where CP_i is the binary mask indicating the boundary of OP_i.

The class-aware term is used to punish the c lass-irrelevant regions in the EPRM. Suppose that the c^{th} class is associated with the EPRM and F_c indicates the c^{th} channel of the final output feature map, the class-irrelevant map $F_c^{'}$ is obtained by reserving the pixels with low responses in F_c. Concretely, $F_c^{'}$ keeps the responses of those pixels lower than the mean value of F_c by setting the responses of the remaining pixels to zero. The class-aware term is then defined as follows:

$$S_i^{cla} = -\sum_{x,y} F_c^{'}(x,y) \cdot OP_i(x,y) \quad (7)$$

The saliency-aware term is designed to encourage the salient regions in the EPRM. The saliency map assigns the salient regions with higher saliency values, and we enhance the responses of salient regions in the EPRM by multiplying the EPRM with the saliency map Sm as follows:

$$S_i^{sal} = \sum_{x,y} E(x,y) \cdot Sm(x,y) \cdot OP_i(x,y) \quad (8)$$

In Eq. (8), the element-wise multiplication between EPRM and saliency map is the saliency-guided EPRM refinement, as shown in Fig. 4. We can observe from the examples in Fig. 4 that the EPRMs can highlight most parts of objects in images while some background regions are also falsely activated in some EPRMs. A pixel with a higher saliency value is highly likely to belong to an object instance. Thus, after the saliency-guided EPRM refinement using Eq. (8), the salient regions indicated by the saliency maps as shown in Fig. 4(b) have stronger responses in the refined EPRMs as shown in Fig. 4(d), while the background regions indicated by the saliency maps have weaker responses in the refined EPRMs. This confirms that the saliency map is helpful to better represent the object instances and suppress the background noise.

With the introduction of the saliency-aware term in Eq. (8) as well as the other three terms in Eqs. (5)–(7), we calculate the scoring metric for each candidate object proposal using Eq. (4). We separately select for each instance representation the most matching object proposal, which has the maximum scoring metric. After retrieving the most matching object proposals for all instances, we apply the non-maximum suppression scheme to integrate the overlapped object proposals for generating the instance mask. The non-maximum suppression is important to segment the instances. As the MSER discovers much object-related regions which are helpful to represent the instance. These object related regions will be separately assigned a proposal. Some regions represent the different parts of the instance. The non-maximum suppression will combine these highly overlapped proposals as these proposals are likely to belong to the same instance.

3.4. Implementation details

Following the implementation of PRM [9], we use the ResNet-50 [2] as the backbone and remove the last two fully connected layers. The parameters of ResNet-50 are pre-trained on ImageNet [17], and then finetuned on the PASCAL VOC 2012 training set [15] and the COCO training set [16] with corresponding class labels, respectively. The initial learning rate of the backbone is set to 10^{-4}, and the other part of the network is trained with an initial learning rate of 0.01. A mini-batch size is set to 16 for the SGD optimizer [34]. The momentum and weight decay are set to 0.9 and 10^{-4}, respectively. The threshold δ in Eq. (2) is set to 30. The parameters in Eq. (4), *i.e.* α, β, γ and ω in the scoring metric are empirically set as in [9].

4. Experimental results

4.1. Datasets and evaluation metrics

We train and evaluate our framework on PASCAL VOC 2012 dataset [15] and COCO dataset [16]. These two datasets are widely used for image classification [35], object detection [36], semantic segmentation [37], and instance segmentation [9].

PASCAL VOC 2012: For the image classification task, it contains 5,717 images for training and 5,823 images for validation. For the instance segmentation task, it contains 1,442 images for training and 1,449 images for validation. Each image in the training set and validation set is annotated with pixel-level mask to indicate the class label and instance label of each pixel. The dataset contains 20 classes of objects including inanimate objects such as airplanes and living objects such as humans. Following the experimental setting in [9], we adopt the image classification dataset, which includes the training set and validation set with a total of 11,540 (5,717 + 5,823) images and the corresponding image-level labels, to train a group of image classification networks using our MSER method, and we evaluate our framework on the validation set for instance segmentation including 1,449 images.

COCO: It contains 82,783 images for training and 40,504 images for validation. The dataset contains 80 classes of objects including inanimate objects and living objects. We adopt the image classification dataset, which includes the training set with a total of 82,783 images and the corresponding image-level labels, to train a group of image classification networks using our MSER method, and we evaluate our framework on the validation set for instance segmentation including 40,504 images.

Evaluation metrics: We use the most widely used four evaluation metrics, *i.e.* mean average precision (*mAP*) for Intersection-over-Union (IoU) of 0.25, 0.5, 0.75 and Average Best Overlap (ABO) [38] to evaluate the instance segmentation performance.

Table 1
Analysis of the parameters in SGPO on the PASCAL VOC 2012 validation set in percentage %.

Parameters	Metric	1.3	1.1	1(Ours)	1/1.1	1/1.3	0
α	$mAP^r_{0.25}\uparrow$	44.0	45.2	**46.1**	45.1	43.7	26.89
	$mAP^r_{0.5}\uparrow$	25.0	26.2	**27.3**	26.3	25.0	9.36
	$mAP^r_{0.75}\uparrow$	9.4	10.1	**10.4**	9.7	9.3	2.39
β	$mAP^r_{0.25}\uparrow$	43.9	44.9	**46.1**	45.2	43.9	26.88
	$mAP^r_{0.5}\uparrow$	24.6	26.0	**27.3**	26.4	25.2	9.40
	$mAP^r_{0.75}\uparrow$	9.5	9.8	**10.4**	10.0	9.1	2.39
γ	$mAP^r_{0.25}\uparrow$	44.7	45.4	**46.1**	45.6	44.8	40.1
	$mAP^r_{0.5}\uparrow$	25.9	26.5	**27.3**	26.3	25.9	23.5
	$mAP^r_{0.75}\uparrow$	9.5	9.8	**10.4**	10.0	9.9	9.4
ω	$mAP^r_{0.25}\uparrow$	44.7	45.5	**46.1**	45.5	44.6	32.6
	$mAP^r_{0.5}\uparrow$	25.5	26.2	**27.3**	26.7	25.9	16.1
	$mAP^r_{0.75}\uparrow$	9.6	9.9	**10.4**	10.1	9.5	4.9

Notably, for a clear comparison, we report results of α and β with two decimals in the last column “0”.

4.2. Parameters Analysis

In this section, we make a detailed analysis for some parameters of our method on the PASCAL VOC 2012. Concretely, we evaluate 1) the effectiveness of the parameters α, β, γ and ω in SGPO; 2) the influence of the maximum clear stage T; 3) the influence of the saliency detection methods, 4) the influence of the parameter $\boldsymbol{\delta}$ in MSER.

1. The effectiveness of the parameters in SGPO. The parameters α, β, γ and ω in SGPO are empirically set as in [9]. To validate the effectiveness of parameter settings for α, β, γ and ω, we adjust them to evaluate the impact of each parameter separately. We simply multiply the five rates including 1.3, 1.1, 1, 1/1.1 and 1/1.3 to each parameter and evaluate the performance. Table 1 shows the influence of adjusting a single parameter on the performance. For example, we enlarge the value of parameter α by multiplying α with 1.3, and the results are shown in the “α” line and the “1.3” column of Table 1. As the value of each parameter increases or decreases, there is a certain performance deterioration.

Beside, we separately set the parameters α, β, γ and ω to 0 to validate the effectiveness of the four terms including instance-aware term S_i^{ins} , boundary-aware term S_i^{bou} , class-aware term S_i^{cla} and saliency-aware term S_i^{sal} in SGPO. The corresponding results are shown in the column “0″ of Table 1. From the comparison between the column “1(Ours)” and the column “0” in Table 1, we find that all the four terms are helpful to the performance. For example, on $mAP^r_{0.25}$, S_i^{ins}, S_i^{bou}, S_i^{cla} and S_i^{sal} bring 19.21% (26.89%→46.1%), 19.22% (26.88%→46.1%), 6.0% (40.1%→46.1%) and 13.5% (32.6%→46.1%) improvement, respectively.

From the comparison between the column “1(Ours)” and the column “0” in Table 1, we find that the instance-aware term S_i^{ins} and the boundary-aware term S_i^{bou} contribute more to the performance, when compared with the class-aware term S_i^{cla} and the saliency term S_i^{sal}. The instance-aware term S_i^{ins} indicates the main parts of object in the image. The boundary-aware term S_i^{bou} indicates the boundary of object in the image. In some cases, the boundary-aware term S_i^{bou} is more helpful to find the objects like bicycles, chairs and so on, due that the boundary provides more discriminative information to distinguish them. Since the two terms, S_i^{ins} and S_i^{bou}, are more important, we first adjust the two parameters, α and β, to achieve the best performance using only S_i^{ins} and S_i^{bou}. Then we add the class-aware term S_i^{cla} and the saliency-aware term S_i^{sal} to further improve the performance. The class-aware term S_i^{cla} can provide background information, and the saliency-aware term S_i^{sal} can provide clearer object boundary information. We exploit salient regions to distinguish foreground and background, and use the foreground information to enhance the object information in the instance-aware term

Table 2
Analysis of the maximum erasing stage number, T, in MSER on the PASCAL VOC 2012 validation set in percentage %.

T	$mAP^r_{0.25}\uparrow$	$mAP^r_{0.5}\uparrow$	$mAP^r_{0.75}\uparrow$
1	44.0	25.4	9.3
2 (Ours)	**46.1**	**27.3**	**10.4**
3	44.2	26.1	9.5
4	40.9	24.1	8.7

Table 3
Analysis of saliency detection methods on the PASCAL VOC 2012 validation set in percentage %.

Method	$mAP^r_{0.25}\uparrow$	$mAP^r_{0.5}\uparrow$	$mAP^r_{0.75}\uparrow$
UCF [39]	40.0	22.0	7.5
Amulet [40]	41.9	22.7	8.0
MLFI-MSFF [41]	41.4	22.4	7.9
R3Net [32] (Ours)	**46.1**	**27.3**	**10.4**

Table 4
Analysis of the threshold δ in MSER on the PASCAL VOC 2012 validation set in percentage %.

δ	$mAP^r_{0.25}\uparrow$	$mAP^r_{0.5}\uparrow$	$mAP^r_{0.75}\uparrow$
10	45.2	26.4	9.7
20	45.1	26.2	9.5
30 (Ours)	**46.1**	**27.3**	**10.4**
40	44.9	26.4	9.6
50	43.7	25.3	9.1
60	43.9	25.4	9.5

Table 5
Comparison of mean average precision (mAP) and ABO among different methods on the PASCAL VOC 2012 validation set in percentage %. The red is the best, the green is the second best and the blue is the third best.

Method	$mAP^r_{0.25}\uparrow$	$mAP^r_{0.5}\uparrow$	$mAP^r_{0.75}\uparrow$	ABO
CAM [14]	20.4	7.8	2.5	23.0
SPN [42]	26.4	12.7	4.4	27.1
MELM [43]	36.9	22.9	8.4	32.9
PRM [9]	44.3	26.8	9.0	37.6
IAM [10]	45.9	28.8	11.9	41.9
Ours	46.1	27.3	10.4	41.7

Table 6
Comparison of mean average precision (mAP) and ABO among different methods on the COCO validation set in percentage %. Bold is the best.

Method	$mAP^r_{0.25}\uparrow$	$mAP^r_{0.5}\uparrow$	$mAP^r_{0.75}\uparrow$	ABO
PRM [9]	5.8	2.2	0.5	16.4
Ours	**7.1**	**3.7**	**1.2**	**20.6**

S_i^{ins}. As above, we first adjust the two parameters, α and β, to achieve the best performance using only the instance-aware term S_i^{ins} and the boundary-aware term S_i^{bou}. Then we adjust the parameter ω to exploit the saliency-aware term S_i^{sal} to improve the performance. Finally, we add the class-aware term S_i^{cla} and adjust the parameter γ to further improve the performance. The codes of our method as well as parameter settings are available at https://github.com/jetshz/MSER-SGPO.

2. The influence of the maximum erasing stage number T. To illustrate the influence of the maximum erasing stage number, T, we evaluate our MSER method with different values of T, *i.e.*, 1, 2, 3, 4, and the results are shown in Table 2. When T is set to 2, the performance is better than that with the other values of T. Compared with $T = 1$ (*i.e.* without MSER), this demonstrates the effectiveness of our MSER method. When T further increases to 3 and 4, the erased regions become larger but the performance degrades. With the increase of iteration times, more and more object-related regions are erased. However, if we excessively repeat the iteration process, object-related regions could be totally erased and some object-irrelevant regions, *i.e.* false positive regions, will be introduced. These object-irrelevant regions will disturb the proposal ordering and do harm to the performance. Therefore, we set T to 2 in our MSER method.

3. The influence of saliency detection methods. To illustrate the influence of saliency detection methods, we choose several other saliency detection methods, including UCF [39], Amulet [40] and MLFI-MSFF [41], to replace R3Net [32] for generating the saliency maps. The results of using different saliency detection methods are shown in Table 3. It can be seen from Table 3 that using the saliency detection method R3Net in our method achieves the best performance.

4. The influence of the threshold δ in MSER. To illustrate the influence of the threshold δ in MSER, we evaluate our method with different values of δ, *i.e.* 10, 20, …, 60, and the results are shown in Table 4. It can be seen from Table 4 that the best performance is achieved when the threshold δ is set to 30. A higher value of δ results in fewer erased regions, and thus the contribution of MSER is weakened. Although a lower value of δ results in more erased regions, some object-irrelevant regions are also introduced in the process of MSER. These object-irrelevant regions will disturb the proposal ordering and do harm to the performance.

4.3. Performance evaluation and comparison

For a fair comparison, we quantitatively compare our method against previous four weakly supervised instance segmentation methods including CAM [14], SPN [42], MELM [43] and PRM [9] to predict the instance masks. Notably, due to the lack of publicly available codes of CAM, SPN and MELM, the performance data of the three methods on the PASCAL VOC 2012 dataset [15] are borrowed from [9], and for the PRM method, we finetuned the parameters with the publicly available code of PRM provided by the authors. Due to the lack of publicly available codes of CAM, SPN and MELM, we only report the experimental results on the COCO dataset of our method and the PRM method. As shown in Table 5, we can see that our method achieves the competitive instance segmentation performance in terms of all the four evaluation metrics. In terms of $mAP^r_{0.25}$, our method achieves the best pe rformance (*i.e.* 46.1%). In terms of the other three evaluation metrics, our method ranks the 2nd (*i.e.* $mAP^r_{0.5}$: 27.3%, $mAP^r_{0.75}$: 10.4%,*ABO*: 41.7%), slightly lower than the best method, IAM, on the PASCAL VOC 2012 dataset. As shown in Table 6, the performance of our method on the COCO dataset is better than that of the PRM method in terms of all the four evaluation metrics, but lower than the performance achieved on the PASCAL VOC 2012 dataset. Compared with the PASCAL VOC 2012 dataset, the COCO dataset contains more object instances per image, larger scale variations among different object instances, and more various apperances of instances in the same object category, and thus the COCO dataset is more challenging for weakly supervised instance segmentation.

In Fig. 5, we show the segmentation results of the PRM method and our segmentation results on some example images which contain multiple objects and complex background. The first three examples indicate the complementarity between saliency map and EPRM. Although the regions with the high saliency values may be parts of object instances, the saliency map contributes to the distinction between object regions and background regions, and helps to retrieve the instance mask with accurate boundaries. The next three examples show some complex scenes with overlapping objects. In these examples, the saliency maps may highlight a large region containing multiple objects or highlight separate object parts. In the 4th row, although the saliency map fails to highlight the whole objects, it helps to identify the two different instances in the image. In the 5th and 6th row, although the saliency maps

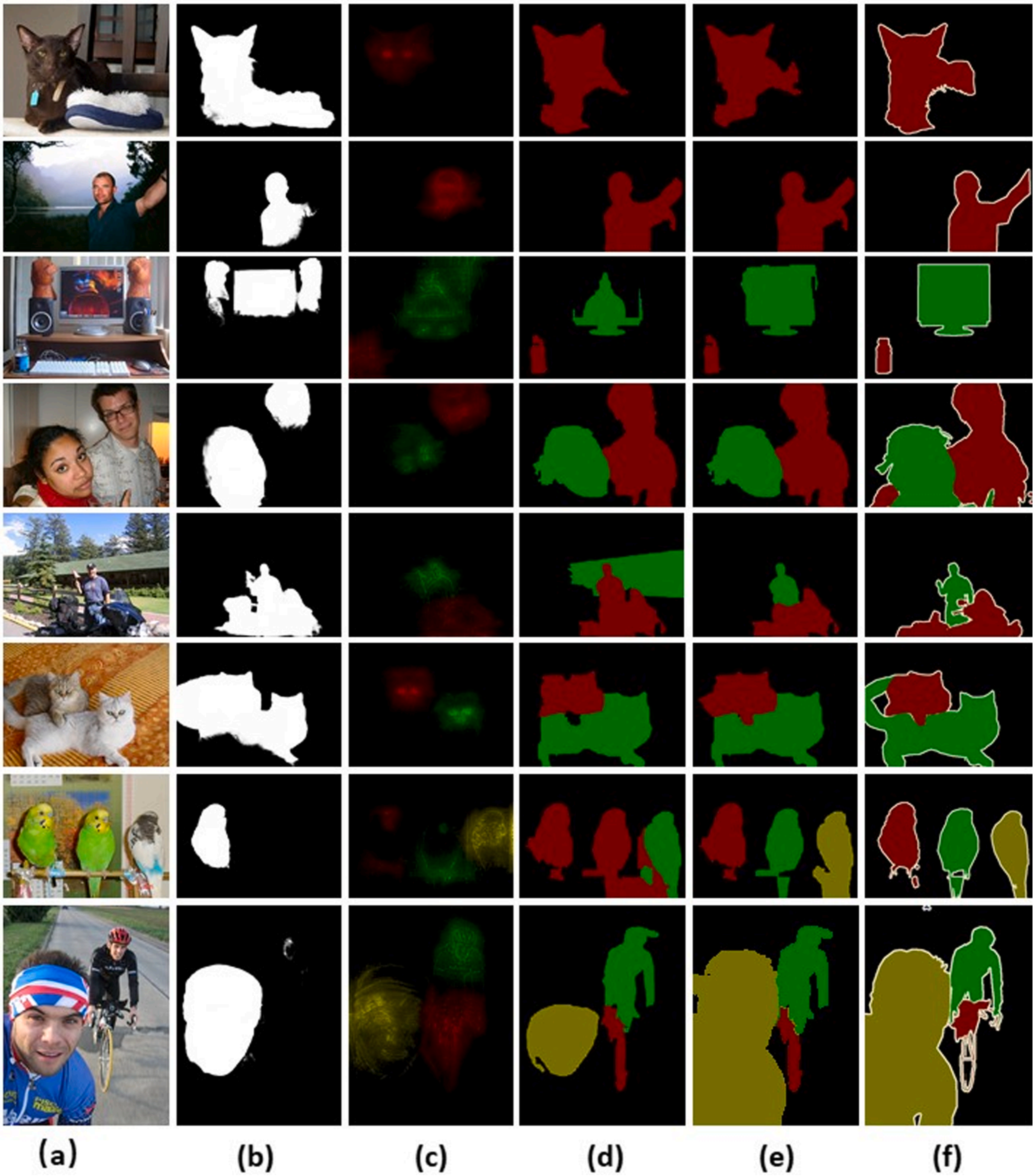

Fig. 5. Weakly supervised instance segmentation examples of our framework. (a) Input image, (b) saliency map, (c) EPRM, segmentation results of (d) PRM [9] and (e) our method, and (f) ground truth.

cannot distinguish different objects, they help to alleviate the adverse effect of background regions. The last two examples (the 7th and 8th row) indicate the situation that the image contains several object instances while the saliency map only highlights one of them. In such a situation, our MSER method can search for more object-related regions of all instances, and helps to generate the accurate instance masks.

Besides, we also report the time complexity of our method and the PRM method tested on a PC with a NVIDIA TITAN X GPU. The running time of our method to process a 448 × 448 image is 1.52 s, while the PRM method takes 1.20 s to process a 448 × 448 image. Our method shows a comparable inference speed.

Table 7
Ablation study of the proposed framework on the PASCAL VOC 2012 validation set in percentage %.

Model	$mAP^r_{0.25}\uparrow$	$mAP^r_{0.5}\uparrow$	$mAP^r_{0.75}\uparrow$	ABO
PRM	43.9	25.1	8.8	40.9
PRM + MSER	44.9	25.7	9.0	40.5
PRM + SGPO	44.0	25.4	9.3	41.1
PRM + MSER + SGPO	**46.1**	**27.3**	**10.4**	**41.7**

4.4. Ablation study

In this section, we present a more detailed examination of our framework on the PASCAL VOC 2012 validation set. To investigate the individual contributions of MSER method and SGPO method, we change one component each time. From the ablation study results shown in Table 7, we can clearly observe that both MSER and SGPO contribute to the better instance segmentation performance.

Specifically, to validate the contribution of MSER, we delete the SGPO from our framework and obtain a variant to segment the images, named PRM + MSER. From the comparison between PRM and PRM + MSER, we can find out our MSER method helps to achieve the higher performance (*e.g.* $mAP^r_{0.25}$: 43.9%→44.9%, $mAP^r_{0.5}$: 25.1%→25.7% and $mAP^r_{0.75}$: 8.8%→9.0%), due that the MSER method is able to find more object-related regions, which result in a more complete representation of object instances.

To investigate the contribution of SGPO, we remove the MSER from our framework and use the initial PRM for instance segmentation, called PRM + SGPO. From the comparison between PRM and PRM + SGPO, we observe that our SGPO method is helpful to achieve the better segmentation results (*e.g.* $mAP^r_{0.25}$: 43.9%→44.0%, $mAP^r_{0.5}$: 25.1%→25.4% and $mAP^r_{0.75}$: 8.8%→9.3%), due that the SGPO method can alleviate background noise when ordering the object proposals and can help to retrieve more accurate object proposals.

To validate the performance of combining MSER and SGPO, we compare our complete framework, *i.e.* PRM + MSER + SGPO with the other three variants. We can observe from Table 7 that our complete framework, which inherits the advantages of both MSER and SGPO, outperforming any single method, *i.e.* PRM + MSER or PRM + SGPO, on all the three metrics.

5. Conclusion

In this paper, we propose an effective weakly supervised instance segmentation framework using image-level annotations. In this framework, we propose a multi-stage erasing refinement method and a saliency-guided proposals ordering method. The former method trains multiple networks with iteratively erased images to discover new object-related regions to form the enhanced instance representation with more detailed shape and location information of object instances. The latter method introduces saliency map to emphasize pixels in salient regions to better rank object proposals for instance segmentation. Experimental results demonstrate that our framework with the two proposed methods effectively improves the instance segmentation performance.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grants 61771301 and 61906121.

References

[1] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: Proc. ICLR, San Diego, CA, USA, 2015, pp. 1–14.

[2] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proc. IEEE CVPR, Las Vegas, NV, USA, 2016, pp. 770–778.

[3] J. Dai, K. He, J. Sun, Convolutional feature masking for joint object and stuff segmentation, in: Proc. IEEE CVPR, Boston, MA, USA, 2015, pp. 3992–4000.

[4] Y. Li, H. Qi, J. Dai, X. Ji, Y. Wei, Fully convolutional instance-aware semantic segmentation, in: Proc. IEEE CVPR, Honolulu, HI, USA, 2017, pp. 2359–2367.

[5] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask R-CNN, in: Proc. IEEE CVPR, Honolulu, HI, USA, 2017, pp. 2359–2367.

[6] L.-C. Chen, A. Hermans, G. Papandreou, F. Schroff, P. Wang, H. Adam, Masklab: Instance segmentation by refining object detection with semantic and direction features, in: Proc. IEEE CVPR, Salt Lake City, UT, USA, 2018, pp. 4013–4022.

[7] S. Liu, L. Qi, H. Qin, J. Shi, J. Jia, Path aggregation network for instance segmentation, in: Proc. IEEE CVPR, Salt Lake City, UT, USA, 2018, pp. 8759–8768.

[8] K. Chen, J. Pang, J. Wang, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Shi, W. Ouyang, C. Loy, D. Lin, Hybrid task cascade for instance segmentation, in: Proc. IEEE CVPR, Long Beach, CA, USA, 2019, pp. 4974–4983.

[9] Y. Zhou, Y. Zhu, Q. Ye, Q. Qiu, J. Jiao, Weakly supervised instance segmentation using class peak response, in: Proc. IEEE CVPR, Salt Lake City, UT, USA, 2018, pp. 3791–3800.

[10] Y. Zhu, Y. Zhou, H. Xu, Q. Ye, D. Doermann, J. Jiao, Learning Instance Activation Maps for Weakly Supervised Instance Segmentation, in: Proc. IEEE CVPR, Long Beach, CA, USA, 2019, pp. 3116–3125.

[11] H. Cholakkal, G. Sun, F.S. Khan, L. Shao, Object counting and instance segmentation with image-level supervision, in: Proc. IEEE CVPR, Long Beach, CA, USA, 2019, pp. 12397–12405.

[12] W. Ge, S. Guo, W. Huang, M.R. Scott, Label-PEnet: Sequential label propagation and enhancement networks for weakly supervised instance segmentation, in: Proc. IEEE CVPR, Long Beach, CA, USA, 2019, pp. 3345–3354.

[13] F. Meng, K. Luo, H. Li, Q. Wu, X. Xu, Weakly supervised semantic segmentation by a class-level multiple group cosegmentation and foreground fusion strategy, IEEE Trans. Circ. Syst. Video Technol. (2019), https://doi.org/10.1109/TCSVT.2019.2962073.

[14] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, Learning deep features for discriminative localization, in: Proc. IEEE CVPR, Las Vegas, NV, USA, 2016, pp. 2921–2929.

[15] M. Everingham, S.M.A. Eslami, L.V. Gool, C.K.I. Williams, J. Winn, A. Zisserman, The pascal visual object classes challenge: A retrospective, Int. J. Comput. Vis. 111 (1) (2015) 98–136.

[16] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, C.O.C.O. Microsoft, Common objects in context, in: Proc. ECCV, Zurich, Switzerland, 2014, pp. 740–755.

[17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, F.F. Li, Imagenet: A large-scale hierarchical image database, in: Proc. IEEE CVPR, Miami Beach, FL, USA, 2009, pp. 248–255.

[18] I.H. Laradji, N. Rostamzadeh, P.O. Pinheiro, D. Vázquez, M. Schmidt, Instance segmentation with point supervision, arXiv preprint arXiv:1906.06392, 2019.

[19] A. Khoreva, R. Benenson, J. Hosang, M. Hein, B. Schiele, Simple does it: Weakly supervised instance and semantic segmentation, in: Proc. IEEE CVPR, Honolulu, HI, USA, 2017, pp. 876–885.

[20] R. Girshick, Fast R-CNN, in: Proc. IEEE ICCV, Santiago, Chile, 2015, pp. 1440–1448.

[21] D. Lin, J. Dai, J. Jia, K. He, J. Sun, Scribblesup: Scribble-supervised convolutional networks for semantic segmentation, in: Proc. IEEE CVPR, Las Vegas, NV, USA, 2016, pp. 3159–3167.

[22] A. Kolesnikov, C.H. Lampert, Seed, expand and constrain: Three principles for weakly-supervised image segmentation, in: Proc. ECCV, Amsterdam, The Netherlands, 2016, pp. 695–711.

[23] Y. Wei, X. Liang, Y. Chen, Z. Jie, Y. Xiao, Y. Zhao, S. Yan, Learning to segment with image-level annotations, Pattern Recognit. 59 (2016) 234–244.

[24] Y. Wei, X. Liang, Y. Chen, X. Shen, Stc: A simple to complex framework for weakly-supervised semantic segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 39 (11) (2016) 2314–2320.

[25] B. Jin, M.V.O. Segovia, S. Susstrunk, Webly supervised semantic segmentation, in: Proc. IEEE CVPR, Honolulu, HI, USA, 2017, pp. 3626–3635.

[26] J. Ahn, S. Kwak, Learning pixel-level semantic affinity with image-level supervision for weakly supervised semantic segmentation, in: Proc. IEEE CVPR, Salt Lake City, UT, USA, 2018, pp. 4981–4990.

[27] Z. Huang, X. Wang, J. Wang, W. Liu, Weakly-supervised semantic segmentation network with deep seeded region growing, in: Proc. IEEE CVPR, Salt Lake City, UT, USA, 2018, pp. 7014–7023.

[28] Y. Wei, H. Xiao, H. Shi, Z. Jie, J. Feng, Revisiting dilated convolution: A simple approach for weakly-and semi-supervised semantic segmentation, in: Proc. IEEE CVPR, Salt Lake City, UT, USA, 2018, pp. 7268–7277.

[29] X. Zhang, Y. Wei, J. Feng, Y. Yang, Adversarial complementary learning for weakly supervised object localization, in: Proc. IEEE CVPR, Salt Lake City, UT, USA, 2018, pp. 1325–1334.

[30] Y. Wei, J. Feng, X. Liang, M.M. Chen, Object region mining with adversarial erasing: A simple classification to semantic segmentation approach, in: Proc. IEEE CVPR, Honolulu, HI, USA, 2017, pp. 1568–1576.

[31] K. Li, Z. Wu, K. Peng, J. Ernst, Y. Fu, Tell me where to look: Guided attention inference network, in: Proc. IEEE CVPR, Salt Lake City, UT, USA, 2018, pp. 9215–9223.

[32] Z. Deng, X. Hu, L. Zhu, X. Xu, J. Qin, G. Han, P.-A. Heng, R3Net, Recurrent residual refinement network for saliency detection, in: Proc. AAAI, New Orleans, LA, USA, 2018, pp. 684–690.

[33] K.-K. Maninis, J. Pont-Tuset, P. Arbeláez, L.V. Gool, Convolutional oriented boundaries: From image segmentation to high-level tasks, IEEE Trans. Pattern Anal. Mach. Intell. 40 (4) (2017) 819–833.

[34] L. Bottou, Large-scale machine learning with stochastic gradient descent, in: Proc. COMPSTAT, Paris, France, 2010, pp. 177–186.

[35] F. Zhou, Y. Lin, Fine-grained image classification by exploring bipartite-graph labels, in: Proc. IEEE CVPR, Las Vegas, NV, USA, 2016, pp. 1124–1133.

[36] J. Redmon, S. Divval, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: Proc. IEEE CVPR, Las Vegas, NV, USA, 2016, pp. 779–788.

[37] L. Ye, Z. Liu, Y. Wang, Learning semantic segmentation with diverse supervision, in: Proc. IEEE WACV, Lake Tahoe, USA, 2018, pp. 1461–1469.

[38] J. Pont-Tuset, L. Van Gool, Boosting object proposals: From pascal to coco, in: Proc. IEEE ICCV, Santiago, Chile, 2015, pp. 1546–1554.

[39] P. Zhang, D. Wang, H. Lu, H. Wang, B. Yin, Learning uncertain convolutional features for accurate saliency detection, in: Proc. IEEE ICCV, Venice, Italy, 2017, pp. 212–221.

[40] P. Zhang, D. Wang, H. Lu, H. Wang, X. Ruan, Amulet: Aggregating multi-level convolutional features for salient object detection, in: Proc. IEEE ICCV, Venice, Italy, 2017, pp. 202–211.

[41] M. Huang, Z. Liu, L. Ye, X. Zhou, Y. Wang, Saliency detection via multi-level integration and multi-scale fusion neural networks, Neurocomputing 364 (2019) 310–321.

[42] Y. Zhu, Y. Zhou, Q. Ye, Q. Qiu, J. Jiao, Soft proposal networks for weakly supervised object localization, in: Proc. IEEE CVPR, Honolulu, HI, USA, 2017, pp. 1841–1850.

[43] F. Wan, P. Wei, J. Jiao, Z. Han, Q. Ye, Min-entropy latent model for weakly supervised object detection, in: Proc. IEEE CVPR, Salt Lake City, UT, USA, 2018, pp. 1297–1306.