This article has been accepted for publication in IEEE Transactions on Pattern Analysis and Machine Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3227116

Interactive Object Segmentation with Inside-Outside Guidance

Shiyin Zhang, Shikui Wei, Jun Hao Liew, Kunyang Han,
Yao Zhao, and Yunchao Wei

Abstract—This work explores how to harvest precise object segmentation masks while minimizing the human interaction cost. To achieve this, we propose a simple yet effective interaction scheme, named Inside-Outside Guidance (IOG). Concretely, we leverage an inside point that is clicked near the object center and two outside points at the symmetrical corner locations (top-left and bottom-right or top-right and bottom-left) of an almost-tight bounding box that encloses the target object. The interaction results in a total of one foreground click and four background clicks for segmentation. The advantages of our IOG are four-fold: 1) the two outside points can help remove distractions from other objects or background; 2) the inside point can help eliminate the unrelated regions inside the bounding box; 3) the inside and outside points are easily identified, reducing the confusion raised by the state-of-the-art DEXTR [1] in labeling some extreme samples; 4) it naturally supports additional click annotations for further correction. Despite its simplicity, our IOG not only achieves state-of-the-art performance on several popular benchmarks such as GrabCut [2], PASCAL [3] and MS COCO [4], but also demonstrates strong generalization capability across different domains such as street scenes (Cityscapes [5]), aerial imagery (Rooftop [6] and Agriculture-Vision [7]) and medical images (ssTEM [8]). Code is available at https://github.com/shiyinzhang/Inside-Outside-Guidance.

Index Terms—Interactive segmentation, Image segmentation, Deep learning

1 Introduction

Over the past few years, we have witnessed a revolutionary advancement in semantic [9], [10], [11], [12], [13], [14], [15], [16], [17], [18] and instance segmentation [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29] for different domains, such as general scenes [3], [30], [31], autonomous driving [5], [32], [33], aerial imagery [6], [7], medical diagnosis [8], [34], *etc*. Successful segmentation models are usually built on the shoulders of large volumes of high-quality training data. However, the process to create the pixel-level training data necessary to build these models is often expensive, laborious and time-consuming. Thus, interactive segmentation, which allows the human annotators to quickly extract the object-of-interest by providing some user inputs such as bounding boxes [35], [2], [36] or clicks [37], [38], [39], [40], appears to be an attractive and efficient way to reduce the annotation effort.

Recently, Maninis *et al.* [1] explored the use of extreme points (*left-most, right-most, top, bottom pixels* of an object) for interactive image segmentation. Despite its simplicity, the extreme points have demonstrated fast interactive annotation speed and high segmentation quality across different application domains. Nevertheless, we argue that the clicking paradigm of extreme points also brings some issues: 1) annotating extreme points requires users to carefully click at the object boundaries, which usually consumes much more time as compared to the common clicking setting where users can click at *any* of the interior and exterior of object regions; 2) the annotation process can sometimes be confusing when multiple extreme points appear at similar spatial locations (pencil in Figure (a)) or when there are unrelated objects or background lying inside the target object (dog in Figure (a)).

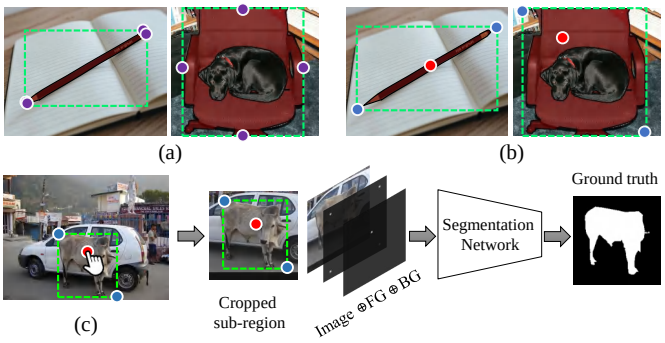

Fig. 1. **Comparison of clicking paradigm:** (a) User inputs of DEXTR [1]. (b) User inputs of the proposed IOG-Click method. (c) **Overview of our IOG-Click framework**. Our IOG paradigm consists of 3 clicks, *i.e.*, an interior click and two exterior clicks. Based on these 3 clicks, we crop the RGB image and concatenate it with the foreground and background click maps to form the input for our segmentation network to predict the segmentation mask.

To tackle the aforementioned issues as well as to promote the effectiveness and efficiency of the interactive process, we propose an approach named Inside-Outside Guidance (IOG), which requires only **three** points (an inside point and two outside points) to indicate the target object. Specifically, the inside point usually locates around the center of the object instance while the two outside ones can be clicked at any symmetrical corner locations of a tight bounding box enclosing the target instance (either the *top-left and bottom-right* or *top-right and bottom-left* pixels). Figure (b) shows two examples of our proposed labeling scheme. Similar to [1], our IOG relaxes the generated bounding box by several

- *S. Zhang, S. Wei, K. Han, Y. Zhao and Y. Wei are with the Institute of Information Science, Beijing Jiaotong University and Beijing Key Laboratory of Advanced Information Science and Network Technology. Email: {shiyinzhang, shkwei, kunyanghan, yzhao, yunchao.wei}@bjtu.edu.cn. Y. Wei is the corresponding author.*
- *J.H. Liew is with National University of Singapore, 117583, Singapore. Email: liewjunhao@u.nus.edu.*

pixels before cropping from the input image to include context. This results in a total of one foreground and four background clicks (two clicked outside points and two additional inferred ones based on the bounding box), which are then encoded as foreground/ background localization heatmaps and concatenated with the cropped image for training the segmentation network. The overview of our IOG is shown in Figure (c).

Our IOG strategy not only improves the annotation speed by reducing the confusion raised by [1], but also naturally supports annotation of additional points at the erroneous regions for further refinement. We perform extensive experiments on GrabCut [2], PASCAL [3] and MS COCO [30] to demonstrate the effectiveness of our IOG as an annotation tool. In particular, given only three points, our IOG achieves 93.2% mIoU score on PASCAL, which is the new state-of-the-art. Our IOG can further improve the performance to to 94.4% by applying the 4th click for correction.

In addition, we also show that our model generalizes well in cross-domain annotation, where our PASCAL- or COCO-trained model produces high quality segmentation masks when annotating street scenes [5], aerial imagery [6], [7] and medical images [8] **without** the need of fine-tuning. Beyond this, we also propose a simple two-stage solution that enables our IOG to harvest precise instance segmentation masks from the off-the-shelf datasets with bounding box annotations such as ImageNet [4] and Open Images [41] without any human interaction. We hope this work can significantly benefit the future researchers in collecting large-scale pixel-level annotations.

The contributions of this work are summarized as follows:

- We introduce a new Inside-Outside Guidance (IOG) scheme to tackle the interactive object segmentation task. Despite its simplicity, our IOG achieves the state-of-the-art performance on all popular benchmarks, and shows strong generalization ability for multiple cross-domain benchmarks.
- We investigate several principle adjustment strategies to make our IOG be better applied to real-world data annotation scenarios. Extensive user studies are performed to verify the effectiveness of the proposed adjustment strategies.
- Our IOG can be employed to produce accurate instance masks for existing datasets with off-the-shelf bounding box annotations. Benefiting from the superiority of our IOG, we contribute a new dataset, *i.e.*, `Pixel-ImageNet`[1], which includes 0.615M instance masks from 1K classes.

This paper is an extension of our previous conference version [42]. Comparing to the initial version, this work makes the following improvements. First, we extend the inside guidance of our IOG from click- to scribble-paradigm, which we call **IOG-Scribble**. We show that significant improvements in accuracy can be obtained without incurring too much annotation burden. Besides, we additionally introduce a collaborative training strategy that utilizes coarsely and finely annotated data to further improve the segmentation quality along object boundaries. Second, more user studies are performed to examine how to adjust the interaction paradigm for practical annotation scenarios. Third, we perform additional experiments on the much challenging Agriculture-Vision dataset [7], where we show that our IOG greatly outperforms the baseline even on images with unclear boundaries. Lastly, we contribute Intelligent Pixel Annotation Tool (IPAT) [2], a web-based annotation interface based on our IOG, which plays an important role in annotating our `Pixel-ImageNet` dataset. We hope this can help reduce the annotation cost and time for future segmentation benchmarks.

1. https://github.com/shiyinzhang/Pixel-ImageNet
2. https://github.com/KunyangHan/interactive-segmentation-editor

2 Related Work

Interactive Segmentation: Prior to deep learning, early methods mainly pose interactive segmentation as an optimization problem to separate the foreground and background pixels. For example, Boykov and Jolly [43] formulate interactive segmentation task as a graph cut optimization problem. GrabCut [2] develops an iterative optimization technique and employed bounding boxes to guide the segmentation process. Bai and Sapiro [44] adopts weighted geodesic distance to classify each pixel into foreground and background class. Nevertheless, all these conventional approaches typically struggle in the case of complicated scenes (*e.g.*, substantial overlapping between foreground and background appearances, complex illumination condition *etc.*) due to the use of low-level features.

Given the success of deep convolutional neural networks (CNN) in semantic segmentation task, recent interactive segmentation have been mainly driven by CNN-based approaches [35], [45], [46], [37], [38], [1], [47], [48], [49], [50], [51], [52], [53], [54]. According to the human inputs, the interactive segmentation approached can be roughly divided into click-based [37], [38], [1], [47], [48], [49], [50], [51] and polygon-based [52], [53], [54].

Click-based methods. iFCN [37] is first proposed to conduct interactive segmentation by guiding a CNN with positive (foreground) and negative (background) points clicked by the users. RIS-Net [38] improves the iFCN by augmenting a local context branch. Maninis *et al.* [1] propose DEXTR that leverages only 4 extreme points for segmentation. MultiSeg [49] presents a scale-diverse interactive segmentation network, which can generate diverse yet plausible segmentation results conforming to the given user input. BRS [50] adopts back-propagation refinement strategy to correct the mislabeled pixels. However, the continuous steps of forward and backward pass often lead to high computational budget per click, which is time consuming. f-BRS [51] improves the BRS by requiring only a small part of a network for forward and backward passes, which greatly saves the calculation consumption. Andriluka *et al.* [46] further extend interaction scheme from instance-level to full-image level. *Polygon-based methods.* Polygon-RNN [52] poses interactive segmentation as a polygon prediction task, which is convenient for users to interact with. Polygon-RNN allows users to correct the vertex of the polygon for more accurate result. Polygon-RNN++ [53] improves the Polygon-RNN by updating the network architecture, using reinforcement learning as train strategy, and increasing the resolution of the output polygon. Curve-GCN [54] uses graph convolutional networks to predict the vertices of the polygon.

Semantic Segmentation: The success of CNN-based interactive segmentation algorithms [37], [38], [39], [1] have benefited significantly from the development of semantic segmentation architecture, especially, FCN [9], DeepLab series [13], [14], [15], PSP [11], CCNet [16] and SPGNet [17]. Several recent newer deep architectures studies have gained higher predictive performance. Typically, the spatial pyramid pooling structure, such as Pyramid Scene Parsing (PSP) [55] module and Atrous Spatial Pyramid Pooling (ASPP) [56] module, utilizes multi-scale pooling layers with diverse steps to combine local information about global information. Encoder-decoder structure [57], [58] has also advanced the

state-of-the-art by combining multi-dimensional features. The encoder module acquires stronger semantic information but reduces the size of the feature. The decoder module aims at recovering the spatial information. Cascaded structure [59] contains multi-stage which provides current practice for performance refining. All the stages have gradually facilitated the prediction from coarse to fine, by end-to-end training. In this work, we investigate which type of network is more suitable for conducting interactive segmentation tasks and choose to adopt a coarse-to-fine network structure [60] as the backbone of our IOG method. We experimentally validate our choice can further boost the accuracy of interactive segmentation by a large margin.

Instance Segmentation: Our work is also related to instance segmentation task, which operates on the detection and segmentation of individual objects. We also focus on individual objects of each image. There are the following examples of the instance segmentation. Two-stage instance segmentation methods such as [29], [28], [61], [62], first generate a set of region-of-interests (ROIs) and then segment them into masks. One-stage instance segmentation methods such as [63], [26], [64], [65], [66] generate position sensitive maps to acquire the final segmentation mask. However, instance segmentation and interactive segmentation have two fundamental differences. First, interactive segmentation requires the guidance from human. We can decide what we want to segment according to the user's interaction. Second, interactive segmentation often supports further refinement if the predictions are incorrect.

Weakly Supervised Segmentation: Among many alternatives in addressing the expensive pixel-level annotations, weakly supervised learning has been extensively studied in the literature. Particularly, image-level labels [67], [68], [69], [70], [71], [72], points [73], [74], bounding boxes [75], [76], scribbles [77], [78], [79] have been employed as guidance to supervise the training of semantic segmentation networks. Different from these methods, our proposed IOG still relies on fully annotated masks as supervision and utilizes three additional points as the guidance to produce the segmentation mask of the target object.

Other Works on Interactive Annotation: Some works attempt to improve annotation efficiency from other perspectives. Interactive full image segmentation [80] aims at segmenting all object and stuff regions simultaneously in one image. For each object in one image, the other objects are used as background information. Rupprecht *et al.* [81], [82] use natural language as feedback for interactive segmentation correction. They combine an interactive segmentation framework with a language module whose input is like "there is a man riding in the corner". Finally, several works [83] propose an interactive framework for annotating 3D object by drawing scribble in 2D views.

3 METHOD

3.1 Inside-Outside Guidance

Our Inside-Outside Guidance (IOG) clicking paradigm consists of two components: inside guidance and outside guidance. Depending on the interaction mode, the inside guidance contains two possible instantiations, *i.e.*, click and scribble, which we refer to **IOG-Click** and **IOG-Scribble**, respectively. The click-based guidance consists of three points: an interior click (inside point) located roughly at the object center and two exterior clicks (outside points) at any symmetrical corner locations (either *top-left* and *bottom-right* or *top-right* and *bottom-left*) that form an almost-tight bounding box enclosing the target-of-interest. On the other hand, compared with the click-based paradigm, scribble-based guidance can often achieve better segmentation accuracy given slight additional annotation overhead. In particular, it consists of two exterior clicks (same as the click-based guidance) and coarse scribble(s) marked across the object-to-segment. In this way, the two exterior clicks, together with two additional inferred ones based on the generated bounding box, provide an "outside" guidance (indicating the background regions) while the interior click or scribble gives an "inside" guidance (indicating the foreground regions), thus giving the name *Inside-Outside Guidance (IOG)*. Figure 2 shows an example of our IOG clicking paradigm.

Outside Guidance: The outside guidance is formulated by the corners of the bounding box enclosing the object. However, it was previously reported that drawing a tight box can be time consuming ([84] reported 25.5s for drawing one box on ImageNet [4][3]). This is due to the difficulty of clicking on the corners of an imaginary box where these corners are often not on the object [87]. Thus, several adjustments are usually needed to ensure the resulting box is tight. However, with some simple modifications to the annotation interface, such as using a horizontal and a vertical guide line to make the box visible when clicking on a corner, the burden of drawing a bounding box can be largely relieved as shown in Figure 2(a)-(b). Moreover, in our case, we do not necessarily need a tight bounding box where an almost-tight box usually suffices. In our user study, we observe that drawing a bounding box typically take about 6.7s with the help of the guide lines.

Inside Guidance: The main purpose of inside guidance is to disambiguate the segmentation target from its surrounding background since there could be multiple objects within the same box. In this work, we provide two possible instantiations of inside guidance, *i.e.*, click and scribble. The click-based inside guidance is formulated as an interior click located around the object center whereas the scribble-based inside guidance represents the object-of-interest with a coarse scribble. The corresponding IOG are respectively denoted as IOG-Click and IOG-Scribble.

(a) IOG-Click: To simulate clicks annotated by human annotators, we propose to sample the inside point at the location that is furthest away from the object boundaries. In particular, let $\mathcal{F}$ and $\mathcal{B}$ denote the pixels belonging to the foreground and background, respectively, we first compute a distance map D based on Euclidean distance transformation as follows:

$$D_i = \min_{\forall j \in \mathcal{B}} \text{dist}(i, j), \tag{1}$$

where D_i refers to the value of D at pixel location i while $\text{dist}(i, j)$ denotes the Euclidean distance between pixel locations i and j. Then, the interior click is sampled at the location $k = \arg\max_{\forall i \in \mathcal{F}} D_i$. The validity of such sampling scheme is verified in Section 4.5 by comparing with the actual interior click collected from real users. Note that annotating the inside point is very fast, taking about 1.5s in our user study.

Compared with existing click-based [37], [1] and box-based [35] interactive segmentation approaches, our proposed IOG has the best of both worlds: (i) **flexibility**: since the annotated three points are encoded as foreground and background clicks, our

3. Some papers reported much faster timings (*e.g.*[85] reported 10.21s while [86] reported 7.0s). However, [87] argue that the annotated boxes are of low quality (not tight around the object).

This article has been accepted for publication in IEEE Transactions on Pattern Analysis and Machine Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3227116

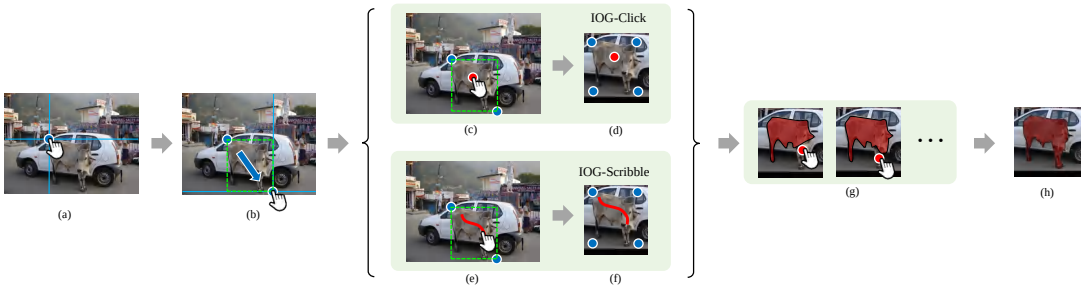

Fig. 2. **Inside-Outside Guidance.** (a) The vertical and horizontal guide lines are used to assist the user in clicking on the corner of an imaginary box enclosing the object. (b) A box is generated on-the-fly when the user moves the cursor. (c)(d) and (e)(f) show two different instantiations of inside guidance, *i.e.*, **IOG-Click** and **IOG-Scribble**. (c) For IOG-Click, an interior click is placed around the object center whereas for (e) IOG-Scribble, a scribble is placed across the object-of-interest. (d)(f) The box is relaxed by several pixels before cropping to include context. The interior click/scribble (red) with four exterior clicks (two clicked corners and two automatically inferred ones) (blue) constitute our IOG that encode the foreground and background regions, respectively. (g) Our method naturally supports additional clicks annotation for further refinement.

IOG naturally supports additional clicks annotations for further correction (Figure 2(e)); (ii) **more information**: our approach encodes more prior information about the object, including the location of hard background and the rough size of the target.

(b) IOG-Scribble: We additionally provide an alternative interaction strategy that employs scribbles to indicate the object-of-interest. Compared with IOG-Click, scribbles provide more prior information about the target, such as the spatial extent and rough shape of the object (Figure 2). To simulate scribbles for training, we propose to sample the inside scribble as the line crossing most part of the object [88]. Similar to IOG-Click, we also compare with real user-drawn scribbles to verify its validity. Although IOG-Scribble incurs slight additional annotation overhead ($3s$ in total according to our user study), it brings significant improvement in segmentation quality as demonstrated in the experimental section.

Construct Click & Scribble Representations: We use the same click representation as DEXTR[1] by centering a 2D Gaussian around each click, creating two separate heatmaps for foreground and background clicks. For IOG-Scribble, we directly employ the scribble mask as the foreground heatmap without additional processing. In particular, the scribble mask M can be computed as follows:

$$M_i = \begin{cases} 1, & i \in \mathcal{S}, \\ 0, & i \notin \mathcal{S}, \end{cases} \tag{2}$$

where $\mathcal{S}$ denotes the set of pixels belonging to the scribble. The resulting heatmaps are then concatenated with the RGB input image to form a 5-channel input for the segmentation network. Similar to [1], the bounding box is first relaxed by several pixels to include context, followed by cropping to focus on the object-of-interest (Figure 2(d)).

3.2 Segmentation Network

Here, we discuss the architectural design of our segmentation network. We employ a ResNet-50 [90]-based DeepLabv3+ [15] as our starting point and we already observe decent segmentation performance (90.0% IoU on PASCAL), demonstrating the effectiveness of our proposed IOG. Nevertheless, closer inspection on the segmentation quality reveals that segmentation errors mostly occur along the object boundaries as shown in Figure 4. Simply replacing the backbone with a deeper network such as ResNet-101 only brings marginal improvement (Vanilla IOG in Figure 6 right), suggesting some architectural modifications have to be made to ensure the network focuses on refining the inaccurate segmentation along the object boundaries.

In this work, we propose to adopt a coarse-to-fine design for addressing the aforementioned issue (Figure 3). In particular, we employ a cascaded structure similar to [60] which was originally proposed for human pose estimation task. Specifically, the segmentation network consists of two subnetworks. The first subnetwork, CoarseNet applies an FPN-like design [10] that progressively fuses the semantic information from the deeper layers with low-level details from the earlier layers via lateral connections. The CoarseNet consists of three components, *i.e.*, backbone, global context module, and decoder. Different from [60], we append a Pyramid Scene Parsing (PSP) module [11] as the global context module at the deepest layer of the backbone for enriching the representation with global contextual information. Given a coarse prediction from the CoarseNet, the second subnetwork, FineNet aims at recovering the missing boundary details. This is achieved with a multi-scale fusion structure that fuses the information across different levels in the CoarseNet via upsampling and concatenation operations. Similar to [60], we also apply more convolution blocks for features at deeper layers for better trade-off between the accuracy and efficiency. We refer the readers to the supplementary materials for more details. Note that we do **not** claim any novelty in the network design. Instead, our contribution lies in the finding that a coarse-to-fine structure is necessary for obtaining more precise segmentation masks whereas stacking more layers does not. We believe other coarse-to-fine structure might also work and we leave it as our future works.

Composition of Input: To generate inputs for the segmentation network, we first employ the Inside-Outside guidance to annotate the target object. Then, we crop the RGB image based on the bounding box and resize it to 512×512. Finally, we concatenate the cropped RGB image with the two-channel Gaussian heatmaps generated from the click or scribble representations in Section 3.1. The detailed pipeline can be found in Figure 3.

Training and Testing: Our segmentation network is trained end-to-end using binary cross-entropy loss. In addition, we also apply side losses at each level of CoarseNet as a form of deep supervision [60]. During inference, the segmentation mask can be obtained by simply thresholding the final network prediction. Since our approach does not involve any post-processing, it is extremely fast, where a single forward pass on a ResNet-101

This article has been accepted for publication in IEEE Transactions on Pattern Analysis and Machine Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3227116

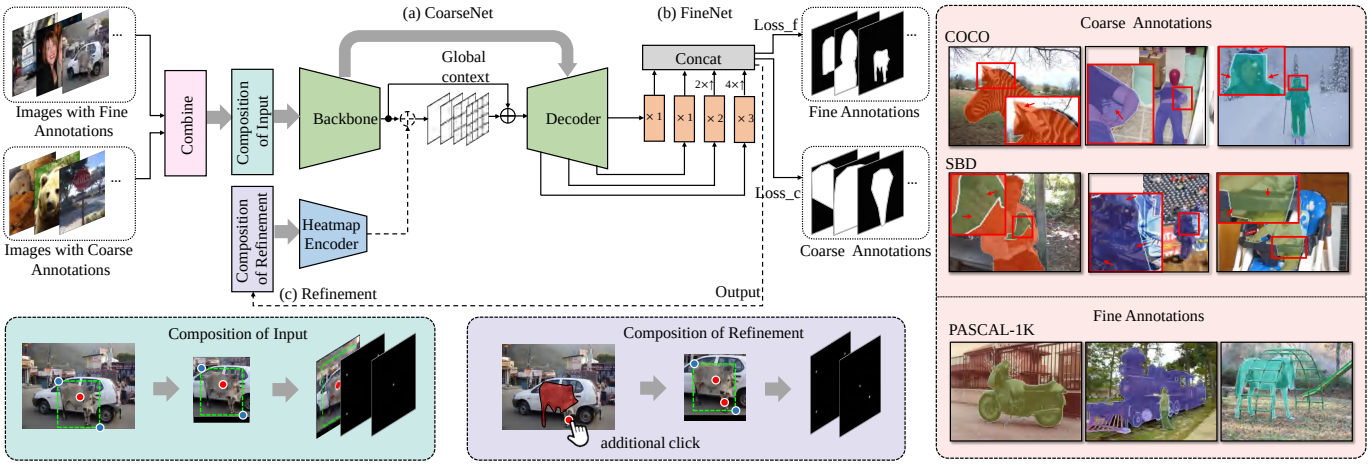

Fig. 3. **Network architecture.** (a)-(b) Our segmentation network adopts a coarse-to-fine structure similar to [60], augmented with a pyramid scene parsing (PSP) module [11] for aggregating global contextual information. (c) We also append a lightweight branch before the PSP module to accept the additional clicks input for interactive refinement. To improve the segmentation quality along object boundaries, our collaborative training strategy augments the coarsely annotated samples (*e.g.*, SBD [89] or COCO [30]) with finely annotated samples (*e.g.*, PASCAL VOC [3]) within each training batch, progressively increasing the ratio of finely annotated samples throughout the training process.

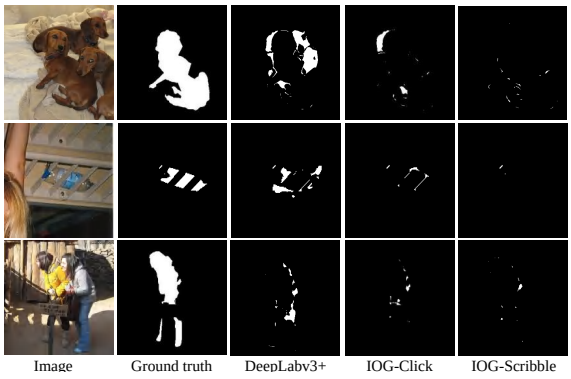

Fig. 4. **Qualitative comparison in terms of segmentation errors.** Note that the segmentation errors mostly occur along the object boundaries when using DeepLabv3+ [15] as backbone whereas our coarse-to-fine structure produces precise boundaries.

backbone typically requires only 20 ms on a Nvidia GeForce GTX 1080 GPU. It is thus well-suited for practical interactive image segmentation application.

3.3 Refinement

Our IOG naturally supports the scheme of adding additional foreground or background clicks for further correction if annotators are not satisfied with the current segmentation output. To achieve this target, we append a lightweight branch (heatmap encoder) before the PSP module to accept the two-channel Gaussian heatmaps encoding all the foreground and background clicks (Figure 3(c)). We empirically found that this setting not only works better than modifying the inputs at the beginning of the segmentation network, but also runs much faster since the encoder features only needs to be computed once. During training, we adopt an iterative training strategy to simulate the interactive process where an additional click is introduced to the erroneous regions by the user for correction. More specifically, depending on the interaction mode, either three clicks (IOG-Click) or a scribble with two exterior clicks (IOG-Scribble) are used to first obtain an initial segmentation mask. A new click is then added to the center of the largest erroneous region and second forward pass is conducted. Note that our IOG-Scribble also supports adding of corrective scribbles, but we only consider clicks for refinement in this work for simplicity. Results presented in Section 4.3 shows that such iterative training strategy is necessary.

Composition of Refinement: The corrective click is added to the previous clicks set before generating the updated two-channel Gaussian heatmaps, which are then fed into the heatmap encoder (Figure 3).

3.4 Collaborative Training Strategy

Existing interactive object segmentation algorithms [1], [49], [91] are typically trained on the combination of a finely annotated PASCAL VOC [3] and a coarsely annotated SBD [89] dataset (Fig. 3). However, we notice that the coarser annotations often lead to classification confusion along the boundary regions. While introducing an additional edge refinement branch could alleviate this issue, it comes at the cost of increased computation. Instead, we proposed a simple collaborative training strategy to improve the segmentation quality along the object boundaries without incurring extra computation. Specifically, we combine coarsely annotated images and finely annotated images within each training batch, and progressively increase the ratio of finely annotated samples α throughout the training process:

$$I_{batch} = I_{\text{coarse}} \cup I_{\text{fine}} \tag{3}$$

$$\alpha = \frac{|I_{\text{fine}}|}{|I_{\text{coarse}}|}, \quad \alpha \in [0, 1] \tag{4}$$

where I_{coarse} and I_{fine} refer to the image set with coarse and fine annotations, respectively. The final loss $\mathcal{L}$ can then be computed as follows:

$$\mathcal{L} = \beta \mathcal{L}_{\text{coarse}} + (1 - \beta)\mathcal{L}_{\text{fine}}, \quad \beta \in [0, 1] \tag{5}$$

where $\mathcal{L}_{\text{coarse}}$ and $\mathcal{L}_{\text{fine}}$ correspondingly denote the loss of coarsely and finely annotated samples. In our experiments, we use PASCAL-1k [4] as the finely annotated dataset to assist training on PASCAL-10k and COCO datasets.

4. We denote the PASCAL train set augmented with additional labels from SBD [89] and the one without SBD labels as PASCAL-10k (10,582 images) and PASCAL-1k (1,464 images), respectively.

Methods	Number of Clicks P@85%	Number of Clicks G@90%	IoU(%) @ 4 clicks P	IoU(%) @ 4 clicks G
Graph cut [ICCV01] [43]	> 20	> 20	41.1	59.3
Random walker [TPAMI06] [92]	16.1	15	55.1	56.9
Geodesic matting [ICCV07] [44]	> 20	> 20	45.9	55.6
iFCN [CVPR16] [35]	8.7	7.5	75.2	84.0
RIS-Net [ICCV17] [38]	5.7	6	80.7	85.0
DEXTR [CVPR18] [1]	4	4	91.5	94.4
ITIS [BMVC18] [39]	3.4	5.7	-	-
CMG [CVPR19] [93]	3.58	3.62	-	-
BRS [CVPR19] [50]	6.59	3.60	-	-
F-BRS-B [CVPR20] [51]	4.81	2.72	-	-
FCA-Net [CVPR20] [47]	2.14	2.96	-	-
IOG (**ours**, outside only)	2	2	90.9	91.4
IOG-Click (**ours**)	3	3	93.2	96.3
IOG-Click† (**ours**)	4	4	94.4	96.9

TABLE 1
Comparison with the state-of-the-art methods on PASCAL (P) and GrabCut (G) in terms of the number of clicks to reach a certain IoU and in terms of quality at 4 clicks. † denotes our IOG with refinement.

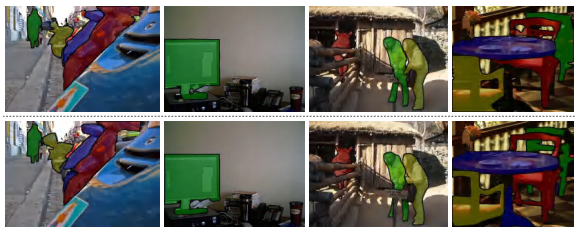

Fig. 5. Comparison between IOG-Click (first row) and IOG-Scribble (second row). On the basis of not increasing too much annotation effort, our IOG-Scribble achieves better segmentation quality than IOG-Click.

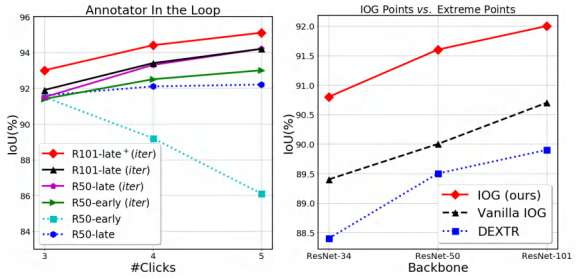

Fig. 6. (left) The effect of iterative training for interactive refinement. "early" and "late" denote adding clicks input to the beginning or intermediate layer of the network, respectively. "iter" implies iterative training (Section 3.3) while "+" denotes training on larger dataset (PASCAL-10k). (right) Comparison between IOG points and extreme points.

Backbone	Context	FineNet	Side losses	Dataset	IoU
ResNet-50	✗	✓	✓	PASCAL-1k	91.2
ResNet-50	✓	✗	✓	PASCAL-1k	90.8
ResNet-50	✓	✓	✗	PASCAL-1k	90.6
ResNet-50	✓	✓	✓	PASCAL-1k	91.6
ResNet-50	✓	✓	✓	PASCAL-10k	92.8
ResNet-101	✓	✓	✓	PASCAL-1k	92.0
ResNet-101	✓	✓	✓	PASCAL-10k	93.2

TABLE 2
Ablation Study. Justification of each component in the segmentation network on the PASCAL VOC 2012 *val* set.

4 Experiments

4.1 Comparison with the State-of-the-art Approaches

We conduct extensive experiments on **11** publicly available benchmarks, including PASCAL [3], GrabCut [2], COCO [30], ImageNet [4], Open Images [4], Cityscapes [5], Rooftop [6], Agriculture-Vision [7], ssTEM [8], Pascal-Context [94], and COCO-Stuff [95], to demonstrate the effectiveness and the generalization capabilities of our IOG. We choose ResNet-50 and ResNet-101 as the two backbones of the IOG for fair comparison with previous approaches. Following the common practice [1], we employ PASCAL as the main benchmark to verify the importance of each component proposed in our IOG.

4.2 Implementation Details

Training and Testing Details: IOG is trained on PASCAL 2012 Segmentation for a maximum of 100 epochs or on MS COCO 2014 for a maximum of 10 epochs. We acquire the results from the best performing epoch. For PASCAL, the batch size is set to 5 whereas for COCO, we train on 2 GPUs with an effective batch size of 10. For COCO, we also construct a set of "void" pixels around the boundaries of the ground truth masks and ignore them during training. The learning rate, momentum and weight decay are set to 10^{-8}, 0.9 and 5×10^{-4}, respectively. When using collaborative training strategy, α gradually increases from 0 to 0.3 for PASCAL whereas for COCO, we fix α to 1. We set β to 0.5.

Simulated Outside Click: We use the ground truth masks to generate the inside-outside points for training. For outside points, we take the corners of the bounding-box extracted from ground truth masks and relax by 10 pixels to simulate a loosen box provided by real users.

Simulated Inside Click: For the inside click, we sample a click that is furthest from the object boundaries. To simulate the randomness in manual annotation, we apply random perturbation during training to improve the model's robustness to real user clicks. The effects of perturbation are studied in Section 4.5.

Simulated Inside Scribble: For the inside scribble, we sample a scribble that covers the object following [88]. In cases when the object is split into several components (*e.g.*, due to occlusion), we additionally sample a scribble for each object component. Different from the inside click, we do not apply random perturbation to the simulated scribble. When tested with human annotations, we fine-tune the model using 30% of PASCAL-1k with manually-drawn scribbles collected from real users.

We compare our IOG with the state-of-the-art approaches on two popular benchmarks, *i.e.*, PASCAL and GrabCut. The results are summarized in Table 1. Here, we only report the performance of click-based IOG for fair comparison. We first notice that when evaluated in term of number of clicks needed to reach a certain performance (*e.g.*, 85% IoU in PASCAL), our IOG with only outside guidance (please refer to Section 4.6 for more details) already outperforms all the state-of-the-art methods. When inspecting closer, we further observe that our complete IOG model with only 3 clicks performs significantly better than the best-performing models by more than 1.7% and 1.9% IoU on PASCAL and GrabCut, respectively. When allowing iterative refinement (*i.e.*, from 3 to 4 clicks), the performance can be further enhanced to 94.4% and 96.9%, which well demonstrates the effectiveness of our IOG in handling the additional user inputs for further correction.

4.3 Ablation Study

Justification of Each Component of IOG: We perform ablation experiments on PASCAL VOC *val* set to validate the effectiveness of each component in our segmentation network. Particularly, we quantitatively justify various design choices, including the different backbones (ResNet-50 *v.s.* ResNet-101), different number of

Train	Test	DEXTR	IOG-Click	IOG-Scribble
PASCAL	COCO MVal(seen)	79.9%	81.7%	86.4%
PASCAL	COCO MVal(unseen)	80.3%	82.1%	86.9%
PASCAL	COCO MVal	80.1%	81.9%	86.6%
COCO	COCO MVal	82.1%	85.2%	88.9%
COCO	PASCAL	87.8%	91.6%	92.3%
PASCAL	PASCAL	89.8%	93.2%	96.4%

TABLE 3
Comparison in terms of generalization ability between the state-of-the-art DEXTR and our IOG. Top and bottom rows correspond to *Unseen Classes* and *Generalization* settings, respectively.

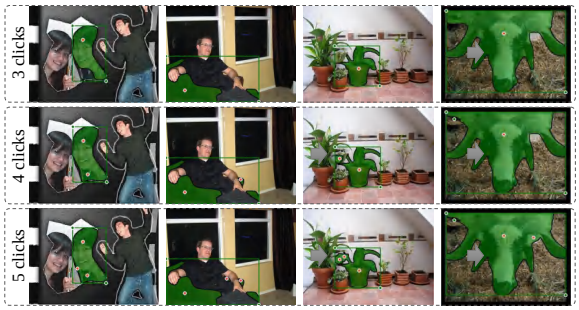

Fig. 7. **Interactive refinement.** Our proposed IOG supports interactive adding of new clicks for further refinement.

training images (PASCAL-1K *v.s.* PASCAL-10K), inclusion of PSP module for global contextual information (Context), FineNet and the use of side losses for training. As shown in Table 2, "Context", "FineNet" and "Side losses" can respectively lead to performance boost of 0.4%, 0.8% and 1.0% under the setting of ResNet-50 and PASCAL-1K. When augmenting additional labels from SBD (PASCAL-10k), the performance can be further improved from 91.6% to 92.8%. Finally, we obtain the state-of-the-art performance when replacing the backbone with ResNet-101 (93.2%).

Iterative Training for Interactive Refinement: In the previous section, we have demonstrated the effectiveness of our IOG under the default setting when only 3 clicks are provided. Next, we examine the case when the user is not satisfied with the result and wants to annotate additional clicks for correction. Specifically, we progressively add a new click to the center of the largest erroneous regions similar to [37], [39]. The results are summarized in Figure 6 (left). We can observe that: 1) additional clicks do not bring significant performance gains without iterative training, demonstrating the importance of iterative training for interactive refinement; 2) adding the clicks to the intermediate layers of the segmentation network (Section 3.3) is more effective than modifying the inputs at the beginning of the network. An interesting observation is that adding clicks to the beginning of the model without iterative training will lead to performance degradation. One possible reason is that the inside points always locate around the object center whereas the newly added correction clicks are

Methods	PASCAL (IoU)	
	w/o Refinement	w/ Refinement
IOG-Click	93.2	94.4
IOG-Scribble	96.4	96.8

TABLE 4
Comparison between IOG-Click and IOG-Scribble on PASCAL *Val* Set.

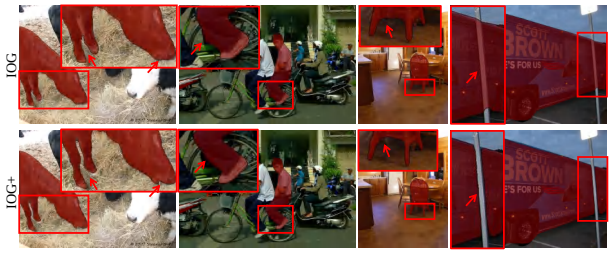

Fig. 8. Comparison between IOG and IOG with collaborative training strategy (IOG+). The IOG+ achieves better boundary segmentation quality than IOG.

usually distributed near the object boundaries, which confuses the trained model and harms the performance. Some qualitative examples of interactive refinement can be found in Figure 7.

IOG Points *v.s.* Extreme Points: We study the performance of our proposed IOG points when compared with the extreme points used in DEXTR. For fair comparison, we use the released code[5] and re-train DEXTR using DeepLabv3+ [15] as the fully convolutional architecture on PASCAL-1K. All the models are pre-trained only on ImageNet [4]. We conduct experiments using three different backbones, *i.e.*, ResNet-34, ResNet-50 and ResNet-101, to validate the robustness of the proposed method. As shown in Figure 6 (right), our proposed IOG points consistently outperform the extreme points given the same network architecture (Vanilla IOG *vs.* DEXTR). When using a coarse-to-fine network structure (Section 3.2), we can see that our IOG significantly outperforms the baselines by a large margin. Interestingly, our IOG with ResNet-34 as backbone already surpasses the state-of-the-art DEXTR using ResNet-101, demonstrating the effectiveness of the proposed IOG over the extreme points.

IOG-Click *v.s.* IOG-Scribble: We also compare the different interaction paradigm of IOG (click *v.s.* scribble). Both models are trained on PASCAL-10k and evaluated on PASCAL VOC *val* set for fair comparison. As shown in Table 4, IOG-Scribble significantly outperforms IOG-Click by 3.2%, demonstrating the effectiveness of scribbles in encoding more prior object information (*e.g.*, spatial extent and rough object shape) to assist segmentation task. More interestingly, IOG-Scribble without refinement attains much higher accuracy than that of IOG-Click with refinement (96.4% *v.s.* 94.4%). Despite the already high accuracy, we notice that the performance can be further boosted to 96.8% when applying iterative refinement (*i.e.*, adding 4th click). Although drawing a scribble introduces slight annotation overhead compared to clicking ($3s$ for scribbling *v.s.* $1.5s$ for clicking), we argue that the performance boost makes it more effective for practical application. Some qualitative comparison between IOG-Click and IOG-Scribble can be found in Figure 5. Please refer to our supplementary materials for more examples.

Collaborative Training Strategy: We also evaluate the performance of IOG trained with collaborative training strategy (IOG+). We use Boundary-IoU [96] metric to evaluate the segmentation quality on edge regions. As shown in Table 6, IOG+ can respectively lead to a performance boost of 0.8% and 1.0% on PASCAL and COCO when click is employed as the inside guidance. When choosing scribble as the inside guidance, the performance can be further enhanced from 84.5% to 85.0% in PASCAL and 84.3% to

5. https://github.com/scaelles/DEXTR-PyTorch

Method	r	Simulated Inside Guidance	Manual Inside Guidance
IOG-Click	0	93.2	90.8
	10	92.9	91.6
	30	92.8	92.3
	50	92.0	92.0
IOG-Scribble	N.A.	96.4	90.7
IOG-Scribble‡	N.A.	95.9	94.4

TABLE 5
Robustness to user variance on inside guidance. r denotes the radius of perturbation applied on the inside click during training. All the models are trained on PASCAL-10k and tested on PASCAL *val* set. ‡ implies that the IOG-Scribble model is fine-tuned on 30% of PASCAL-1k training set with manually labeled scribbles.

Method	Inside guidance	dataset	IoU(%)	Boundary-IoU(%)
IOG	Click	PASCAL	93.2	78.8
IOG+	Click	PASCAL	93.4	79.6
IOG	Click	COCO	85.2	79.9
IOG+	Click	COCO	85.7	80.9
IOG	Scribble	PASCAL	96.4	84.5
IOG+	Scribble	PASCAL	96.6	85.0
IOG	Scribble	COCO	88.9	84.3
IOG+	Scribble	COCO	89.9	85.9

TABLE 6
Collaborative training strategy. IOG+ implies that the IOG model is trained with the collaborative training strategy.

85.9% in COCO. Some qualitative comparisons between IOG and IOG+ can be found in Figure 8.

4.4 Generalization

To verify the generalization capability of our IOG, we perform extensive experiments on both in-domain and cross-domain datasets and compare with the state-of-the-art approaches.

4.4.1 *In-domain*

For in-domain datasets, we compare with the state-of-the-art DEXTR on both things categories and stuff categories. For things categories, we follow the setting in [1], and compare the performance on two benchmarks, *i.e.*, PASCAL and COCO mini-val (MVal). For stuff categories, we perform qualitative analysis on PASCAL-Context [94] and COCO-Stuff [95].

PASCAL↔COCO: Following [1], we inspect the model's generalization capability from two perspectives: (1) generalization to unseen classes (*Unseen Classes* setting); (2) generalization to other dataset (*Generalization* setting). For the *Unseen Classes* setting, we leverage the model trained on PASCAL and evaluate its IoU on COCO MVal *seen* (*i.e.*, images with the same categories as PASCAL) and COCO MVal *unseen* (*i.e.*, images with different categories as PASCAL). For the *Generalization* setting, we train the model on PASCAL (or COCO) and evaluate the performance on COCO MVal (or PASCAL), regardless of the testing categories. As shown in Table 3, our IOG makes consistent improvements over DEXTR on various settings despite using only 3 clicks. We also show that IOG-Scribble significantly outperforms IOG-Click with a performance boost of 4.8% on unseen class.

Things→Stuff: In Figure 11, we show some qualitative results of our IOG fine-tuned on PASCAL-Context [94] and COCO-Stuff [95] to verify the performance of our IOG when segmenting "stuff" categories. The results show that our IOG generalizes well to background classes too.

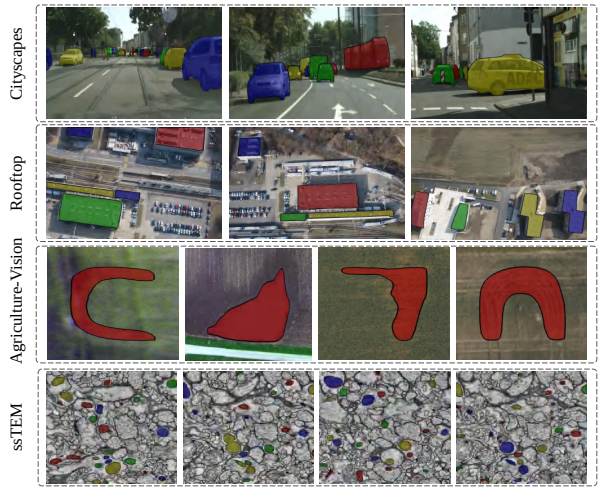

Fig. 9. **Cross-domain performance.** Qualitative results of our IOG on Cityscapes, Agricultural-Vision, Rooftop, and ssTEM.

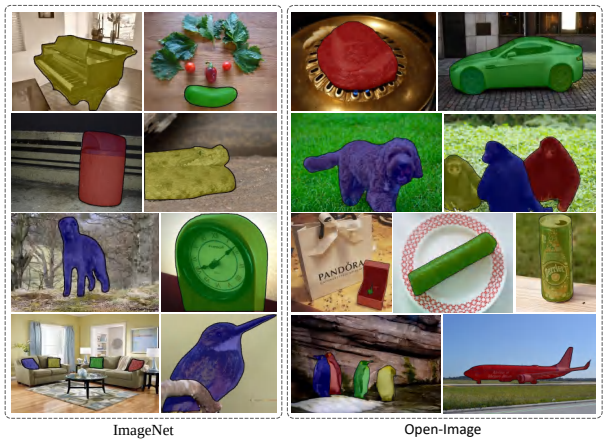

Fig. 10. Qualitative results on ImageNet (top) and Open Image (bottom) using our proposed 2-stage approach. Note that only bounding box annotations are provided.

4.4.2 *Cross-domain*

In the previous section, we have demonstrated the generalization capability of our IOG on unseen classes and across different datasets (train on PASCAL and test on COCO and vice versa). However, images in both the PASCAL and COCO datasets are of general scenes while a powerful annotation tool should generalize well even on different imagery types. In the following section, we examine the generalization ability of our model on different domains, including aerial imagery (Rooftop [6]), medical images (ssTEM [8]), street scenes (Cityscapes [5]) and agriculture images (Agriculture-Vision [7]). Some qualitative examples can be found in Figures 9.

Cityscapes [5]: Following [54], [53], we first evaluate the performance of our IOG-Click on Cityscapes. Interestingly, we found that our PASCAL-trained model already performs on-par with the Cityscapes-trained methods. This suggests that our IOG generalizes well even across different domains. Moreover, the model performance can be further improved by fine-tuning using only 10% of the new dataset, where our model significantly outperforms all other baselines. In addition, our IOG-Scribble further improves the performance to 87.9% with fine-tuning when scribble is employed as the inside guidance.

This article has been accepted for publication in IEEE Transactions on Pattern Analysis and Machine Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3227116

Scenes	Methods	Train	Test	Finetune	Backbone	Number of Clicks	IoU(%)
Street Scenes	Curve-GCN [54]	Cityscapes	Cityscapes	N.A.	ResNet-50	2	76.3
	Curve-GCN [54]	Cityscapes	Cityscapes	N.A.	ResNet-50	2.4	77.6
	Curve-GCN [54]	Cityscapes	Cityscapes	N.A.	ResNet-50	3.6	80.2
	DEXTR [54]	Cityscapes	Cityscapes	N.A.	ResNet-101	4	79.4
	IOG-Click **(ours)**	PASCAL	Cityscapes	✗	ResNet-50	3	77.9
	IOG-Click **(ours)**	PASCAL	Cityscapes	✓	ResNet-50	3	82.2
	IOG-Click **(ours)**	PASCAL	Cityscapes	✓	ResNet-101	3	82.7
	IOG-Scribble **(ours)**	PASCAL	Cityscapes	✓	ResNet-101	-	87.6
	IOG-Click **(ours)**	COCO	Cityscapes	✓	ResNet-101	3	83.8
	IOG-Scribble **(ours)**	COCO	Cityscapes	✓	ResNet-101	-	87.9
	IOG-Click+ **(ours)**	COCO	Cityscapes	✓	ResNet-101	3	83.9
	IOG-Scribble+ **(ours)**	COCO	Cityscapes	✓	ResNet-101	-	**88.2**
Aerial Imagery	Curve-GCN [54]	CityScapes	Rooftop	✗	ResNet-50	2	68.3
	Curve-GCN [54]	CityScapes	Rooftop	✓	ResNet-50	2	78.2
	IOG-Click **(ours)**	PASCAL	Rooftop	✗	ResNet-50	3	90.7
	IOG-Click **(ours)**	PASCAL	Rooftop	✓	ResNet-50	3	92.8
	IOG-Click **(ours)**	PASCAL	Rooftop	✓	ResNet-101	3	93.6
	IOG-Click **(ours)**	COCO	Rooftop	✓	ResNet-101	3	94.0
	IOG-Click+ **(ours)**	COCO	Rooftop	✓	ResNet-101	3	**94.1**
Medical Images	Curve-GCN [54]	CityScapes	ssTEM	✗	ResNet-50	2	60.9
	IOG-Click **(ours)**	PASCAL	ssTEM	✗	ResNet-50	3	81.4
	IOG-Click **(ours)**	PASCAL	ssTEM	✗	ResNet-101	3	83.7
	IOG-Click **(ours)**	COCO	ssTEM	✗	ResNet-101	3	96.1
	IOG-Click+ **(ours)**	COCO	ssTEM	✗	ResNet-101	3	**97.0**
Agriculture Images	DEXTR [1]	PASCAL	Agriculture-Vision	✓	ResNet-101	4	52.9
	IOG-Click **(ours)**	PASCAL	Agriculture-Vision	✓	ResNet-101	3	66.0
	IOG-Scribble **(ours)**	PASCAL	Agriculture-Vision	✓	ResNet-101	-	80.6
	IOG-Click **(ours)**	COCO	Agriculture-Vision	✓	ResNet-101	3	66.9
	IOG-Scribble **(ours)**	COCO	Agriculture-Vision	✓	ResNet-101	-	81.6
	IOG-Click+ **(ours)**	COCO	Agriculture-Vision	✓	ResNet-101	3	67.6
	IOG-Scribble+ **(ours)**	COCO	Agriculture-Vision	✓	ResNet-101	-	**82.7**

TABLE 7
Cross-domain analysis on Cityscapes [5], Rooftop [6], ssTEM [8] and Agriculture-Vision [7]. "Finetune" indicates that the method is fine-tuned on a small set of the domain dataset (10%).

Rooftop [6]: We also evaluate our IOG-Click on Rooftop [6], an aerial imagery dataset. Similar to the observation made in Cityscapes, our IOG-Click outperforms Curve-GCN [54] by a significant margin even **without** fine-tuning. Fine-tuning on only 10% of Rooftop dataset can further introduce 2.1% improvement to the PASCAL-trained model. Similar to Cityscapes, we also observe that deeper network (ResNet-101) and COCO-trained model leads to further performance boost.

ssTEM [8]: We follow the evaluation scheme in [54], [53] by evaluating on ssTEM [8] benchmark. Note that ssTEM does not have a training split, therefore we do not perform fine-tuning on this dataset. As shown in Table 7, our IOG-Click significantly outperforms the baseline by more than 20%, demonstrating the strong generalization capability of our approach.

Agriculture-Vision [7]: Lastly, we also applied our IOG-Click on the more challenging Agriculture-Vision dataset [7]. For fair comparison, we also fine-tuned the officially released DEXTR [1] using 10% of Agriculture-Vision data. Table 7 shows that our IOG-Click surpasses DEXTR by a great margin (13.1%). Moreover, we found that IOG-Scribble further improves upon IOG-Click by 14.6%. This is possibly because the images in Agriculture-Vision are of much poorer contrast (Figure 9), therefore scribbles play an important role in outlining the rough object shape and guiding the segmentation process.

Collaborative Training Strategy: We also observe that performance on **all domains** can be substantially improved when collaborative training strategy is adopted, especially on the more challenging medical images and agriculture scenes, demonstrating its effectiveness.

4.5 Robustness to user variance

In the previous experiments, we examine the effectiveness of our IOG using the simulated inside point as inputs. Nevertheless, in practice, it is often difficult for the users to reach consensus when choosing the inside point or drawing a scribble representing the object although the users usually make consistent choices in annotating the outside points. The inconsistent inputs between training and testing will often have a negative impact on the segmentation performance, especially when applied to real annotation scenario. In this section, we discuss ways to improve the model's robustness to user variance when choosing the inside guidance.

For click-based inside guidance, we randomly perturb the position of the inside points during training. In particular, we first identify a circular region centered at the inside point extracted from the ground truth mask with a pre-defined radius (r). Then, we randomly sample a click from this region to serve as the inside point for training. To validate the effectiveness of the proposed modification, we collected the inside points annotations on all instances in PASCAL *val* set from 5 different users. As shown in Table 5, we first notice a large performance degradation when

This article has been accepted for publication in IEEE Transactions on Pattern Analysis and Machine Intelligence. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3227116

Fig. 11. **Qualitative results on PASCAL [3], PASCAL-Context [94], COCO [30] and COCO-Stuff [95]**. Each instance with the simulated inside-outside points and the corresponding segmentation masks are overlayed on the input image.

Method	Backbone	Train	IoU
(A) Crop	ResNet-50	PASCAL-1k	87.5
(B) Geo	ResNet-50	PASCAL-1k	89.5
(C) Sim	ResNet-50	PASCAL-1k	86.1
(D) Outside only	ResNet-50	PASCAL-1k	89.5
(D) Outside only	ResNet-101	PASCAL-10k	90.9
(E) 2-stage	ResNet-101	PASCAL-10k	91.1

TABLE 8
Extension to dataset with box annotations only. All the results are reported on PASCAL *val* using box annotations only.

testing the perturbation-free model with the human-provided inputs (from 93.2 to 90.8). However, the performance gaps gradually reduce when larger perturbation is applied during training. The model reaches the best trade-off when r is 30.

On the other hand, it is unclear how to perform random perturbation to the generated scribbles in scribble-based inside guidance. Instead, we collected manually-drawn scribbles by human annotators on 30% of the PASCAL-1k training set and fine-tuned the model on this subset to improve the model's robustness to real user inputs. Similar to the previous experiment, we also collected manually annotated scribbles on PASCAL *val* set from 5 different users and evaluated the performance.

From Table 5, we observed that IOG-Scribble trained with synthetic scribbles suffers from severe performance degradation when tested with human-drawn scribbles (from 96.4 to 90.7). Fine-tuning on 30% of PASCAL-1k *train* set with collected real user-drawn scribbles helps reduce the gap significantly (from 5.7% to 1.5% IoU). In overall, our IOG-Scribble significantly outperforms IOG-Click on both simulated and real user-provided inputs.

4.6 More Discussions

Extension to Datasets with Box Annotations Only: Many existing off-the-shelf datasets such as ImageNet [4] and Open Images [41], have provided bounding box annotations. Here, we explore how to quickly harvest high-quality instance segmentation masks using our IOG when only bounding box annotations are available. Specifically, we consider the annotated bounding box as an incomplete annotation for our IOG where the inside point is absent. To this end, we propose a simple two-stage solution using a small network to predict a coarse mask based on the bounding box, where the mask is used to infer the inside point candidates for IOG later. We compare this against the following baselines and the results are summarized in Table 8.

(A) **Crop:** We train a network that takes the cropped RGB image as input and predicts the segmentation.
(B) **Geo:** We train a network that takes the geometric center of the box as inside point for segmentation.
(C) **Sim:** We train our IOG with simulated clicks (Section 3.1) but using the geometric center of the given box as inside point during test time.
(D) **Outside only:** We train a single network that takes the outside points only to perform segmentation.
(E) **2-stage:** We extract the inside point from the segmentation masks produced by (D) and pass to our IOG for the final prediction.

We first observe that the setting (C) performs poorly due to train-test inconsistency. On the other hand, the methods (B) and (D) have similar performance. This is because the geometric center of the box always locates the same location after cropping, thus the network learns to ignore this input. By adopting stronger backbone and more training images, the performance of (D) can be further improved. Finally, taking the inside point from the segmentation masks predicted by (D) as inputs for our IOG produces the best result. Some qualitative results on ImageNet and Open Images are shown in Figure 10. With the annotated bounding boxes (∼0.615M) of ILSVRC-LOC, we apply our IOG to collect their pixel-level annotations, named `Pixel-ImageNet`, which are publicly available at https://github.com/shiyinzhang/Pixel-ImageNet. For more details,

please refer to our supplementary materials.

5 Conclusion

We propose a simple yet effective Inside-Outside Guidance (IOG) approach for minimizing the pixel-level labeling cost. The proposed IOG requires only three points from the users, *i.e.* an inside point near the object center and two outside points that form a box enclosing the target object. On top of that, we also proposed an IOG variant that employs coarsely-drawn scribbles as inside guidance and demonstrated significant improvement in segmentation quality while only incurring slight annotation overhead. In addition, our IOG naturally supports interactive annotation of additional points for further correction. Despite its simplicity, extensive experiments show that our model generalizes well across different datasets and domains. Lastly, we contribute an Intelligent Pixel Annotation Tool (IPAT) with our IOG, which is employed to construct a new large-scale pixel-level dataset called `Pixel-ImageNet`. We hope this work can help ease the pixel-level data collection for the future research.

6 Acknowledgement

This work was supported by the National Key R&D Program of China (No. 2018AAA0102100, No. 2021ZD0112100), and the National Natural Science Foundation of China (No. U1936212, No. 62120106009, No. 61972022), the Fundamental Research Funds for the Central Universities (No. K22RC00010).

References

[1] K.-K. Maninis, S. Caelles, J. Pont-Tuset, and L. Van Gool, "Deep extreme cut: From extreme points to object segmentation," in *CVPR*, 2018. 1, 2, 3, 4, 5, 6, 8, 9

[2] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," in *ACM ToG*, 2004. 1, 2, 6

[3] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *IJCV*, 2010. 1, 2, 5, 6, 10

[4] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *IJCV*, 2015. 1, 2, 3, 6, 7, 10

[5] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *CVPR*, 2016. 1, 2, 6, 8, 9

[6] X. Sun, C. M. Christoudias, and P. Fua, "Free-shape polygonal object localization," in *ECCV*. Springer, 2014, pp. 317–332. 1, 2, 6, 8, 9

[7] M. T. Chiu, X. Xu, Y. Wei, Z. Huang, A. Schwing, R. Brunner, H. Khachatrian, H. Karapetyan, I. Dozier, G. Rose *et al.*, "Agriculture-vision: A large aerial image database for agricultural pattern analysis," in *CVPR*, 2020. 1, 2, 6, 8, 9

[8] S. Gerhard, J. Funke, J. Martel, A. Cardona, and R. Fetter, "Segmented anisotropic sstem dataset of neural tissue," *figshare*, pp. 0–0, 2013. 1, 2, 6, 8, 9

[9] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015, pp. 3431–3440. 1, 2

[10] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *CVPR*, 2017. 1, 4

[11] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *CVPR*, 2017. 1, 2, 4, 5

[12] H. Zhao, Y. Zhang, S. Liu, J. Shi, C. Change Loy, D. Lin, and J. Jia, "Psanet: Point-wise spatial attention network for scene parsing," in *ECCV*, 2018. 1

[13] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *TPAMI*, vol. 40, no. 4, pp. 834–848, 2017. 1, 2

[14] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017. 1, 2

[15] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *ECCV*, 2018. 1, 2, 4, 5, 7

[16] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "Ccnet: Criss-cross attention for semantic segmentation," in *ICCV*, 2019, pp. 603–612. 1, 2

[17] B. Cheng, L.-C. Chen, Y. Wei, Y. Zhu, Z. Huang, J. Xiong, T. S. Huang, W.-M. Hwu, and H. Shi, "Spgnet: Semantic prediction guidance for scene parsing," in *ICCV*, 2019, pp. 5218–5228. 1, 2

[18] Z. Huang, Y. Wei, X. Wang, H. Shi, W. Liu, and T. S. Huang, "Alignseg: Feature-aligned segmentation networks," *arXiv preprint arXiv:2003.00872*, 2020. 1

[19] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *ICCV*, 2017. 1

[20] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei, "Fully convolutional instance-aware semantic segmentation," in *CVPR*, 2017, pp. 2359–2367. 1

[21] H. Chen, K. Sun, Z. Tian, C. Shen, Y. Huang, and Y. Yan, "BlendMask: Top-down meets bottom-up for instance segmentation," in *CVPR*, 2020. 1

[22] X. Wang, T. Kong, C. Shen, Y. Jiang, and L. Li, "SOLO: Segmenting objects by locations," *arXiv preprint arXiv:1912.04488*, 2019. 1

[23] X. Chen, R. Girshick, K. He, and P. Dollár, "Tensormask: A foundation for dense object segmentation," in *ICCV*, 2019, pp. 2061–2069. 1

[24] E. Xie, P. Sun, X. Song, W. Wang, X. Liu, D. Liang, C. Shen, and P. Luo, "Polarmask: Single shot instance segmentation with polar representation," 2020. 1

[25] A. Kirillov, Y. Wu, K. He, and R. Girshick, "Pointrend: Image segmentation as rendering," *arXiv preprint arXiv:1912.08193*, 2019. 1

[26] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "Yolact: Real-time instance segmentation," in *ICCV*, 2019, pp. 9157–9166. 1, 3

[27] K. Chen, J. Pang, J. Wang, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Shi, W. Ouyang, C. C. Loy, and D. Lin, "Hybrid task cascade for instance segmentation," in *CVPR*, 2019. 1

[28] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *CVPR*, 2018, pp. 8759–8768. 1, 3

[29] Z. Huang, L. Huang, Y. Gong, C. Huang, and X. Wang, "Mask scoring r-cnn," in *CVPR*, 2019, pp. 6409–6418. 1, 3

[30] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *ECCV*, 2014. 1, 2, 5, 6, 10

[31] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, "Scene parsing through ade20k dataset," in *CVPR*, 2017, pp. 633–641. 1

[32] G. Neuhold, T. Ollmann, S. Rota Bulò, and P. Kontschieder, "The mapillary vistas dataset for semantic understanding of street scenes," in *ICCV*, 2017. [Online]. Available: https://www.mapillary.com/dataset/vistas 1

[33] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *CVPR*, 2012, pp. 3354–3361. 1

[34] A. Suinesiaputra, B. R. Cowan, A. O. Al-Agamy, M. A. Elattar, N. Ayache, A. S. Fahmy, A. M. Khalifa, P. Medrano-Gracia, M.-P. Jolly, A. H. Kadish *et al.*, "A collaborative resource to build consensus for automated left ventricular segmentation of cardiac mr images," *Medical image analysis*, vol. 18, no. 1, pp. 50–62, 2014. 1

[35] X. Ning, B. Price, S. Cohen, J. Yang, and T. Huang, "Deep grabcut for object selection," in *2017 British Machine Vision Conference*, 2017. 1, 2, 3, 6

[36] J. Wu, Y. Zhao, J.-Y. Zhu, S. Luo, and Z. Tu, "Milcut: A sweeping line multiple instance learning paradigm for interactive image segmentation," in *CVPR*, 2014, pp. 256–263. 1

[37] N. Xu, B. Price, S. Cohen, J. Yang, and T. S. Huang, "Deep interactive object selection," in *CVPR*, 2016, pp. 373–381. 1, 2, 3, 7

[38] J. H. Liew, Y. Wei, W. Xiong, S.-H. Ong, and J. Feng, "Regional interactive image segmentation networks," in *ICCV*. IEEE, 2017, pp. 2746–2754. 1, 2, 6

[39] S. Mahadevan, P. Voigtlaender, and B. Leibe, "Iteratively trained interactive segmentation," *British Machine Vision Conference*, 2018. 1, 2, 6, 7

[40] Z. Li, Q. Chen, and V. Koltun, "Interactive image segmentation with latent diversity," in *CVPR*, 2018, pp. 577–585. 1

[41] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, T. Duerig *et al.*, "The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale," *arXiv preprint arXiv:1811.00982*, 2018. 2, 10

[42] S. Zhang, J. H. Liew, Y. Wei, S. Wei, and Y. Zhao, "Interactive object segmentation with inside-outside guidance," in *Proceedings of*

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 12 234–12 244. 2

[43] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in nd images," in *ICCV*, 2001. 2, 6

[44] X. Bai and G. Sapiro, "A geodesic framework for fast interactive image and video segmentation and matting," in *ICCV*, 2007. 2, 6

[45] H. Ding, S. Cohen, B. Price, and X. Jiang, "Phraseclick: Toward achieving flexible interactive segmentation by phrase and click," in *ECCV*, 2020. 2

[46] M. Andriluka, S. Pellegrini, S. Popov, and V. Ferrari, "Efficient full image interactive segmentation by leveraging within-image appearance similarity," *arXiv preprint arXiv:2007.08173*, 2020. 2

[47] Z. Lin, "Interactive image segmentation with first click attention," in *Computer Vision Pattern Recognition*, 2020. 2, 6

[48] R. Benenson, S. Popov, and V. Ferrari, "Large-scale interactive object segmentation with human annotators," 2019. 2

[49] J. H. Liew, S. Cohen, B. Price, L. Mai, S.-H. Ong, and J. Feng, "Multiseg: Semantically meaningful, scale-diverse segmentations from minimal user input," in *ICCV*, 2019, pp. 662–670. 2, 5

[50] W. D. Jang and C. S. Kim, "Interactive image segmentation via backpropagating refinement scheme," in *CVPR*, 2019. 2, 6

[51] K. Sofiiuk, I. Petrov, O. Barinova, and A. Konushin, "F-brs: Rethinking backpropagating refinement for interactive segmentation," in *CVPR*, 2020, pp. 8623–8632. 2, 6

[52] L. Castrejon, K. Kundu, R. Urtasun, and S. Fidler, "Annotating object instances with a polygon-rnn," in *CVPR*, 2017, pp. 5230–5238. 2

[53] D. Acuna, H. Ling, A. Kar, and S. Fidler, "Efficient interactive annotation of segmentation datasets with polygon-rnn++," in *CVPR*, 2018, pp. 859–868. 2, 8, 9

[54] H. Ling, J. Gao, A. Kar, W. Chen, and S. Fidler, "Fast interactive object annotation with curve-gcn," in *CVPR*, 2019, pp. 5257–5266. 2, 8, 9

[55] Z. Hengshuang, "Pyramid scene parsing network," 2017. 2

[56] L.-C. Chen, "Encoder-decoder with atrous separable convolution for semantic image segmentation," 2018. 2

[57] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," 2015. 2

[58] Vijay, Badrinarayanan, Alex, Kendall, Roberto, and Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for scene segmentation." *IEEE Transactions on Pattern Analysis Machine Intelligence*, 2017. 2

[59] W. Li, Z. Wang, B. Yin, Q. Peng, Y. Du, T. Xiao, G. Yu, H. Lu, Y. Wei, and J. Sun, "Rethinking on multi-stage networks for human pose estimation," 2019. 3

[60] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, and J. Sun, "Cascaded pyramid network for multi-person pose estimation," in *CVPR*, 2018. 3, 4, 5

[61] H. Kaiming, G. Georgia, D. Piotr, and G. Ross, "Mask r-cnn," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. PP, pp. 1–1, 2017. 3

[62] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," pp. 3150–3158, 2016. 3

[63] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "Yolact++: Better real-time instance segmentation," *arXiv preprint arXiv:1912.06218*, 2019. 3

[64] X. Wang, T. Kong, C. Shen, Y. Jiang, and L. Li, "Solo: Segmenting objects by locations," 2019. 3

[65] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Simultaneous detection and segmentation," in *European Conference on Computer Vision*, 2014. 3

[66] A. Kirillov, E. Levinkov, B. Andres, B. Savchynskyy, and C. Rother, "Instancecut: From edges to instances with multicut," in *IEEE Conference on Computer Vision Pattern Recognition*, 2017. 3

[67] G. Papandreou, L.-C. Chen, K. P. Murphy, and A. L. Yuille, "Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation," in *ICCV*, 2015, pp. 1742–1750. 3

[68] Y. Wei, J. Feng, X. Liang, M.-M. Cheng, Y. Zhao, and S. Yan, "Object region mining with adversarial erasing: A simple classification to semantic segmentation approach," in *CVPR*, 2017, pp. 1568–1576. 3

[69] Y. Wei, H. Xiao, H. Shi, Z. Jie, J. Feng, and T. S. Huang, "Revisiting dilated convolution: A simple approach for weakly-and semi-supervised semantic segmentation," in *CVPR*, 2018, pp. 7268–7277. 3

[70] Y. Wei, X. Liang, Y. Chen, X. Shen, M.-M. Cheng, J. Feng, Y. Zhao, and S. Yan, "Stc: A simple to complex framework for weakly-supervised semantic segmentation," *TPAMI*, vol. 39, no. 11, pp. 2314–2320, 2016. 3

[71] Q. Hou, P. Jiang, Y. Wei, and M.-M. Cheng, "Self-erasing network for integral object attention," in *NeurIPS*, 2018, pp. 549–559. 3

[72] P.-T. Jiang, Q. Hou, Y. Cao, M.-M. Cheng, Y. Wei, and H.-K. Xiong, "Integral object mining via online attention accumulation," in *ICCV*, 2019, pp. 2070–2079. 3

[73] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei, "What's the point: Semantic segmentation with point supervision," in *ECCV*, 2016, pp. 549–565. 3

[74] R. Qian, Y. Wei, H. Shi, J. Li, J. Liu, and T. Huang, "Weakly supervised scene parsing with point-based distance metric learning," in *AAAI*, vol. 33, 2019, pp. 8843–8850. 3

[75] J. Dai, K. He, and J. Sun, "Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation," in *ICCV*, 2015, pp. 1635–1643. 3

[76] A. Khoreva, R. Benenson, J. Hosang, M. Hein, and B. Schiele, "Simple does it: Weakly supervised instance and semantic segmentation," in *CVPR*, 2017, pp. 876–885. 3

[77] D. Lin, J. Dai, J. Jia, K. He, and J. Sun, "Scribblesup: Scribble-supervised convolutional networks for semantic segmentation," in *CVPR*, 2016, pp. 3159–3167. 3

[78] M. Tang, A. Djelouah, F. Perazzi, Y. Boykov, and C. Schroers, "Normalized cut loss for weakly-supervised cnn segmentation," in *CVPR*, 2018, pp. 1818–1827. 3

[79] B. Wang, G. Qi, S. Tang, T. Zhang, Y. Wei, L. Li, and Y. Zhang, "Boundary perception guidance: a scribble-supervised semantic segmentation approach," in *IJCAI*, 2019, pp. 3663–3669. 3

[80] E. Agustsson, J. R. R. Uijlings, and V. Ferrari, "Interactive full image segmentation," 2018. 3

[81] C. Rupprecht, I. Laina, N. Navab, G. D. Hager, and F. Tombari, "Guide me: Interacting with deep networks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018. 3

[82] H. Ding, "Phraseclick: Toward achieving flexible interactive segmentation by phrase and click," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 3

[83] T. Shen, "Interactive annotation of 3d object geometry using 2d scribbles," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 3

[84] H. Su, J. Deng, and L. Fei-Fei, "Crowdsourcing annotations for visual object detection," in *AAAI*, 2012. 3

[85] O. Russakovsky, L.-J. Li, and L. Fei-Fei, "Best of both worlds: human-machine collaboration for object annotation," in *CVPR*, 2015, pp. 2121–2131. 3

[86] S. Dutt Jain and K. Grauman, "Predicting sufficient annotation strength for interactive foreground segmentation," in *ICCV*, 2013, pp. 1313–1320. 3

[87] D. P. Papadopoulos, J. R. Uijlings, F. Keller, and V. Ferrari, "Extreme clicking for efficient object annotation," in *ICCV*, 2017, pp. 4930–4939. 3

[88] S. Caelles, A. Montes, K.-K. Maninis, Y. Chen, L. Van Gool, F. Perazzi, and J. Pont-Tuset, "The 2018 davis challenge on video object segmentation," *arXiv:1803.00557*, 2018. 4, 6

[89] B. Hariharan, P. Arbeláez, L. Bourdev, S. Maji, and J. Malik, "Semantic contours from inverse detectors," in *ICCV*, 2011. 5

[90] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016. 4

[91] Z. Lin, Z. Zhang, L.-Z. Chen, M.-M. Cheng, and S.-P. Lu, "Interactive image segmentation with first click attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 13 339–13 348. 5

[92] L. Grady, "Random walks for image segmentation," *TPAMI*, 2006. 6

[93] S. Majumder and A. Yao, "Content-aware multi-level guidance for interactive instance segmentation," in *CVPR*, 2019. 6

[94] R. Mottaghi, X. Chen, X. Liu, N. Cho, S. Lee, S. Fidler, R. Urtasun, and A. L. Yuille, "The role of context for object detection and semantic segmentation in the wild," pp. 891–898, 2014. 6, 8, 10

[95] H. Caesar, J. Uijlings, and V. Ferrari, "Coco-stuff: Thing and stuff classes in context," pp. 1209–1218, 2018. 6, 8, 10

[96] B. Cheng, R. Girshick, P. Dollár, A. C. Berg, and A. Kirillov, "Boundary IoU: Improving object-centric image segmentation evaluation," in *CVPR*, 2021. 7

Shiyin Zhang is currently a Ph.D. student with Beijing Jiaotong University, Beijing, China.

Shikui Wei is currently a full Professor with the Institute of Information Science, Beijing Jiaotong University, Beijing, China.

Jun Hao Liew is a research fellow with National University of Singapore, Singapore.

Kunyang Han is currently a Ph.D. student with Beijing Jiaotong University, Beijing, China.

Yao Zhao is now the Director of the Institute of Information Science, Beijing Jiaotong University (BJTU), Beijing, China.

Yunchao Wei is currently a full Professor with the Institute of Information Science, Beijing Jiaotong University, Beijing, China.