Computer Vision and Image Understanding 207 (2021) 103207

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

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

Instance-level salient object segmentation

Guanbin Li [a,*], Pengxiang Yan [a], Yuan Xie [b], Guisheng Wang [d], Liang Lin [a], Yizhou Yu [c]

[a] School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, 510006, China
[b] DarkMatter AI Research, China
[c] Deepwise AI Lab, China
[d] Department of Radiology, The Third Medical Centre, Chinese PLA General Hospital, Beijing, China

ARTICLE INFO

Communicated by Nikos Paragios

MSC:
41A05
41A10
65D05
65D17

Keywords:
Salient object detection
Salient object contour detection
Salient instance segmentation
Multiscale refinement network

ABSTRACT

Image saliency detection has recently achieved great success due to the development of deep convolutional neural networks. However, most of the existing salient object detection methods cannot identify individual object instances in the detected salient region. In this paper, we present a salient instance segmentation method that produces a saliency map with distinct object instance labels for an input image. Our method consists of three primary steps, *i.e.*, salient region inference, salient object contours detection, and salient object instances identification. For the first two steps, we propose a multiscale saliency refinement network, which generates high-quality salient region masks and salient object contours. For the last step, we propose a morphology algorithm that incorporates detected salient regions and salient object contours to generate promising salient object instance segmentation results. To promote further research and evaluation of salient instance segmentation, we also construct a new database (ILSO-2K) of 2,000 images with pixel-wise salient instance annotations. Experimental results demonstrate that our proposed method is capable of achieving satisfactory performance over six public benchmarks for salient region detection as well as on our new dataset for salient instance segmentation. The source code and proposed dataset will be public available at https://github.com/Kinpzz/MSRNet-CVIU.

1. Introduction

Salient object detection aims at locating the most noticeable and visually distinctive object regions in images and segmenting them out from the background. Since the results of salient object detection can reflect the relative importance of visual contents in an image, they can be used to narrow the scope of visual processing and lower computational cost. As a result, it usually serves as a pre-processing step and has been applied to a variety of computer vision applications to improve their performance. These applications include action recognition (Rutishauser et al., 2004), video summarization (Ma et al., 2005), object detection (Navalpakkam and Itti, 2006), robotic perception (Sugano et al., 2010), visual tracking (Wu et al., 2014), image retrieval (Gao et al., 2015), semantic segmentation (Wei et al., 2017), saliency-aware video segmentation (Wang et al., 2017c), photo cropping (Wang et al., 2018a), *etc.*

In recent years, with the development of deep convolutional neural networks, the performance of salient object detection has been improved by a large margin (Liu and Han, 2016; Hou et al., 2017; Deng et al., 2018; Wang et al., 2018b; Wu et al., 2019a). Nevertheless, for an input image, most of the existing methods are only designed to detect pixels that belong to any salient object, *i.e.*, a dense saliency map, but unaware of which object instance these pixels belong to. According to Zhang et al. (2016), we refer to the task performed by these methods "salient region detection". As shown in Fig. 1, we address a more challenging task, instance-level salient object segmentation (or *salient instance segmentation* for short), which aims to identify individual object instances in the detected salient regions. To achieve this goal, the next generation of salient object detection methods need to perform more detailed parsing within the detected salient regions, which is of great significance for practical applications, such as multi-label image recognition (Wei et al., 2016), image captioning (Karpathy and Fei-Fei, 2015), various weakly supervised or unsupervised learning scenarios (Lai and Gong, 2016; Chen and Gupta, 2015).

In this paper, we propose to decompose the salient instance segmentation task into three sub-tasks as follows. (1) Estimating a binary saliency map. In this sub-task, a pixel-wise saliency mask is predicted to indicate the detected salient regions in the input image. (2) Detecting salient object contours. In this sub-task, we perform contour detection for individual salient object instances. Different from traditional edge detection (Movahedi and Elder, 2010), such contour detection is expected to highlight the salient object contours but suppress edges that do not belong to boundaries of any salient object instances. (3)

* Corresponding author.
E-mail address: liguanbin@mail.sysu.edu.cn (G. Li).

https://doi.org/10.1016/j.cviu.2021.103207
Received 18 June 2020; Received in revised form 26 March 2021; Accepted 27 March 2021
Available online 30 March 2021

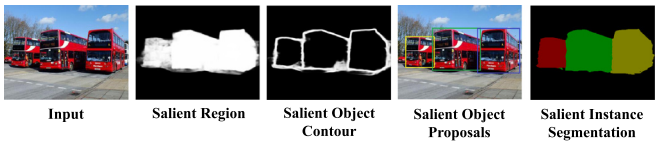

Fig. 1. An example of instance-level salient object segmentation. Left: input image. Middle left: detected salient region. Middle: detected salient object contour. Middle right: salient object proposals. Right: the result of salient instance segmentation. Different colors indicate different object instances in the detected salient region. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Identifying salient object instances. In this sub-task, we incorporate detected salient region and salient object contours to generate salient object instances. Finally, a further refinement based on CRF (Krähenbühl and Koltun, 2011) is employed to improve the spatial coherence of salient instance proposals and generate an instance-level salient object segmentation map.

Several recent salient region detection methods (Li and Yu, 2016; Liu and Han, 2016; Wang et al., 2016a) are based on fully convolutional neural networks (FCNs) as FCNs can be trained end-to-end with high-efficiency to produce accurate results. However, they still have their own limitations. Most of these methods learn to infer saliency by exploring the features of multilevel convolutional layers (Wang et al., 2016a; Liu and Han, 2016). As their results are derived from receptive fields of a uniform size, they may not perform well on detecting salient objects at multiple scales. Though Li and Yu (2016) proposed to combine a multiscale FCN and a segment-level spatial pooling stream to compensate for this deficiency, their iterative training process is complex and time consuming since the end-to-end training only covers the first stream but not the second one. Moreover, the resolution of their final saliency map is only 1/8 of the original input resolution, making it infeasible to detect small salient objects accurately or be applied to salient object contour detection.

Given the sub-tasks as mentioned above for salient instance segmentation, we propose a deep multiscale saliency refinement network, which can generate promising results for both salient region detection and salient object contour detection. Specifically, our deep network consists of three major components, including a bottom-up backbone network for feature extraction, an ASPP (Chen et al., 2017) module with attentional weights for multiscale feature fusion, and a top-down stream for feature refinement. It is designed to integrate the low-level information from the bottom-up network and high-level information from the top-down stream. The high-level information contains more semantic knowledge but lacks spatial details, while the low-level information retains more spatial details with higher resolution. Therefore, such information integration is of great benefit for pixel-wise segmentation tasks including both salient region detection and salient object contour detection.

Given the detected salient regions, we incorporate the detected contours of salient object instances to generate a number of salient instance proposals. Although the detected salient regions and salient object contours are of high quality, the generated salient instance proposals are still noisy. We then filter out these noisy proposals and produce a compact set of segmented salient object instances. Finally, a fully connected CRF model (Krähenbühl and Koltun, 2011) is further employed to improve spatial coherence in the initial salient instance segmentation.

In summary, this paper has the following contributions:

- We introduce a fully convolutional multiscale refinement network (MSRNet), for salient region detection, which can not only integrate bottom-up and top-down information for high-precision saliency inference but also attentionally combine multiscale features to discover salient object at multiple scales. Experimental results demonstrate that the proposed network can achieve satisfactory performance on salient region detection without any pre-/post-processing.
- MSRNet is well applicable to salient object contour detection, making it possible to separate individual salient object instances in detected salient regions. When further incorporated with CRF-based refinement, our method can generate salient instance maps of high-quality.
- We create a new challenging dataset with pixel-wise salient instance annotations for further research and evaluation of salient instance segmentation. Benchmark results for salient contour detection and salient instance segmentation are both provided using a framework based on MSRNet.

This paper is an extended version of Li et al. (2017), it provides a reformative framework based on a multiscale refinement network for salient instance segmentation including a more complete introduction and analysis. Specifically, for multiscale refinement network, we propose to exploit the multiscale information in feature level and attentively combine the multiscale features instead of using multiple input sizes in duplicated refinement networks. For salient instance proposal, we propose to utilize both detected salient regions and salient object contours to better generate salient instance proposals instead of using only salient object contour. Moreover, we have extended the scale of the dataset for salient instance segmentation by introducing more challenging images with pixel-wise instance saliency annotations. Experimental results show that the redesigned framework achieves superior performance with faster speed on salient region detection, salient object contour detection, and salient instance segmentation.

The remainder of this paper is organized as follows. Section 2 reviews the works that are most relevant to our proposed method. Section 3 introduces the framework for instance-level salient object segmentation, including our proposed MSRNet. Section 4 presents the construction of the new dataset (ILSO-2K) for salient instance segmentation. Extensive experimental comparisons are presented in Section 5. Finally, Section 6 makes a conclusion to this paper.

2. Related work

In this section, we mainly focus on discussing works on the topics that are most relevant to our proposed method.

2.1. Salient region detection

Conventional salient region detection methods generally rely on various hand-crafted features, such as contrast (Cheng et al., 2015), background (Wang et al., 2016b), center prior (Klein and Frintrop, 2011), and so on (Liu et al., 2011; Goferman et al., 2012; Li et al., 2014; Huo et al., 2017).

Recently, the development of deep convolutional neural networks (DCNNs) has brought tremendous improvement to many vision tasks, including salient region detection (Li and Yu, 2015; Zhao et al., 2015; Lee et al., 2016; Li and Yu, 2016; Liu and Han, 2016; Wang et al., 2016a). In particular, these methods based on fully convolutional neural network (FCN) (Zhang et al., 2017a,b; Wang et al., 2017a; Hou et al., 2017; Chen et al., 2018; Deng et al., 2018; Zhang et al., 2018; Wang et al., 2018b, 2019b; Wu et al., 2019a; Zeng et al., 2019; Wang et al., 2019c) has become the dominant methods in this field due to the end-to-end trainable capability and high computational efficiency of FCN. Liu and Han (2016) proposed an end-to-end deep hierarchical saliency network to generate a coarse saliency map and a hierarchical recurrent convolutional neural network for further refinement. It does not rely on image segmentation and can produce a saliency map by directly feedforwarding testing images through the network. Wang et al. (2016a) proposed another end-to-end recurrent fully convolutional network for salient region detection. Their deep network incorporates saliency prior knowledge for more accurate inference through iterative

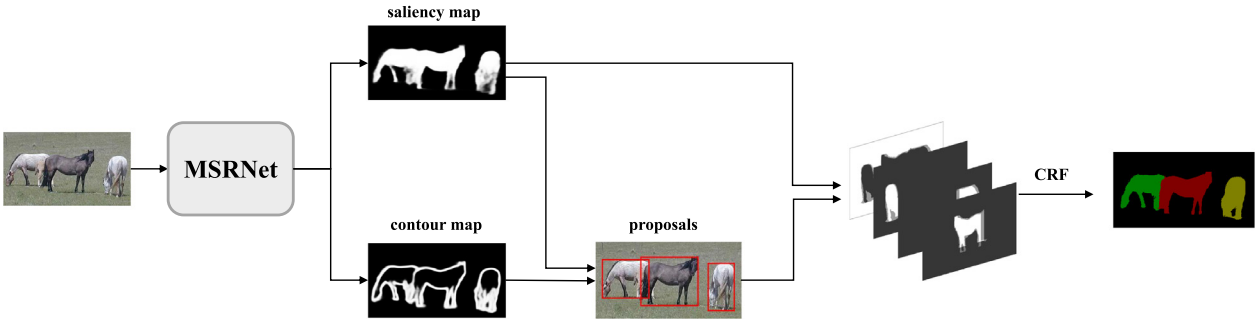

Fig. 2. Our overall framework for instance-level salient object segmentation. It consists of a multiscale refinement network (MSRNet) for salient region detection and salient object contour detection, a morphology algorithm for salient instance proposal generation, and a CRF-based refinement for salient instance segmentation.

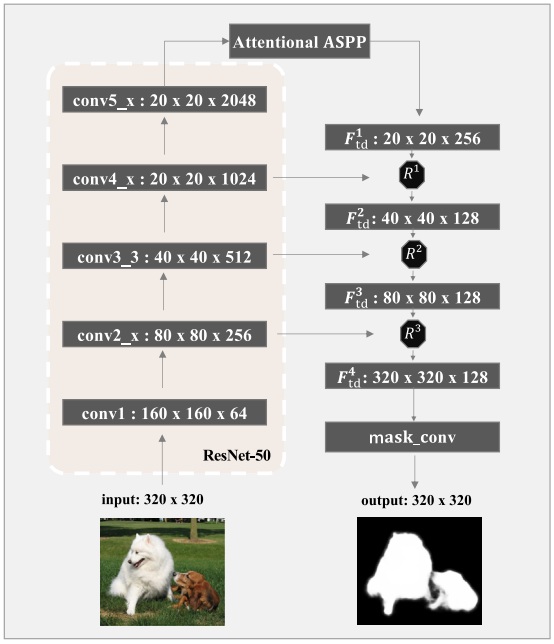

Fig. 3. The architecture of our multiscale refinement network (MSRNet) for salient region detection and salient object contour detection.

refinement with a recurrent network structure. Hou et al. (2017) introduced short connections between the deeper and shallower side-output layers of FCN-based salient region detection networks. This architecture takes full advantage of multilevel and multiscale features and thus helps better locate the most salient region. Wang et al. (2018b) proposed a novel localization-to-refinement architecture for salient region detection, which consists of a global recurrent localization network to locate salient objects and a local boundary refinement network to refine the salient region by the spatial relationships between each pixel and its neighbors. Wang et al. (2019b) proposed a unified salient region detection framework that integrates both top-down and bottom-up saliency inference in an iterative and cooperative manner. Wang et al. (2019c) proposed to learn salient object detection from eye fixations through a novel attentive saliency network. Wei et al. (2020) proposed a label decoupling network to decompose the original saliency maps into body maps and detail maps and they were further used to supervise the learning of body and detail futures of salient objects. By fusing the body and detail futures, the proposed network can generate precise saliency maps.

More recently, some researchers proposed better solutions to integrate edge detection into the unified framework of salient object detection to assist the generation of saliency maps (Qin et al., 2019; Liu et al., 2019a; Wu et al., 2019a; Feng et al., 2019; Wu et al., 2019b; Zhao et al., 2019; Wang et al., 2019d). Qin et al. (2019) designed a hybrid loss for boundary-aware salient object detection on pixel-level, patch-level, and map-level. Liu et al. (2019a) proposed a simple pooling-based module and feature aggregation module for jointly training salient object detection with standard edge detection. Feng et al. (2019) proposed the attentive feedback modules to better explore the structure of salient objects and a boundary-enhanced loss to further learn exquisite object boundaries. Wu et al. (2019b) proposed a stacking cross refinement unit to simultaneously refine multi-level features of salient object detection and edge detection. Wang et al. (2019d) proposed to learn saliency from multiscale information by combining a pyramid attention module and a salient edge detection module. Zhao et al. (2019) proposed a edge-guided network to model the complementarity between salient edge information and salient object information. Since the better integration of edge information, these methods usually show better performance on salient object detection, especially around salient object boundaries. Moreover, a more comprehensive survey about those DCNN-based salient object detection methods can be found in Wang et al. (2019a).

2.2. Instance-aware semantic segmentation

Instance-aware semantic segmentation is a challenging problem related to both object detection and semantic segmentation. It requires not only to correctly detect all objects in an image but also to segment each instance accurately. This problem was first raised by Hariharan et al. (2014) and has been extensively studied in recent years. Overall, it can be solved in an end-to-end integrated model (Romera-Paredes and Torr, 2016; Dai et al., 2016a; Wang et al., 2020) or formulated as a multi-task learning problem incorporating both object detection and semantic segmentation (Dai et al., 2016b; He et al., 2017; Chen et al., 2020).

Inspired by this problem, we were the first to propose the new task, salient instance segmentation, in our preliminary work (Li et al., 2017). Salient instance segmentation aims at simultaneously detecting the most salient regions and identifying each salient instance inside them. It is a more generic but meanwhile more challenging problem compared with instance-aware semantic segmentation since salient objects are not associated with any predefined set of semantic categories. We believe solutions to such generic problems are more valuable in practice as it is infeasible to enumerate all object categories and prepare enough pixel-wise training data for each category.

2.3. Relation to previous methods

Different from those edge-assisted SOD methods mentioned in Section 2.1, our redesigned MSRNet is a refined version of our conference MRNet (Li et al., 2017), which aims at promoting the saliency details by introducing the low-level features into high-level ones via a novel refinement architecture. MSRNet is well applicable for salient object contour detection but it does not directly integrate the edge information

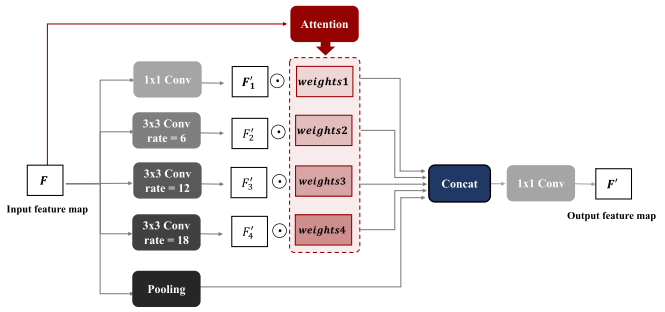

Fig. 4. The architecture of ASPP with attention module.

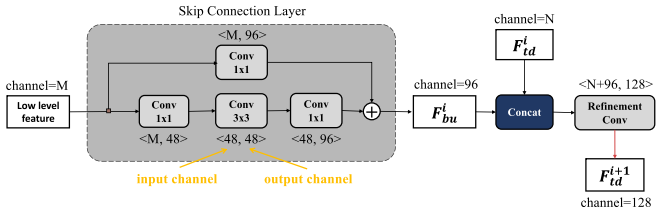

Fig. 5. The architecture of our refinement module.

to assist the salient region detection. The detected contours of salient instances are mainly designed to assist the salient instance segmentation part.

Moreover, since we proposed the salient instance segmentation task in our preliminary work (Li et al., 2017), several solutions (Fan et al., 2019b; Pei et al., 2020) have been proposed to solve this challenging task. (Fan et al., 2019b) proposed a single-stage salient framework based on a detection network with a novel segmentation branch, which further considers the context of bounding boxes. Pei et al. (2020) proposed a multitask network to simultaneously perform salient object subitizing (Zhang et al., 2015) and salient region detection. Then a clustering algorithm was further designed to segment the detected salient regions into salient instances based on subitizing. Fan et al. (2019a) proposed an instance-aware video salient object detection datasets with real human eye-fixation data, which further promoted the development of instance-level salient object detection.

3. Salient instance segmentation

In this section, we introduce the overall framework for instance-level salient object segmentation and then elaborate on the details of each component of the framework.

3.1. Overall framework

As shown in Fig. 2, the architecture of our framework for salient instance segmentation consists of four cascaded components, including salient region detection, salient object contour detection, salient instance proposal generation, and salient instance refinement. Specifically, we first propose a FCN-based multiscale refinement network and apply it for both salient region and contour detection (Section 3.2). Based on the detected salient regions and contours, we then generate salient instance proposals using a morphology algorithm (Section 3.3). Finally, we integrate the output of the previous three steps into a CRF (Krähenbühl and Koltun, 2011) model to generate the final salient instance segmentation (Section 3.4).

3.2. Multiscale refinement network

We formulate both salient region detection and salient object contour detection as a binary segmentation problem and calculate the probability of salient region/contour for each pixel. FCNs have been widely used in pixel-wise segmentation problems and have achieved great success in salient region detection (Li and Yu, 2016; Hou et al., 2017; Wang et al., 2018b, 2019b) and object contour detection (Xie and Tu, 2015; Yang et al., 2016; Liu et al., 2017). However, when we first proposed salient instance segmentation in our preliminary work (Li et al., 2017), none of the existing methods had attempted to address these two problems in a unified network at that time. Since salient objects have different scales, we propose a multiscale refinement network (MSRNet) for both salient region detection and salient object contour detection. In the earlier version of this paper, we proposed to use three parallel networks with shared weights to extract features from images with different input scales. Although this scheme can effectively extract features from different scales, it trebles the computational cost. In this paper, we propose a more efficient and effective method to detect salient objects at multiple scales. As shown in Fig. 3, we first adopt a backbone network (Section 3.2.1) with an attentional atrous spatial pyramid pooling (ASPP) (Chen et al., 2017) module (Section 3.2.2) to capture multiscale features by resampling convolutional features extracted at a single scale into multiple scales. Moreover, we use a top-down refinement stream to progressively recover the spatial resolution from the bottom-up backbone network (Section 3.2.3).

3.2.1. Bottom-up backbone network

As shown in Fig. 3, we modify the ResNet-50 (He et al., 2016) into a fully convolutional network, which serves as our bottom-up backbone network for feature extraction. Specifically, we use the first five groups of layers of ResNet-50 (*i.e.*, conv1, conv2_x, ..., conv5_x) leaving out the following average pooling layer and fully connected layer. Moreover, we modify the stride of the convolutional layers in conv5_x to $(1, 1)$ to make the bottom-up feature map denser and set its dilation rate to 2 to retain the original receptive field of the filters. Thus, the bottom-up backbone network extracts high-level features with a $1/16$ resolution of the original input image.

3.2.2. Multiscale feature fusion with attentional weights

As ASPP (Chen et al., 2017) is a practical module to capture multiscale information using parallel atrous convolutional layers with different dilation rates, we attach an ASPP on the top of our bottom-up backbone network to capture and fuse the features extracted with different field-of-views using attentional weights. The ASPP module consists of four parallel components, including a 1×1 convolutional layer, and three 3×3 atrous convolutional layers with $rate = \{6, 12, 18\}$, and a global average pooling layer. Element-wise multiplication is further performed between the output feature map of each convolutional layer in ASPP and its attentional weights since attentional weights can reflect how much attention should be paid to features at different spatial locations and different scales.

As shown in Fig. 4, the attention module takes as input the output feature map of the bottom-up backbone network. The attention module consists of four parallel 3×3 convolutional layers with 256 channels, each of which generates attentional weights for each parallel convolutional layer in ASPP. Therefore, the attention module learns a soft weight for each spatial location and each scale of features. Finally, the output feature maps at different scales are concatenated and fed into a 1×1 convolutional layer with 256 channels to fuse feature at multiple scales and produce a new feature map F_{td}^{1} for the top-down refinement stream shown in Fig. 3.

3.2.3. Top-down refinement stream

Although the bottom-up backbone network can effectively extract high-level features and the multiscale feature fusion can efficiently exploit the features of salient objects at multiple scales, these processes also bring spatial information loss for input images, which is harmful to pixel-wise segmentation tasks, such as the considered salient region

detection and salient object contour detection. To compensate for the loss of spatial information in the bottom-up process, high-level segmentation features need to be passed from the top layers and further integrate with low-level cues, such as colors and textures, to restore the resolution of output saliency maps for both salient region and salient object contour detection.

Inspired by Pinheiro et al. (2016), we propose a refinement module R to combine the high-level features with the low-level cues connected from the bottom-up stream and increase the resolution of the high-level feature map when necessary. As shown in Fig. 3, the refinement stream consists of three stacked refinement modules, which are respectively connected to conv2_x, conv3_x, and conv4_x through a skip connection layer. As shown in Fig. 5, each refinement module R^i takes as input the output feature map F^i_{td} of the previous refinement module in the top-down stream along with the output feature map F^i_{bu} of the skip connection layer attached to the corresponding layer in the bottom-up network. It learns to combine the information from these inputs to produce a new feature map F^{i+1}_{td}, *i.e.*, $F^{i+1}_{td} = R^i(F^i_{td}, F^i_{bu})$. The skip connection layer has a residual bottleneck architecture (He et al., 2016) and downsamples the low-level features to produce a new feature map with 96 channels F^i_{bu}. The refinement module R^i works by first concatenating F^i_{td} and F^i_{bu} and then feeding them to another 3×3 convolutional layer with 128 channels. Finally, an upsampling operation is optionally performed to guarantee that F^i_{td} and F^i_{bu} have the same spatial resolution. Note that the output feature of the last refinement module R^3 is upsampled to the original resolution of the input image before being fed into a convolutional layer with 256 kernels for further refinement and produce a saliency probability map.

3.2.4. Multiscale refinement network training

We train two models based on the same multiscale refinement network architecture to perform two subtasks, *i.e.*, salient region detection and salient object contour detection. We train the models on these two subtasks with separate training sets. As the number of training samples for salient contour detection is much smaller, in practice, we first train a network for salient region detection. Then, a duplicate of this trained network is further fine-tuned for salient contour detection. As the number of "salient object region/contour" and "non-salient-object-region/contour" pixels are imbalanced in each training batch, especially for salient object contour detection, we use a class-balanced cross-entropy function (Xie and Tu, 2015) as the loss function, which can be formulated as follows:

$$L = -\beta \sum_{j \in Y_+} \log P_j - (1-\beta) \sum_{j \in Y_-} \log(1 - P_j), \tag{1}$$

where $\beta = |Y_-|/|Y|$ and $1 - \beta = |Y_+|/|Y|$. Y_+ and Y_- denote the salient and non-salient ground truth label sets, respectively. $P_j = \sigma(a_j) \in [0, 1]$ denotes the probability of that pixel j belongs to salient object region/contour using sigmoid function $\sigma(.)$ on the activation value at pixel j. When training MSRNet for salient region detection, the parameters of the backbone are initialized with an ImageNet (Deng et al., 2009) pre-trained ResNet-50. The parameters of the upsampling operation are initialized with bilinear interpolation weights. The parameters of the rest convolutions in the top-down refinement stream are initialized with random values sampled from a normal distribution with a mean of zero and a standard deviation of 0.01. The model is trained by an Adam (Kingma and Ba, 2014) optimizer with an initializing learning rate of 1e−5.

3.3. Salient instance proposal

In the earlier version of this paper (Li et al., 2017), we proposed to use the multiscale combinatorial grouping (MCG) algorithm (Arbeláez et al., 2014) to generate salient instance proposals from the detected salient object contours. However, MCG generates numerous proposals, which is very time consuming and is also difficult to filter out the noisy ones. Moreover, the lack of consideration on the detected salient region results in the inconsistency of salient instance proposals and detected salient regions. In this section, we propose a simple yet effective morphology algorithm to generate salient instance proposals by using salient object contour to separate occluded salient regions. Here, we define the salient instance proposals as a set of box-level object proposals, which is denoted as P_{bbox}.

Specifically, for an input image, it is fed into the salient region detection network and salient object contour detection network based on MSRNet to generate salient region map and salient object contour map, respectively. And we perform binary classification on its detected salient region map and salient object contour map via simple threshold strategy (threshold is set to 0.5). The binary salient region map and binary salient object contour map are denoted as Rb and Cb, respectively.

Next, we compute the connected components of Rb, which stand for disconnected salient regions and are denoted as $CC_{Rb} = \{cc_1, cc_2, \ldots, cc_n\}$ (n is the number of connected components in Rb). Meanwhile, we filter out the tiny noisy connected components caused by threshold (smaller than 1/20 of the area of the largest connected component in CC_{Rb}).

Then, we use the binary salient contour map Cb to separate occluded salient region Rb. It works by finding the pixels that are active in both salient region map and salient object contour map, and suppressing the corresponding pixels in the binary salient region map Rb to generate a new salient region map Rd, *i.e.*,

$$Rd = Rb - (Rb \cap Cb). \tag{2}$$

After the above preparation, we traverse each connected component of CC_{Rb}. The ith connected component of CC_{Rb} and the corresponding region in Rd of its bounding box are denoted as CC_i and $Rd[BBox(cc_i)]$, respectively. For $Rd[BBox(cc_i)]$, we further compute its connected components, which are denoted as $CC_{Rd[BBox(cc_i)]} = \{cc_{i-1}, cc_{i-2}, \ldots, cc_{i-m}\}$ (m is the number of the connected components in $Rd[BBox(cc_i)]$). To ensure the consistency of detected salient region and salient object contour, we discard the connected components in $CC_{Rd[BBox(cc_i)]}$ that have no salient contour pixels inside. We also discard tiny noisy connected components generated in the process of separating salient regions with salient object contour (smaller than 1/10 of the area of the largest connected component in $Rd[BBox(cc_i)]$).

After the filtering, if the number of left connected components in $CC_{Rd[BBox(cc_i)]}$ is not more than one, the bounding box of cc_i, *i.e.*, $BBox(cc_i)$, will be added to the set of salient instance proposals P_{bbox}. Otherwise, if the number is more than one, the aligned bounding box of each connected component of $CC_{Rd[BBox(cc_i)]}$, *i.e.*, $ABBox(cc_{i-j})$ (j denotes the jth connected component of $CC_{Rd[BBox(cc_i)]}$), will be added to P_{bbox}.

Here, we define $ABBox(cc_{i-j})$ as the aligned bounding box of the connected component cc_{i-j}. Since this process of generating Rd brings details loss for salient regions, the bounding box of cc_{i-j} can only roughly locate the salient instance. Thus, we need to further consider the context of cc_{i-j} from the binary salient region Rb. The proposed aligned bounding box is computed by first extending the width and height of $BBox(cc_{i-j})$ by 5%. Then, if the distance of the edge of the extended bounding box to that of $BBox(cc_i)$ is less than five pixels, the edge of the extended bounding box will be aligned to that of $BBox(cc_i)$.

We call each of the salient instance proposals in P_{bbox} a detected salient instance. We can easily obtain an initial result for salient instance segmentation by labeling the pixels in each salient instance with a unique instance id. A visualization illustration for an example image is present in Fig. 6 to better illustrate the process of generating salient instance proposals.

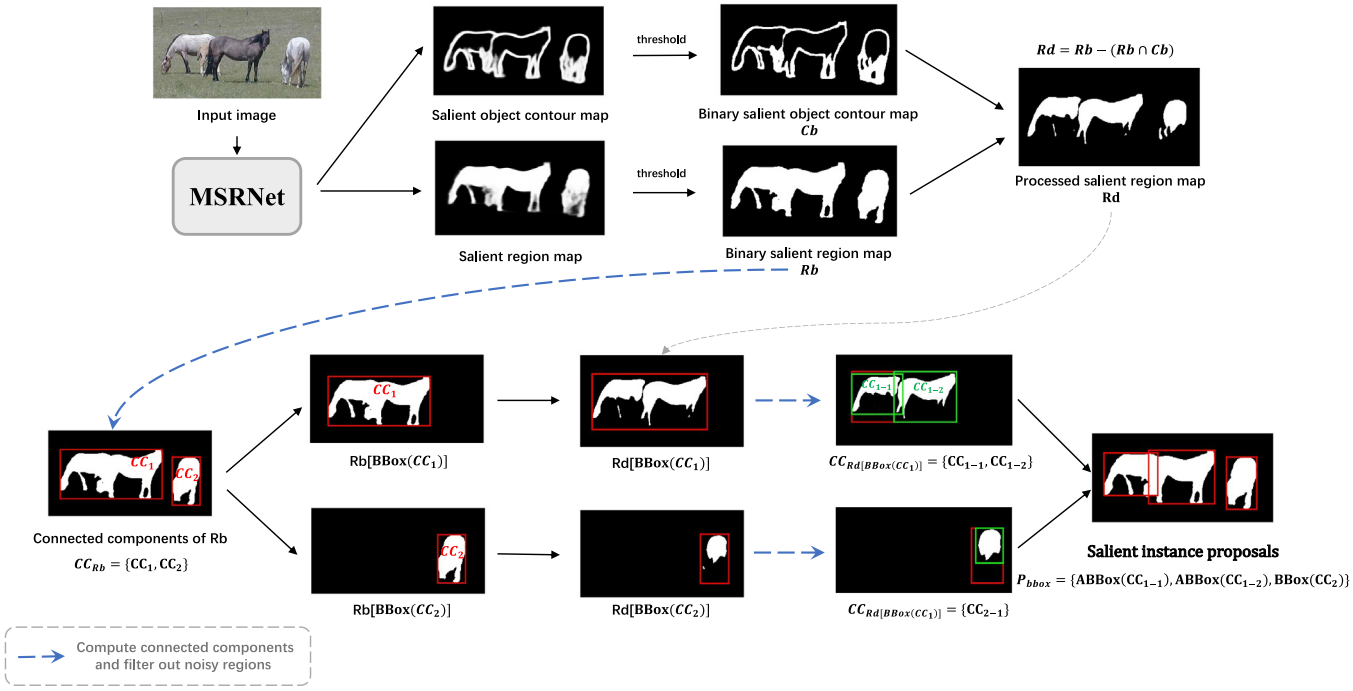

Fig. 6. Visualization illustration of our proposed algorithm for salient instance proposal generation. Please refer to Section 3.3 for more detailed description.

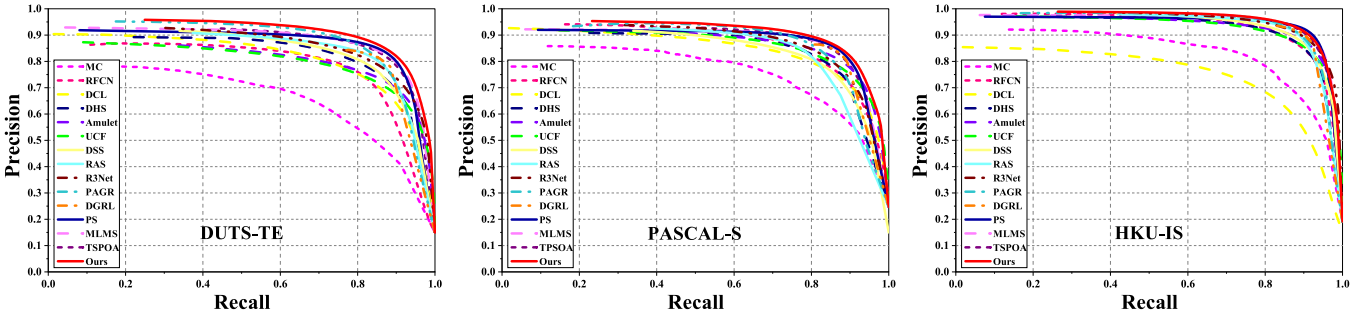

Fig. 7. Comparison of precision–recall curves among 15 representative salient region detection methods (instance-agnostic) on three datasets.

3.4. Refinement for salient instance segmentation

Since there exists losses of details in the processing of generating salient instance proposals, the initial salient instance segmentation results could be incomplete or have overlap. In this section, we introduce a fully connected CRF-based model to refine the initial salient instance segmentation results.

Suppose the number of salient instances is K. We consider the salient instance segmentation as a multi-class labeling problem. In the end, each pixel is assigned with one of the $K+1$ labels using a CRF model. To achieve this goal, we first define a probability map with $K+1$ channels for each input image. Each spatial location of a channel corresponds to the probability that it belongs to the corresponding class of the channel. Here, we treat the background as the first class corresponding to the first channel of the probability map and treat the kth instance as the $(k+1)$th class corresponding to the $(k+1)$th channel of the probability map. Base on the binary salient region map, we define a few rules to generate the probability map as follows:

- For a binary salient region map, the pixels with values equal to 1 are regarded as salient region pixels and those with values equal to 0 are regarded as background pixels.
- If a salient region pixel is covered by a single salient instance (box-level), the probability of this pixel on the channel associated with the salient instance is 1 and is 0 on all other channels.
- If a salient region pixel is not covered by any detected salient instances (box-level), the probability of this pixel on the channel of background is 0 and is $\frac{1}{K}$ on all other channels.
- If a salient region pixel is covered by k overlapping salient instances (box-level), the probability of this pixel on the channels associated with these salient instances is $\frac{1}{k}$ and is 0 on other channels.
- If a background pixel is covered by k overlapping salient instances (box-level), the probability of this pixel on the channels associated with these salient instances as well as on the channel of background is $\frac{1}{k+1}$ and is 0 on other channels.

Given this initial salient instance probability map, we employ a fully connected CRF model (Krähenbühl and Koltun, 2011) for refinement.

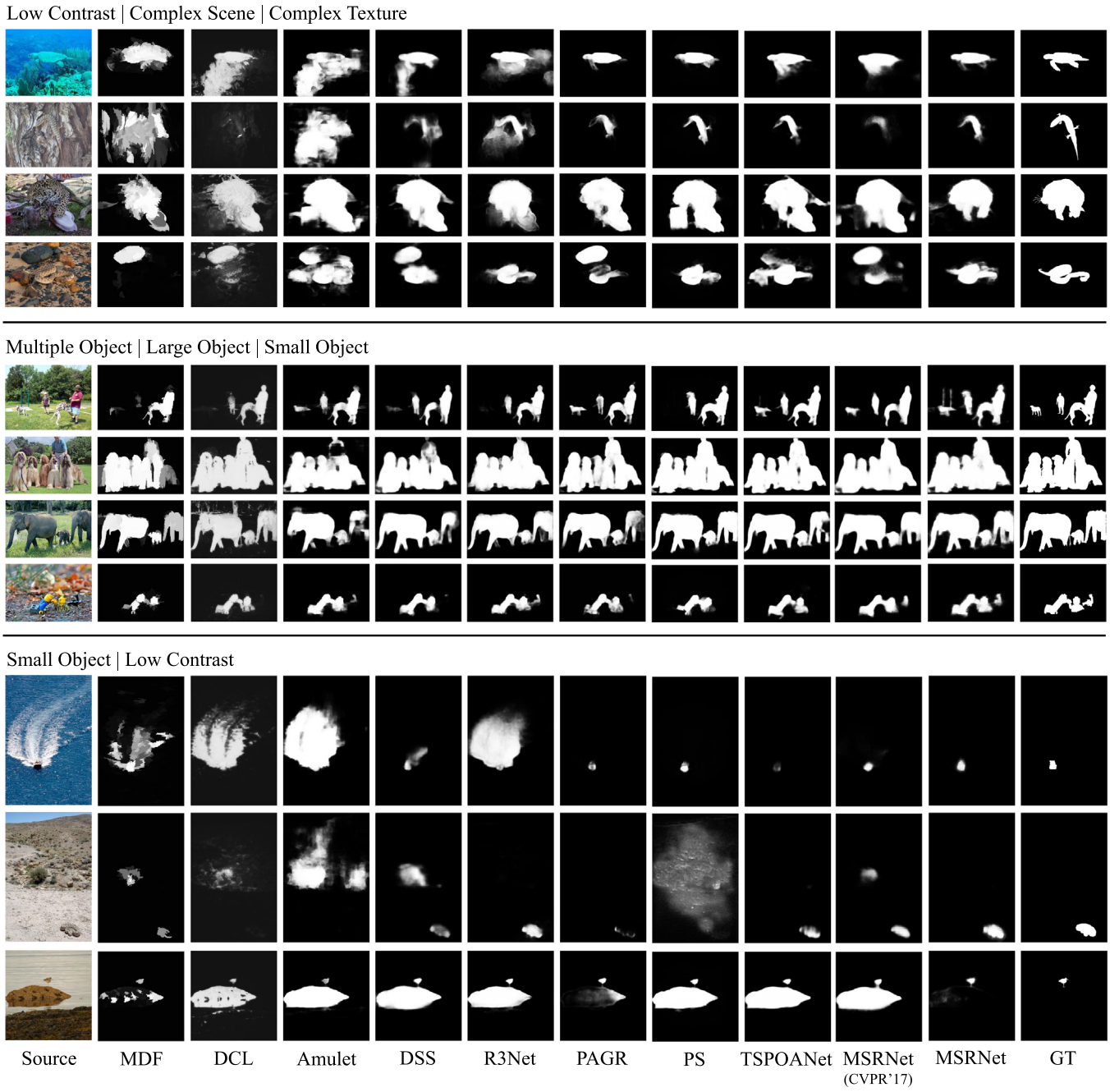

Fig. 8. Visual comparison of the saliency maps generated by state-of-the-art methods, including our conference version and new version of MSRNet. The ground truth (GT) is shown in the last column. We select these images from various challenging cases into multiple groups and highlight the features of images in each group.

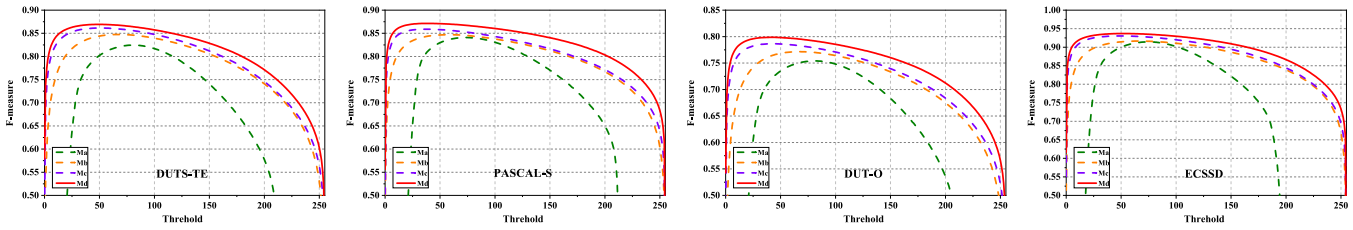

Fig. 9. Ablation study on different components of our proposed multiscale refinement network for salient region detection (instance-agnostic) using F-measure curves.

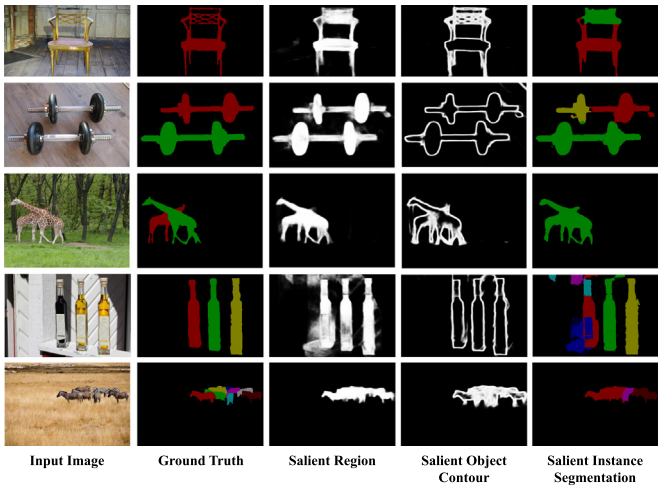

Fig. 10. Failure examples of salient instance segmentation generated by our proposed method.

Table 1
Statistics of our new datasets for salient instance segmentation.

Dataset	#Images	Split			# Object Instances			
		# Train	# Validation	# Test	1	2	3	⩾4
ILSO-1K	1,000	500	200	300	154	504	244	98
Extended	1,000	500	200	300	143	453	231	**173**
ILSO-2K	2,000	1,000	400	600	297	957	475	271

# indicates the number of images.

Specifically, pixel labels are optimized with respect to the following energy function of the CRF:

$$E(x) = -\sum_i \log P(x_i) + \sum_{i,j} \theta_{ij}(x_i, x_j), \tag{3}$$

where x represents a complete label assignment for all pixels and $P(x_i)$ is the probability of pixel i being assigned with the label prescribed by x. $\theta_{ij}(x_i, x_j)$ is a pairwise potential defined as follows:

$$\theta_{ij} = \mu(x_i, x_j)\left[\omega_1 \exp\left(-\frac{\|p_i - p_j\|^2}{2\sigma_\alpha^2} - \frac{\|I_i - I_j\|^2}{2\sigma_\beta^2}\right) + \omega_2 \exp\left(-\frac{\|p_i - p_j\|^2}{2\sigma_\gamma^2}\right)\right], \tag{4}$$

where $\mu(x_i, x_j) = 1$ if $x_i \neq x_j$, and zero otherwise. θ_{ij} involves two kernels. The first bilateral kernel depends on both pixel positions (p) and pixel intensities (I), suggesting adjacent pixels with similar colors to take similar salient instance labels, while the second kernel only depends on the pixel positions when enforcing smoothness. The degree of color similarity and pixel closeness are respectively controlled by two hyper-parameters, σ_α and σ_β. σ_γ controls the scale of the Gaussian kernel. In this paper, we apply the publicly available implementation of Krähenbühl and Koltun (2011) to minimize the above energy. In our experiments, the hyper-parameters are determined through cross-validation using the validation set of our dataset introduced in the next section. The actual values of w_1, w_2, σ_α, σ_β and σ_γ are set to 4.0, 3.0, 49.0, 5.0 and 3.0, respectively.

4. New benchmarks for salient instance segmentation

Since salient instance segmentation was a completely new problem proposed in the preliminary version of this paper (Li et al., 2017), none of the existing datasets could be directly applied to this problem at that time. To promote the study of this problem, we built a new dataset with pixel-wise salient instance annotations in our preliminary version. This new dataset contains 1000 images that are mostly from existing datasets for salient region detection, including ECSSD (Yan et al., 2013), DUT-O (Yang et al., 2013), HKU-IS (Li and Yu, 2015), and MSO (Zhang et al., 2016) datasets. High-quality pixel-wise salient instance labeling and salient object contour are provided for each image. The dataset is divided into three parts, including 500 images for training, 200 images for validation, and 300 images for testing.

In this paper, we further extend the scale of the existing dataset with more challenging samples for salient instance segmentation. In order to distinguish between the dataset in the preliminary version of this paper and the extended dataset in this paper, we name the former ILSO-1K and the latter ILSO-2K. For ILSO-2K, we collected another 1,246 non-copyrighted images from the Internet, most of which contain multiple salient object instances, complex background, or low color contrast. To reduce the label inconsistency, we asked three people to annotate the salient regions with different instance IDs in all selected images using a custom-designed interactive segmentation tool. Only the images with consistent salient instances labeling by all the three annotators are remained. Based on the high-quality salient instance segmentation labels, we can generate the salient instance contours for each image. In the end, another 1000 images with pixel-wise salient object instance labels as well as salient object contour labels are produced to extend the salient instance dataset. The new 1000 images are also randomly divided into three parts, including 500 images for training, 200 images for validation, and 300 images for testing.

In summary, as shown in Table 1, the combination of these new 1000 images and ISLO-1K becomes our new dataset ILSO-2K, which in total has 1000 images for training, 400 images for validation, and 600 images for testing. Moreover, the number of images that have more than 4 salient object instances in the extended 1000 images (173) is larger than that in ILSO-1K (98), which indicates the increment of difficulty in the extended dataset ILSO-2K.

5. Experiments and analyses

5.1. Implementation details

Our proposed MSRNet has been implemented on the MXNet (Chen et al., 2015), a flexible and efficient deep learning platform. For salient region detection, we utilize the training set of DUTS (DUTS-TR) (Wang et al., 2017b) to train our MSRNet. Here, we do not use any validation set and train the model until its training loss converges. To relieve the overfitting when training, we augment the training set by randomly rotating (−10° to 10°), horizontal flipping, and randomly cropping. A workstation with an NVIDIA GTX Titan X GPU and a 2.1 GHz Intel CPU is used for training and testing. It takes about 13 h for our model to converges after 40 epochs on salient region detection. As discussed in Section 3.2.4, this trained model is used as the initial model for salient object contour detection. Since our new dataset ILSO-2K contains only 1000 training images, we perform data augmentation as we used for salient region detection. We fine-tune a duplicated MSRNet on the training set of ILSO-2K for 200 epochs, which takes about 7 h, and keep the lowest validation error on the validation set of ILSO-2K (400 images) as our final model for salient object contour detection.

The batch size is set to 8 on the training phase and set to 32 on the test phase. The input image is resized to 320 × 320. During testing, the final saliency map is resized to the original resolution of the input image. As MSRNet is a fully convolutional network, the inference stage is very efficient.

Table 2
Comparison of quantitative results for salient region detection (instance-agnostic) using maximum F-measure F_β^{max} (larger is better), S-measure S_m (larger is better), and MAE (smaller is better). The top three results on each dataset are shown in **red**, **blue**, and **green**, respectively. "MK" denotes MSRA10K (Cheng et al., 2015), and "MB" denotes MSRA-B (Liu et al., 2011). We also report the inference speed (FPS) and the number of trainable parameters (#Params(M)) of each model.

Methods	FPS	#Params(M)	Training		DUTS-TE			PASCAL-S			DUT-O			HKU-IS			ECSSD			SOD		
			Dataset	#Images	F_β^{max}	S_m	MAE	F_β^{max}	S_m	MAE	F_β^{max}	S_m	MAE	F_β^{max}	S_m	MAE	F_β^{max}	S_m	MAE	F_β^{max}	S_m	MAE
MC (Zhao et al., 2015)	–	–	MK	8,000	.672	.712	.106	.743	.719	.145	.701	.752	.089	.808	.786	.092	.837	.803	.101	.731	.650	.181
MDF (Li and Yu, 2015)	.04	75.68	MB	2,500	.730	.732	.094	.768	.692	.146	.694	.721	.092	.861	.810	.129	.832	.776	.105	.787	.679	.159
RFCN (Wang et al., 2016a)	15	53.00	MK	10,000	.777	.793	.091	.837	.808	.118	.742	.774	.111	.892	.858	.079	.890	.852	.107	.799	.730	.170
ELD (Lee et al., 2016)	–	–	MK	9,000	.738	.753	.093	.773	.757	.123	.715	.750	.092	.839	.820	.074	.867	.839	.079	.764	.705	.155
DCL (Li and Yu, 2016)	13	66.31	MB	2,500	.782	.735	.088	.805	.754	.125	.739	.713	.097	.885	.819	.072	.890	.828	.088	.823	.735	.141
DHS (Liu and Han, 2016)	22	62.22	MK+DUT-O	9,500	.807	.817	.067	.829	.807	.094	n/a	n/a	n/a	.890	.870	.053	.907	.884	.059	.827	.750	.128
Amulet (Zhang et al., 2017a)	19	33.15	MK	10,000	.778	.803	.085	.837	.820	.098	.742	.780	.098	.895	.883	.052	.915	.894	.059	.806	.758	.141
UCF (Zhang et al., 2017b)	23	29.43	MK	10,000	.771	.778	.117	.828	.803	.126	.734	.758	.132	.886	.866	.074	.911	.883	.078	.803	.754	.164
SRM (Wang et al., 2017a)	35	53.18	DUTS	10,533	.827	.834	.059	.847	.832	.085	.769	.797	.069	.906	.887	.046	.917	.895	.054	.843	.742	.127
DSS (Hou et al., 2017)	22	62.23	MB	2,500	.825	.822	.057	.836	.797	.096	.771	.788	.066	.910	.879	.041	.916	.882	.052	.844	.751	.121
RAS (Chen et al., 2018)	43	20.23	MB	2,500	.831	.839	.060	.837	.795	.104	.786	.814	.062	.913	.887	.045	.921	.893	.056	.850	.764	.124
R3Net (Deng et al., 2018)	29	56.16	MK	10,000	.828	.829	.059	.845	.800	.097	.792	.815	.061	.917	.891	.038	.931	.900	.046	.836	.732	.136
PAGR (Zhang et al., 2018)	–	–	DUTS	10,533	.855	.837	.056	.856	.818	.093	.771	.775	.071	.918	.887	.048	.927	.889	.061	.839	.720	.145
DGRL (Wang et al., 2018b)	5	166.08	DUTS	10,533	.829	.841	.050	.854	.836	.072	.774	.806	.062	.910	.895	.036	.922	.903	.041	.845	.771	.104
PS (Wang et al., 2019b)	–	–	MK	10,000	.855	.864	.049	.864	.850	.071	.813	.837	.061	.913	.907	.038	.930	.918	.041	.824	.800	.103
MLMSNet (Wu et al., 2019a)	13	74.38	DUTS+BSDS	10,533+300	.851	.861	.049	.862	.845	.074	.774	.809	.064	.921	.906	.039	.928	.911	.045	.854	.790	.107
HRSOD-DH (Zeng et al., 2019)	20	32.39	DUTS+HRSOD	12,163	.836	.822	.051	.854	.812	.083	.743	.763	.065	.910	.877	.042	.925	.888	.052	.821	.709	.137
TSPOANet (Liu et al., 2019b)	–	–	DUTS	10,533	.850	.859	.049	.861	.841	.078	.784	.818	.061	.919	.902	.038	.919	.907	.047	.854	.775	.115
MSRNet (Li et al., 2017)	9	82.91	MB+HKU-IS	5,000	.829	.840	.061	.855	.840	.081	.782	.808	.073	n/a	n/a	n/a	.911	.896	.054	.836	.779	.113
MSRNet	37	59.56	DUTS	10,533	.869	.868	.052	.871	.851	.075	.799	.819	.070	.922	.901	.046	.937	.912	.048	.862	.801	.101

"n/a": Training on subset. Corresponding test results are excluded here.

Table 3
Ablation study on different components of our proposed multiscale refinement network for salient region detection (instance-agnostic) using maximum F-measure F_β^{max} (larger is better), S-measure S_m (larger is better), weighted F-measure F_β^w (larger is better), and MAE (smaller is better). The best scores marked in **bold**.

Methods	Refinement	ASPP		DUTS-TE (Wang et al., 2017b)				PASCAL-S (Li et al., 2014)				DUT-O (Yang et al., 2013)				ECSSD (Yan et al., 2013)			
		w/o attention	w/ attention	F_β^{max}	S_m	F_β^w	MAE	F_β^{max}	S_m	F_β^w	MAE	F_β^{max}	S_m	F_β^w	MAE	F_β^{max}	S_m	F_β^w	MAE
M_a				0.825	0.698	0.492	0.111	0.841	0.738	0.582	0.141	0.754	0.671	0.456	0.129	0.911	0.791	0.650	0.116
M_b	✓			0.848	0.847	0.692	0.063	0.847	0.828	0.722	0.092	0.772	0.798	0.611	0.082	0.917	0.892	0.814	0.064
M_c	✓	✓		0.861	0.856	0.723	0.060	0.859	0.837	0.747	0.086	0.787	0.807	0.648	0.079	0.930	0.903	0.838	0.055
M_d	✓		✓	**0.869**	**0.868**	**0.751**	**0.052**	**0.871**	**0.851**	**0.779**	**0.075**	**0.799**	**0.819**	**0.675**	**0.070**	**0.937**	**0.912**	**0.861**	**0.048**

5.2. Evaluation on salient region detection

5.2.1. Datasets

To evaluate the performance of our MSRNet on salient region detection, we conduct testing on six benchmark datasets, including DUTS (Wang et al., 2017b), PASCAL-S (Li et al., 2014), DUT-O (Yang et al., 2013), HKU-IS (Li and Yu, 2015), ECSSD (Yan et al., 2013), and SOD (Movahedi and Elder, 2010). These datasets contain a large number of images as well as pixel-wise salient region annotations and have been widely used for salient region detection.

DUTS (Wang et al., 2017b). This dataset contains 10,533 images in the training set, *i.e.*, DUTS-TR, and 5,019 images in the test set, *i.e.*, DUTS-TE. It is currently the largest image salient object detection benchmark, which contains various challenging scenarios for saliency detection. The pixel-wise ground truth saliency masks were annotated by 50 subjects.

PASCAL-S (Li et al., 2014). This dataset contains 850 natural images, each of which contains multiple objects and is from the validation set of PASCAL VOC 2010 segmentation dataset. The ground truth saliency masks were labeled by 12 subjects. The final saliency value of each object is the proportion of being labeled as saliency by the subjects. Here, we binarize the masks at a threshold of 0.5 to obtain binary saliency masks as suggested by Li et al. (2014).

DUT-O (Yang et al., 2013). This dataset contains 5,168 natural images, where both bounding boxes and pixel-wise salient object annotations are provided. Through a careful observation of this dataset, we have noticed that many saliency annotations in this dataset are ambiguous to different human observers. As a consequence, none of the existing salient object detection methods have achieved high accuracy on this dataset.

HKU-IS (Li and Yu, 2015). This dataset contains 4,447 images with high-quality pixel-wise annotation of salient objects. Only the images with low color contrast, complex background, or multiple salient objects were chosen to construct this dataset.

ECSSD (Yan et al., 2013). This dataset contains 1000 semantically meaningful and structurally complex images with pixel-wise annotations.

SOD (Movahedi and Elder, 2010). This dataset contains 300 images with salient object boundaries based on the Berkeley Segmentation Dataset. Pixel-wise annotations of the salient objects in this dataset were generated by Jiang et al. (2013). Many images in this dataset contain multiple salient objects with low contrast that makes this dataset more challenging.

The performance of MSRA-B (Cheng et al., 2015) dataset, which is reported in the earlier version of this paper (Li et al., 2017), is no longer discussed here as it is relatively simple and has achieved very high performance. Noted that as we train our network on the training set of DUTS (DUTS-TR), we evaluate our trained model on the test set of DUTS (DUTS-TE) as well as the entire dataset of all other benchmarks.

5.2.2. Evaluation criteria

We adopt precision–recall curves (PR), F-measure (Achanta et al., 2009), S-measure (Fan et al., 2017), mean absolute error (MAE) (Perazzi et al., 2012), and weighted F-measure (Margolin et al., 2014) to evaluate the performance of MSRNet as well as other state-of-the-art salient region detection methods.

PR curves can be obtained by taking the average of precision and recall values of all images in each dataset and connecting the pairs of average precision and average recall values at different thresholds ([0,255]). Specifically, a pair of Precision and Recall value can be obtained by comparing the ground truth saliency map with the saliency map binarized at a certain threshold, which is defined as:

$$Precision = \frac{TP}{TP+FP}, Recall = \frac{TP}{TP+FN}, \tag{5}$$

where TP, TN, FP, FN denote true-positive, true-negative, false-positive, and false-negative, respectively.

F-measure (Achanta et al., 2009) indicates an overall performance at different thresholds ([0,255]), which can be formulated as a weighted combination of precision and recall:

$$F_\beta = \frac{(1+\beta^2)\cdot Precision \cdot Recall}{\beta^2 \cdot Precision + Recall}, \tag{6}$$

where β^2 is set to 0.3 to weight precision more than recall as suggested by Achanta et al. (2009). We report the whole F-measure curve and also the maximum F-measure, which provides a summary of salient object detection performance.

S-measure (Fan et al., 2017) can evaluate region-aware (S_r) as well as object-aware (S_o) structural similarity between a saliency map and its corresponding ground truth simultaneously:

$$S_m = \alpha \times S_o + (1-\alpha) \times S_r, \tag{7}$$

where α is empirically set to 0.5.

MAE (Perazzi et al., 2012) measures the average pixelwise absolute difference between the binary ground truth $GT \in \{0,1\}^{W\times H}$ and predicted saliency map $S \in [0,1]^{W\times H}$:

$$\text{MAE} = \frac{1}{W\times H}\sum_{x=1}^{W}\sum_{y=1}^{H}|S(x,y) - GT(x,y)|. \tag{8}$$

Weighted F-measure (Margolin et al., 2014) alters the way of computing F-measure by replacing the Precision and Recall with weighted Precision ($Precision^w$) and weighted Recall ($Recall^w$):

$$F_\beta^w = \frac{(1+\beta^2)\cdot Precision^w \cdot Recall^w}{\beta^2 \cdot Precision^w + Recall^w}. \tag{9}$$

5.2.3. Comparison with state-of-the-art

For salient region detection, we compare the proposed MSRNet with other 19 state-of-the-art salient region detection methods, including MC (Zhao et al., 2015), MDF (Li and Yu, 2015), RFCN (Wang et al., 2016a), ELD (Lee et al., 2016), DCL (Li and Yu, 2016), DHS (Liu and Han, 2016), Amulet (Zhang et al., 2017a), UCF (Zhang et al., 2017b), SRM (Wang et al., 2017a), DSS (Hou et al., 2017), RAS (Chen et al., 2018), R3Net (Deng et al., 2018), PAGR (Zhang et al., 2018), DGRL (Wang et al., 2018b), PS (Wang et al., 2019b), MLMSNet (Wu et al., 2019a), HRSOD-DH (Zeng et al., 2019), TSPOANet (Liu et al., 2019b), and MSRNet in our conference version (Li et al., 2017). We use the saliency maps provided by the authors or computed using their released implementations for comparison.

As a part of quantitative evaluation, a comparison of PR curves on three benchmark datasets is presented in Fig. 7. Moreover, a quantitative comparison using maximum F-measure, S-measure, and MAE on six benchmark datasets is given in Table 2. As can be seen, our proposed MSRNet achieves stably and satisfactory performance when compared to state-of-the-art salient region detection (instance-agnostic) methods without resorting to any post-processing techniques or edge labels. More specially, our proposed network performs best w.r.t S-measure, which indicates that the proposed network can generate salient region maps with high region-aware and object-aware structural similarity compared to the ground truth. Besides, as shown the in last two columns in Table 2, our new MSRNet consistently outperforms our conference version by a large margin across all the six benchmark datasets.

A visual comparison is given in Fig. 8. Due to space limitation, we select 9 representative methods to compare with our new version of MSRNet. As can be seen, although some of the state-of-the-art methods perform well in some challenging cases, they still fail to handle other complex cases. By contrast, our redesigned MSRNet can promote the performance of its conference version and accurately detect the salient objects in various challenging scenarios, *e.g.*, low color contrast between salient objects and background (the first two rows), salient objects with complex texture (3rd and 4th rows), multiple salient objects (5th to 8th rows), objects touching the image boundary (7th row), and small salient object (the last four rows).

Table 4
Comparison of quantitative results for salient instance segmentation (instance-level) on the test sets of ILSO-1K and ILSO-2K. Please note that the models reported on each dataset (ILSO-1K, ILSO-2K) are trained on its corresponding training (500 images for ILSO-1K, 1000 images for ILSO-2K) and validation (200 images for ILSO-1K, 400 images for ILSO-2K) sets.

Method	Pub.	ILSO-1K (Li et al., 2017)					ILSO-2K				
		mAP^r@0.5	mAP^r@0.6	mAP^r@0.7	mAP^r@0.8	mAP^r@0.9	mAP^r@0.5	mAP^r@0.6	mAP^r@0.7	mAP^r@0.8	mAP^r@0.9
Ours (Li et al., 2017)	CVPR'17	65.32%	–	52.18%	–	–	–	–	–	–	–
S4Net (Fan et al., 2019b)	CVPR'19	82.84%	78.88%	71.62%	57.26%	23.27%	73.11%	64.22%	52.98%	34.09%	11.90%
Ours		**85.15%**	**81.68%**	**74.75%**	**64.19%**	**37.29%**	**78.32%**	**73.70%**	**66.57%**	**55.55%**	**29.10%**

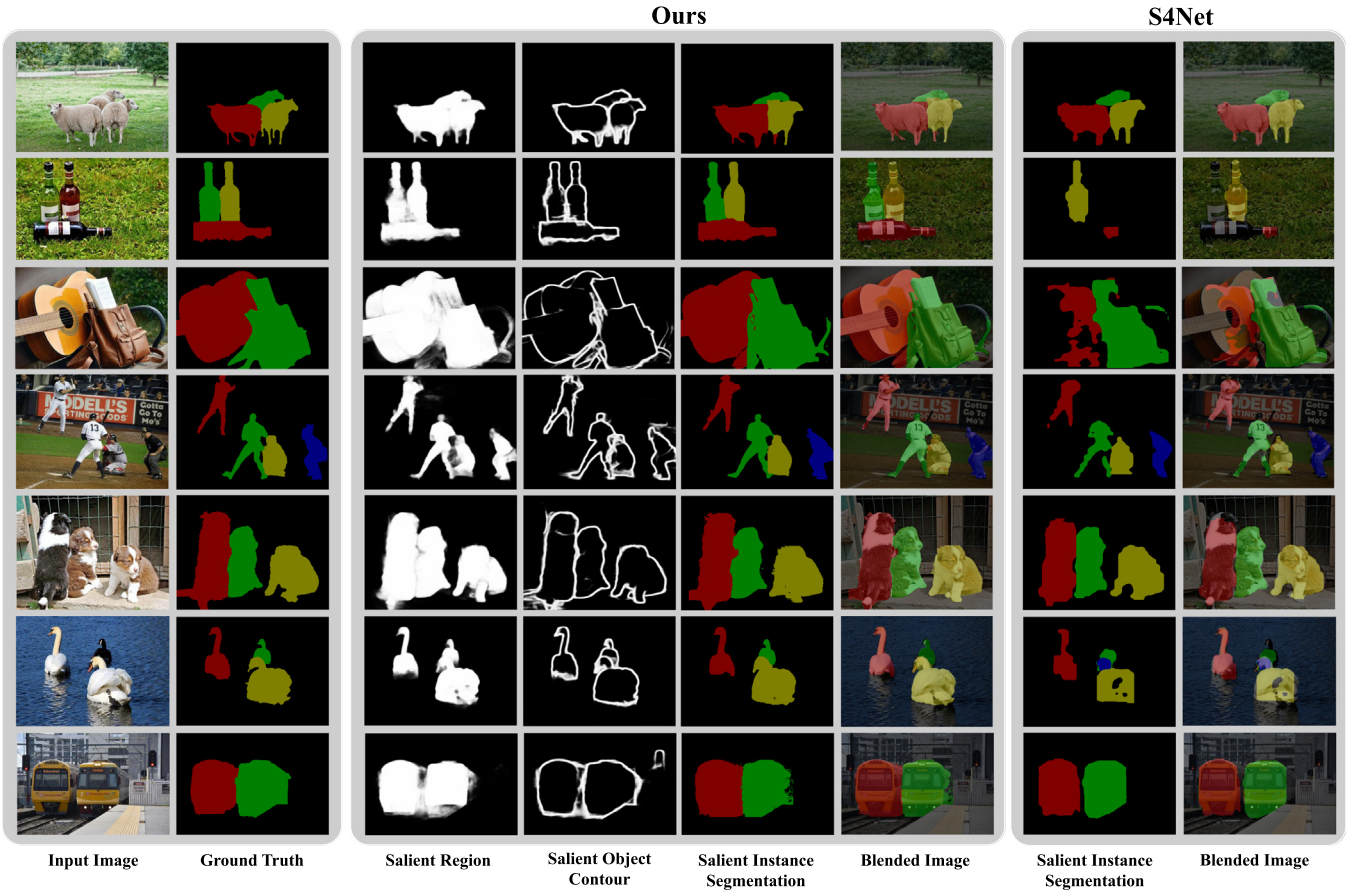

Fig. 11. Examples of salient instance segmentation results generated by our redesigned MSRNet-based framework on the ILSO-2K dataset. For each image, we show the ground truth of salient instance segmentation, the detected salient region map, the detected salient object contour, and the predicted salient instance segmentation result. Moreover, we also provide the salient instance segmentation results of S4Net (Fan et al., 2019b) for qualitative comparison. Note that for salient instance segmentation, different colors indicate different salient object instances. Best viewed in color. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5.2.4. Effectiveness of multiscale refinement network

As discussed in Section 3.2, our proposed MSRN consists of a refined ResNet-50 stream and a learned attentional ASPP for fusing features at different scales. To validate the effectiveness and necessity of each component, we compare MSRNet with its three variants in Table 3 and Fig. 9. Specifically, M_a refers to a modified ResNet-50 backbone network without any refinement architecture. M_b refers to a ResNet-50-based refinement network. M_c refers to a ResNet-50-based refinement network with a vanilla ASPP module. M_d refers to our proposed MSRNet, a ResNet-50-based refinement network with an attentional ASPP module. These three variants are trained using the same strategy as training MSRNet. Quantitative results from the four methods are obtained from the test set of DUTS and the whole dataset of PASCAL-S, DUT-O, and ECSSD. As shown in Table 3 and Fig. 9, MSRNet consistently achieves the best performing in terms of maximum F-measure, S-measure, and F-measure curve. By comparing M_b with M_a, we can find that the refinement architecture brings a significant improvement on all metrics, especially on weighted F-measure, which demonstrates the effectiveness of the refinement module in MSRNet. By comparing M_c and M_d with M_b, we can find that on the basis of refinement architecture, the ASPP module can bring extra performance boost w.r.t maximum F-measure, S-measure, weighted F-measure, and MAE as it can help the model to detect salient objects at multiple scales in a same feature level. The better performance of M_d further demonstrates the effectiveness of our proposed attention model for fusing features at different scales in ASPP. Moreover, the refinement architecture and the attentional ASPP are complementary to each other, which makes MSRNet capable of detecting salient regions more precisely as well as discovering salient objects at multiple scales.

5.3. Evaluation on salient instance segmentation

We adopt two types of performance measures to evaluate the performance of our proposed framework for instance-level salient object segmentation.

First, we use the same performance measures as traditional edge detection (Arbelaez et al., 2011; Xie and Tu, 2015) to evaluate the performance of salient object contour detection, *i.e.*, three standard measures: fixed contour threshold (ODS), per-image best threshold (OIS), and average precision (AP). Second, we define performance measures for salient instance segmentation by drawing inspirations from the evaluation of instance-aware semantic segmentation (Hariharan et al., 2014). Specifically, we adopt mean average precision using region IoU (intersection-over-union) at different thresholds (0.5, 0.6, 0.7, 0.8, and 0.9), which are denoted as $mAP^r@X, X \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$.

5.3.1. Quantitative evaluation

Quantitative benchmark results of salient object contour detection and salient instance segmentation are given in Tables 4 and 5, respectively. Here, we report the results on both ILSO-1K and ILSO-2K datasets. Please note that the reported results on ILSO-1K and ILSO-2K are produced by the models trained on the training and validation sets of ILSO-1K and ILSO-2K, respectively. As can be seen, our proposed salient instance segmentation framework improves the mAP^r@0.5 achieved by our preliminary version by 30.36% and improves mAP^r@0.7 by 43.25% on the test set of ISLO-1K.

Moreover, we also provide a detailed comparison with the state-of-the-art salient instance segmentation algorithm S4Net (Fan et al., 2019b) on both ILSO-1K and ILSO-2K datasets. Here, we use the published implementation of S4Net trained on ILSO-1K to generate the test results of ILSO-1K. And we also re-train the S4Net on ILSO-2K to generate the test results of ILSO-2K using the default settings provided by the authors. As shown in Table 4, our redesigned pipeline for salient instance segmentation consistently outperforms S4Net w.r.t mAP^r with IoU scores of 0.5 to 0.9 on both ILSO-1K and ILSO-2K datasets by a large margin.

5.3.2. Qualitative evaluation

Fig. 11 presents a visualization of the salient region maps, salient contour maps, and salient instance segmentation maps generated by our redesigned pipeline. Moreover, the salient instance segmentation results of S4Net are also provided for better comparison. As shown in the figure, S4Net might fail to segment some spatially connected objects (2nd row) or produce inaccurate candidates (6th row). Moreover, the mask-level results of S4Net are usually rough especially around object boundaries (3rd and 5th rows). By contrast, our proposed MSRNet can not only detect the salient region accurately but also distinguish the contour of salient instances correctly. Based on the detected salient region and contour, our framework can handle challenging scenarios where multiple salient object instances are spatially connected.

5.4. Run-time analysis

In Table 6, we provide a detailed run-time analysis of our proposed pipeline for salient instance segmentation, including the salient region detection (SRD), salient object contour detection (SOCD), and salient instance segmentation (SIS). In our experiments, all timings are measured on a workstation with an NVIDIA GTX Titan X GPU and a 2.1 GHz Intel CPU. The pipeline of our preliminary version (Li et al., 2017) takes about 113 ms to generate a salient region/contour map via the preliminary version of MSRNet and another 437 ms for post-processing via CRF (Krähenbühl and Koltun, 2011). Given the generated salient region and salient object contour maps, it takes about 4900 ms to generate salient instance proposals via MCG (Arbeláez et al., 2014) and another 1600 ms for the CRF-based refinement of salient instance segmentation. In total, the pipeline of our preliminary version takes about 7.6 s to perform salient instance segmentation for an input image.

Table 5
Quantitative benchmark results of salient object contour detection on test sets of our new dataset.

Method	ILSO-1K			ILSO-2K		
	ODS	OIS	AP	ODS	OIS	AP
Ours (Li et al., 2017)	0.719	0.757	0.765	–	–	–
Ours	0.839	0.869	0.767	0.838	0.877	0.783

Table 6
Run-time analysis for the pipeline of salient instance segmentation, including our preliminary version (Li et al., 2017) and redesigned version. As mentioned in Section 3, the pipeline includes the MSRNet for salient region detection (SRD) and salient object contour detection (SOCD), as well as salient proposal generation (Proposal) and salient instance refinement (Refinement) for salient instance segmentation (SIS).

Method	SRD/SOCD	SIS		Total Times
	MSRNet	CRF	Proposal	Refinement	
Ours (Li et al., 2017)	113 ms	437 ms	4,900 ms	1,600 ms	7,600 ms
Ours	27 ms	–	10 ms	1,600 ms	1,664 ms

While, as for the pipeline of our new version, it takes only about 27 ms to perform either salient region detection or salient object contour detection for an input image via the redesigned MSRNet without resorting to any post-processing techniques like CRF, which reaches a real-time speed of 37 FPS. It takes about 10 ms to generate salient instance proposals based on the generated salient region and contour maps. The CRF-based refinement will take another 1600 ms with CRF being the bottleneck for the refinement of the resulted salient instance segmentation. In total, the redesigned pipeline takes about 1.7 s to perform salient instance segmentation for an input image.

5.5. Failure analysis

Although our proposed pipeline can handle many challenging situations, it might still fail in some complex cases. Here, we visualize some failure examples of salient instance segmentation generated by our proposed pipeline in Fig. 10. Since our proposed salient instance segmentation algorithm is based on the salient region and contour maps generated by MSRNet, the salient instance segmentation is quite sensitive to the quality of both salient region and contour maps. For example, in Fig. 10, MSRNet fails to generate accurate salient region maps for elongated objects (2nd row) and transparent objects (4th row), which results in the failure of salient instance segmentation. Moreover, it is also very difficult to precisely separate the spatially connected salient objects, when there are complex object contours (1st rows), severe object overlapping (3rd row), or multiple small salient objects close to each other (5th row).

6. Conclusion

In this paper, we focus on a new problem proposed in the preliminary version of this paper, *i.e.*, salient instance segmentation. To solve this problem, we present a framework to combine the salient object region and contour to generate instance-level salient object segmentation. The essential component of our framework is the multiscale refinement network, an end-to-end trained fully convolutional network that is used to generate high-quality salient region masks and salient object contours with high efficiency. To promote further research and evaluation of salient instance segmentation, we have also extended the scale of existing salient instance datasets bringing a new dataset of 2,000 challenging images with pixel-wise salient instance annotations. Experimental results demonstrate that our proposed method can outperform its preliminary version by a large margin and achieve satisfactory performance on six public benchmarks for salient object region detection as well as on our new dataset for salient instance segmentation.

CRediT authorship contribution statement

Guanbin Li: Conceptualization, Data curation, Supervision, Writing - review & editing. **Pengxiang Yan:** Methodology, Investigation, Software, Writing - original draft. **Yuan Xie:** Methodology, Visualization, Validation. **Guisheng Wang:** Writing - review & editing. **Liang Lin:** Supervision, Resources. **Yizhou Yu:** Supervision.

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.

Acknowledgments

This work was supported in part by the National Key Research and Development Program of China (Grant number: 2019YFC0118100), in part by the Guangdong Basic and Applied Basic Research Foundation, China (Grant number: 2020B1515020048), in part by the Natural Science Foundations of Guangdong, China (Grant number: 2017A03031335) and in part by the National Natural Science Foundation of China (Grant number: 61976250, U1811463). This work was also sponsored by the CCF-Tencent Open Research Fund, China.

References

Achanta, R., Hemami, S., Estrada, F., Süsstrunk, S., 2009. Frequency-tuned salient region detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1597–1604.

Arbelaez, P., Maire, M., Fowlkes, C., Malik, J., 2011. Contour detection and hierarchical image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 33 (5), 898–916.

Arbeláez, P., Pont-Tuset, J., Barron, J.T., Marques, F., Malik, J., 2014. Multiscale combinatorial grouping. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 328–335.

Chen, X., Gupta, A., 2015. Webly supervised learning of convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1431–1439.

Chen, T., Li, M., Li, Y., Lin, M., Wang, N., Wang, M., Xiao, T., Xu, B., Zhang, C., Zhang, Z., 2015. Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems. arXiv preprint arXiv:1512.01274.

Chen, L.-C., Papandreou, G., Schroff, F., Adam, H., 2017. Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587.

Chen, H., Sun, K., Tian, Z., Shen, C., Huang, Y., Yan, Y., 2020. BlendMask: Top-down meets bottom-up for instance segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8573–8581.

Chen, S., Tan, X., Wang, B., Hu, X., 2018. Reverse attention for salient object detection. In: Proceedings of European Conference on Computer Vision. pp. 234–250.

Cheng, M.-M., Mitra, N.J., Huang, X., Torr, P.H., Hu, S.-M., 2015. Global contrast based salient region detection. IEEE Trans. Pattern Anal. Mach. Intell. 37 (3), 569–582.

Dai, J., He, K., Li, Y., Ren, S., Sun, J., 2016a. Instance-sensitive fully convolutional networks. In: Proceedings of European Conference on Computer Vision. pp. 534–549.

Dai, J., He, K., Sun, J., 2016b. Instance-aware semantic segmentation via multi-task network cascades. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3150–3158.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255.

Deng, Z., Hu, X., Zhu, L., Xu, X., Qin, J., Han, G., Heng, P.-A., 2018. R3Net: Recurrent residual refinement network for saliency detection. In: Proceedings of the International Joint Conference on Artificial Intelligence. pp. 684–690.

Fan, R., Cheng, M.-M., Hou, Q., Mu, T.-J., Wang, J., Hu, S.-M., 2019b. S4net: Single stage salient-instance segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6103–6112.

Fan, D.-P., Cheng, M.-M., Liu, Y., Li, T., Borji, A., 2017. Structure-measure: A new way to evaluate foreground maps. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 548–4557.

Fan, D.-P., Wang, W., Cheng, M.-M., Shen, J., 2019a. Shifting more attention to video salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8554–8564.

Feng, M., Lu, H., Ding, E., 2019. Attentive feedback network for boundary-aware salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1623–1632.

Gao, Y., Shi, M., Tao, D., Xu, C., 2015. Database saliency for fast image retrieval. IEEE Trans. Multimed. 17 (3), 359–369.

Goferman, S., Zelnik-Manor, L., Tal, A., 2012. Context-aware saliency detection. IEEE Trans. Pattern Anal. Mach. Intell. 34 (10), 1915–1926.

Hariharan, B., Arbeláez, P., Girshick, R., Malik, J., 2014. Simultaneous detection and segmentation. In: Proceedings of European Conference on Computer Vision. pp. 297–312.

He, K., Gkioxari, G., Dollár, P., Girshick, R., 2017. Mask r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2961–2969.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 770–778.

Hou, Q., Cheng, M.-M., Hu, X., Borji, A., Tu, Z., Torr, P., 2017. Deeply supervised salient object detection with short connections. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5300–5309.

Huo, S., Zhou, Y., Lei, J., Ling, N., Hou, C., 2017. Iterative feedback control-based salient object segmentation. IEEE Trans. Multimed. 20 (6), 1350–1364.

Jiang, H., Wang, J., Yuan, Z., Wu, Y., Zheng, N., Li, S., 2013. Salient object detection: A discriminative regional feature integration approach. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2083–2090.

Karpathy, A., Fei-Fei, L., 2015. Deep visual-semantic alignments for generating image descriptions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3128–3137.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. In: Proceedings of the International Conference on Learning Representations.

Klein, D.A., Frintrop, S., 2011. Center-surround divergence of feature statistics for salient object detection. In: 2011 International Conference on Computer Vision. IEEE, pp. 2214–2219.

Krähenbühl, P., Koltun, V., 2011. Efficient inference in fully connected crfs with gaussian edge potentials. In: Advances in Neural Information Processing Systems. pp. 109–117.

Lai, B., Gong, X., 2016. Saliency guided dictionary learning for weakly-supervised image parsing. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3630–3639.

Lee, G., Tai, Y.-W., Kim, J., 2016. Deep saliency with encoded low level distance map and high level features. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 660–668.

Li, Y., Hou, X., Koch, C., Rehg, J.M., Yuille, A.L., 2014. The secrets of salient object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 280–287.

Li, G., Xie, Y., Lin, L., Yu, Y., 2017. Instance-level salient object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2386–2395.

Li, G., Yu, Y., 2015. Visual saliency based on multiscale deep features. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5455–5463.

Li, G., Yu, Y., 2016. Deep contrast learning for salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 478–487.

Liu, Y., Cheng, M.-M., Hu, X., Wang, K., Bai, X., 2017. Richer convolutional features for edge detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3000–3009.

Liu, N., Han, J., 2016. Dhsnet: Deep hierarchical saliency network for salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 678–686.

Liu, J.-J., Hou, Q., Cheng, M.-M., Feng, J., Jiang, J., 2019a. A simple pooling-based design for real-time salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3917–3926.

Liu, T., Yuan, Z., Sun, J., Wang, J., Zheng, N., Tang, X., Shum, H.-Y., 2011. Learning to detect a salient object. IEEE Trans. Pattern Anal. Mach. Intell. 33 (2), 353–367.

Liu, Y., Zhang, Q., Zhang, D., Han, J., 2019b. Employing deep part-object relationships for salient object detection. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1232–1241.

Ma, Y.-F., Hua, X.-S., Lu, L., Zhang, H.-J., 2005. A generic framework of user attention model and its application in video summarization. IEEE Trans. Multimed. 7 (5), 907–919.

Margolin, R., Zelnik-Manor, L., Tal, A., 2014. How to evaluate foreground maps? In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255.

Movahedi, V., Elder, J.H., 2010. Design and perceptual validation of performance measures for salient object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 49–56.

Navalpakkam, V., Itti, L., 2006. An integrated model of top-down and bottom-up attention for optimizing detection speed. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2049–2056.

Pei, J., Tang, H., Liu, C., Chen, C., 2020. Salient instance segmentation via subitizing and clustering. Neurocomputing.

Perazzi, F., Krähenbühl, P., Pritch, Y., Hornung, A., 2012. Saliency filters: Contrast based filtering for salient region detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 733–740.

Pinheiro, P.O., Lin, T.-Y., Collobert, R., Dollár, P., 2016. Learning to refine object segments. In: Proceedings of European Conference on Computer Vision. pp. 75–91.

Qin, X., Zhang, Z., Huang, C., Gao, C., Dehghan, M., Jagersand, M., 2019. Basnet: Boundary-aware salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7479–7489.

Romera-Paredes, B., Torr, P.H.S., 2016. Recurrent instance segmentation. In: Proceedings of European Conference on Computer Vision. pp. 312–329.

Rutishauser, U., Walther, D., Koch, C., Perona, P., 2004. Is bottom-up attention useful for object recognition? In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Sugano, Y., Matsushita, Y., Sato, Y., 2010. Calibration-free gaze sensing using saliency maps. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 2667–2674.

Wang, T., Borji, A., Zhang, L., Zhang, P., Lu, H., 2017a. A stagewise refinement model for detecting salient objects in images. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4019–4028.

Wang, X., Kong, T., Shen, C., Jiang, Y., Li, L., 2020. Solo: Segmenting objects by locations. In: European Conference on Computer Vision. Springer, pp. 649–665.

Wang, W., Lai, Q., Fu, H., Shen, J., Ling, H., Yang, R., 2019a. Salient object detection in the deep learning era: An in-depth survey. arXiv preprint arXiv:1904.09146.

Wang, L., Lu, H., Wang, Y., Feng, M., Wang, D., Yin, B., Ruan, X., 2017b. Learning to detect salient objects with image-level supervision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 136–145.

Wang, W., Shen, J., Cheng, M.-M., Shao, L., 2019b. An iterative and cooperative top-down and bottom-up inference network for salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5968–5977.

Wang, W., Shen, J., Dong, X., Borji, A., Yang, R., 2019c. Inferring salient objects from human fixations. IEEE Trans. Pattern Anal. Mach. Intell. 42 (8), 1913–1927.

Wang, W., Shen, J., Ling, H., 2018a. A deep network solution for attention and aesthetics aware photo cropping. IEEE Trans. Pattern Anal. Mach. Intell. 41 (7), 1531–1544.

Wang, W., Shen, J., Yang, R., Porikli, F., 2017c. Saliency-aware video object segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 40 (1), 20–33.

Wang, L., Wang, L., Lu, H., Zhang, P., Ruan, X., 2016a. Saliency detection with recurrent fully convolutional networks. In: Proceedings of European Conference on Computer Vision. pp. 825–841.

Wang, Z., Xiang, D., Hou, S., Wu, F., 2016b. Background-driven salient object detection. IEEE Trans. Multimed. 19 (4), 750–762.

Wang, T., Zhang, L., Wang, S., Lu, H., Yang, G., Ruan, X., Borji, A., 2018b. Detect globally, refine locally: A novel approach to saliency detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3127–3135.

Wang, W., Zhao, S., Shen, J., Hoi, S.C., Borji, A., 2019d. Salient object detection with pyramid attention and salient edges. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1448–1457.

Wei, Y., Liang, X., Chen, Y., Shen, X., Cheng, M.-M., Feng, J., Zhao, Y., Yan, S., 2017. Stc: A simple to complex framework for weakly-supervised semantic segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 39 (11), 2314–2320.

Wei, J., Wang, S., Wu, Z., Su, C., Huang, Q., Tian, Q., 2020. Label decoupling framework for salient object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13025–13034.

Wei, Y., Xia, W., Huang, J., Ni, B., Dong, J., Zhao, Y., Yan, S., 2016. HCP: A flexible CNN framework for multi-label image classification. IEEE Trans. Pattern Anal. Mach. Intell. 38 (9), 1901–1907.

Wu, R., Feng, M., Guan, W., Wang, D., Lu, H., Ding, E., 2019a. A mutual learning method for salient object detection with intertwined multi-supervision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8150–8159.

Wu, H., Li, G., Luo, X., 2014. Weighted attentional blocks for probabilistic object tracking. Vis. Comput. 30 (2), 229–243.

Wu, Z., Su, L., Huang, Q., 2019b. Stacked cross refinement network for edge-aware salient object detection. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 7264–7273.

Xie, S., Tu, Z., 2015. Holistically-nested edge detection. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1395–1403.

Yan, Q., Xu, L., Shi, J., Jia, J., 2013. Hierarchical saliency detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1155–1162.

Yang, J., Price, B., Cohen, S., Lee, H., Yang, M.-H., 2016. Object contour detection with a fully convolutional encoder-decoder network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 193–202.

Yang, C., Zhang, L., Lu, H., Ruan, X., Yang, M.-H., 2013. Saliency detection via graph-based manifold ranking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3166–3173.

Zeng, Y., Zhang, P., Zhang, J., Lin, Z., Lu, H., 2019. Towards high-resolution salient object detection. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 7234–7243.

Zhang, J., Ma, S., Sameki, M., Sclaroff, S., Betke, M., Lin, Z., Shen, X., Price, B., Mech, R., 2015. Salient object subitizing. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4045–4054.

Zhang, J., Sclaroff, S., Lin, Z., Shen, X., Price, B., Mech, R., 2016. Unconstrained salient object detection via proposal subset optimization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5733–5742.

Zhang, P., Wang, D., Lu, H., Wang, H., Ruan, X., 2017a. Amulet: Aggregating multi-level convolutional features for salient object detection. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 202–211.

Zhang, P., Wang, D., Lu, H., Wang, H., Yin, B., 2017b. Learning uncertain convolutional features for accurate saliency detection. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 212–221.

Zhang, X., Wang, T., Qi, J., Lu, H., Wang, G., 2018. Progressive attention guided recurrent network for salient object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 714–722.

Zhao, J.-X., Liu, J.-J., Fan, D.-P., Cao, Y., Yang, J., Cheng, M.-M., 2019. EGNet: Edge guidance network for salient object detection. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 8779–8788.

Zhao, R., Ouyang, W., Li, H., Wang, X., 2015. Saliency detection by multi-context deep learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1265–1274.