Interactive Object Segmentation with Inside-Outside Guidance

Shiyin Zhang\textsuperscript{1,2}, Jun Hao Liew\textsuperscript{3}, Yunchao Wei\textsuperscript{4}, Shikui Wei\textsuperscript{1,2}, Yao Zhao\textsuperscript{1,2}

\textsuperscript{1}Institute of Information Science, Beijing Jiaotong University
\textsuperscript{2}Beijing Key Laboratory of Advanced Information Science and Network Technology
\textsuperscript{3}National University of Singapore \textsuperscript{1}ReLER, University of Technology Sydney

https://github.com/shiyinzhang/Inside-Outside-Guidance

Abstract

This paper explores how to harvest precise object segmentation masks while minimizing the human interaction cost. To achieve this, we propose an Inside-Outside Guidance (IOG) approach in this work. 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 a tight bounding box that encloses the target object. This 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 to remove distractions from other objects or background; 2) the inside point can help to 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 in labeling some extreme samples; 4) our approach naturally supports additional clicks annotations for further correction. Despite its simplicity, our IOG not only achieves state-of-the-art performance on several popular benchmarks, but also demonstrates strong generalization capability across different domains such as street scenes, aerial imagery and medical images, without fine-tuning. In addition, we also propose a simple two-stage solution that enables our IOG to produce high quality instance segmentation masks from existing datasets with off-the-shelf bounding boxes such as ImageNet and Open Images, demonstrating the superiority of our IOG as an annotation tool.

1. Introduction

Over the past few years, we have witnessed a revolutionary advancement in semantic\cite{44,40,68,69,10,11,12,30,15,31} and instance segmentation\cite{25,36,8,60,13,65,34,4,9,43,29} for different domains, such as general scenes\cite{20,41,70}, autonomous driving\cite{17,48,21}, aerial imagery\cite{57,16}, medical diagnosis\cite{22,56}, 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, allows the human annotators to quickly extract the object-of-interest by providing some user inputs such as bounding boxes\cite{66,52,64} or clicks\cite{67,38,45,37}, appears to be an attractive and efficient way to reduce the annotation effort.

Recently, Maninis \textit{et al.} \cite{46} explored the use to extreme points of an object (left-most, right-most, top, bottom pixels) 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 1(a)) or when there are unrelated objects or background lying inside the target object (dog in Figure 1(a)).

To tackle the aforementioned issues as well as to pro-
motivate 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 1(b) shows two examples of our proposed labeling scheme. Similar to [46], our IOG relaxes the generated bounding box by several 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 1(c).

Our IOG strategy not only improves the annotation speed by reducing the confusion raised by [46], but also naturally supports annotation of additional points at the erroneous regions for further refinement. We perform extensive experiments on PASCAL [20], GrabCut [52] and COCO [41] 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 further improves to 94.4% when the 4th click is added for interactive 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 [17], aerial imagery [57, 16] and medical images [22] without 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 [53] and Open Images [35] without any human interaction. Finally, we release Pixels-ImageNet, a dataset with 0.615M instance masks of ImageNet [53] collected using our IOG. We hope this work can significantly benefit the future researchers in collecting large-scale pixel-level annotations.

2. Related Work

Interactive Segmentation: Most interactive segmentation methods target at relieving human effort when labeling pixel-level annotations with the help of bounding box [66, 64, 52], clicks [67, 38, 37, 45] or contours [7, 1, 42] as guidance. GrabCut [52] employs bounding boxes to guide the segmentation process, which is one of the pioneering works for interactive segmentation task. Similarly, Xu et al. [66] also take bounding boxes as inputs to train a deep convolutional neural network (CNN) for interactive segmentation. On the other hand, iFCN [67] is 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. Recently, Maninis et al. [46] propose DEXTR that leverages only 4 extreme points for segmentation and achieve the new state-of-the-art. Comparing with DEXTR, we further advance the interactive annotation process by reducing the number of clicks required from 4 to 3. Our IOG not only tackles the issues raised by DEXTR, but also achieves much better segmentation performance. Beyond the above mentioned approaches, some other works [7, 1, 42] alternatively propose to conduct interactive segmentation by directly predicting a polygon or spline around the target object.

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 [50, 61, 63, 62, 27, 32], points [3, 51], bounding boxes [18, 33] and scribbles [39, 58, 59] 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.

Semantic Segmentation: Fully convolutional networks (FCN) [44, 40, 68, 69, 10, 11, 12, 30, 15] has greatly advanced the semantic image segmentation. The success of CNN-based interactive segmentation has benefited significantly from the development of FCNs. Specifically, FCN [44], DeepLab series [10, 11, 12] and PSP [68] have been directly applied to tackle the interactive segmentation [67, 38, 45, 46]. 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 [14] as the backbone of our IOG method. We experimentally validate our choice can further boost the accuracy of interactive segmentation by a large margin.

3. Method

3.1. Inside-Outside Guidance

Our Inside-Outside Guidance clicking paradigm 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.

\[\text{https://github.com/shiyinzhang/Pixel-ImageNet}\]
(Figure 2). 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 gives an “inside” guidance (indicating the foreground regions), thus giving the name Inside-Outside Guidance (IOG).

**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 ([55] reported 25.5s for drawing one box on ImageNet [53]). This is due to the difficulty of clicking on the corners of an imaginary box where these corners are often not on the object [49]. Thus, several adjustments are usually required 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 inside guidance is formulated as an interior click located around the object center. To simulate clicks annotated by human annotators, we propose to sample the inside point is very fast, taking about 1.5s in our user study.

**Clicks representations:** We use the same clicks representation as DEXTR[46] by centering a 2D Gaussian around each click, creating two separate heatmaps for foreground and background clicks. The resulting heatmaps are concatenated with the RGB input image to form a 5-channel input for the network. Similar to [46], 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)).

Compared with existing clicks-based [67, 46] and box-based [66] 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 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.

### 3.2. Segmentation Network

Here, we discuss the architectural design of our segmentation network. We employ a ResNet-50 [26]-based DeepLabv3 [12] 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 3. 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.

---

3 Some papers reported much faster timings (e.g. [54] reported 10.21s while [19] reported 7.0s). However, [49] argue that the annotated boxes are of low quality (not tight around the object).
In this work, we propose to adopt a coarse-to-fine design for addressing the aforementioned issue (Figure 4). In particular, we employ a cascaded structure similar to [14] 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 [40] that progressively fuses the semantic information from the deeper layers with low-level details from the earlier layers via lateral connections. Different from [14], we also append a pyramid scene parsing (PSP) module [68] at the deepest layer 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 upampling and concatenation operations. Different from [14], we also append 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.

**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 [14]. During inference, the segmentation mask is 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 backbone typically requires only 20 ms on a Nvidia GeForce GTX 1080 GPU. It is thus well-suited for practical interactive image segmentation application.

Although our IOG approach requires only three clicks to perform segmentation, our framework naturally supports interactive adding of new foreground and background clicks for further refinement if the user is not satisfied with the current segmentation output. To achieve this, we append a lightweight branch before the PSP module to accept the two-channel Gaussian heatmaps encoding all the foreground and background clicks (Figure 4(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, an initial segmentation mask is first obtained given only three clicks. A new click is then added to the center of the largest erroneous region and second forward pass is conducted. Results presented in Section 4.3 shows that such iterative training strategy is necessary.

### 4. Experiments

We conduct extensive experiments on 11 publicly available benchmarks, including PASCAL [20], Grab-Cut [52], COCO [41], ImageNet [53], Open Images [53], Cityscapes [17], Rooftop [57], Agriculture-Vision [16], ssTEM [22], Pascal-Context [47], and COCO-Stuff [6], 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 [46], we employ PASCAL as the main benchmark to verify the importance of each component proposed in our IOG.

#### 4.1. Implementation Details

**Simulated inside-outside points:** We use the ground truth masks to generate the inside-outside points for training.
Table 1. Comparison with the state-of-the-art methods on PASCAL 2012 Segmentation images with different categories as PASCAL). 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 2, our IOG makes consistent improvements over DEXTR on various settings despite using only 3 clicks. Some qualitative results are shown in Figure 5.

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 vs. ResNet-101), different number of training images (PASCAL-1K vs. PASCAL-10K), inclusion of PSP module for global contextual information (Context), FineNet and the use of side losses for training. As shown in Table 3, “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 further correction. Specifically, we progressively

Table 3. Ablation Study. Justification of each component in the segmentation network on the PASCAL VOC 2012 val set.
add a new click to the center of the largest erroneous regions similar to [67, 45]. 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 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 vs. 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\(^4\) and re-train DEXTR using DeepLabv3+ [12] as the fully convolutional architecture on PASCAL-1K. All the models are pre-trained only on ImageNet [53]. 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.

\(^4\)https://github.com/scaelles/DEXTR-PyTorch
Figure 7. Interactive refinement. Our proposed IOG supports interactive adding of new clicks for further refinement.

4.4. Cross-Domain Evaluation

In Section 4.2, 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 across different domains, including non-PASCAL-style object categories and stuff categories.

Object categories: Following [42, 1], we evaluated the performance of our IOG on 3 imagery types, including street scenes (Cityscapes [17]), aerial imagery (Rooftop [57]) and medical images (ssTEM [22]). The results are summarized in Table 4, 5 and 6. We can see that our model outperforms the baselines by a large margin on Rooftop and ssTEM datasets even without fine-tuning. More interestingly, as shown in Table 4, our PASCAL-trained model already performs on-par with the Cityscapes-trained methods when evaluating on the Cityscapes dataset. 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 the other baselines. In addition, we also applied our IOG on the more challenging Agriculture-Vision dataset [16] and still achieved satisfactory performance when using small amounts of data for fine-tuning. Fig 8 provides some qualitative examples.

Stuff categories: In Fig 5, we show some qualitative results of our IOG fine-tuned on PASCAL-Context [47] and COCO-Stuff [6] to verify the performance of our IOG when segmenting “stuff” categories. The results show that our IOG generalizes well to background classes too.

4.5. More Discussions

Robustness to user variance when choosing the inside points: In the previous experiments, we examine the effectiveness of our IOG using the simulated inside-outside points as inputs. Nevertheless, in practice, it is often difficult for the users to reach consensus when choosing the

<table>
<thead>
<tr>
<th>Methods</th>
<th>Train</th>
<th>Finetune</th>
<th>Backbone</th>
<th>#Clicks</th>
<th>IoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Curve-GCN [42]</td>
<td>Cityscapes</td>
<td>×</td>
<td>ResNet-50</td>
<td>2</td>
<td>68.3</td>
</tr>
<tr>
<td>Curve-GCN [42]</td>
<td>Cityscapes</td>
<td>×</td>
<td>ResNet-50</td>
<td>2</td>
<td>78.2</td>
</tr>
<tr>
<td>IOG (ours)</td>
<td>PASCAL</td>
<td>✓</td>
<td>ResNet-50</td>
<td>3</td>
<td>90.7</td>
</tr>
<tr>
<td>IOG (ours)</td>
<td>PASCAL</td>
<td>✓</td>
<td>ResNet-50</td>
<td>3</td>
<td>92.8</td>
</tr>
<tr>
<td>IOG (ours)</td>
<td>PASCAL</td>
<td>✓</td>
<td>ResNet-101</td>
<td>3</td>
<td>93.6</td>
</tr>
<tr>
<td>IOG (ours)</td>
<td>COCO</td>
<td>✓</td>
<td>ResNet-101</td>
<td>3</td>
<td>94.8</td>
</tr>
</tbody>
</table>

Table 4. Cross domain analysis on Cityscapes [17]. “Fine-tune” indicates that the method is fine-tuned on a small set of the Cityscapes dataset (10%).

<table>
<thead>
<tr>
<th>Methods</th>
<th>Train</th>
<th>Finetune</th>
<th>Backbone</th>
<th>#Clicks</th>
<th>IoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Curve-GCN [42]</td>
<td>CityScapes</td>
<td>×</td>
<td>ResNet-50</td>
<td>2</td>
<td>60.9</td>
</tr>
<tr>
<td>IOG (ours)</td>
<td>PASCAL</td>
<td>✓</td>
<td>ResNet-50</td>
<td>3</td>
<td>81.4</td>
</tr>
<tr>
<td>IOG (ours)</td>
<td>PASCAL</td>
<td>✓</td>
<td>ResNet-101</td>
<td>3</td>
<td>83.7</td>
</tr>
</tbody>
</table>

Table 5. Cross domain analysis on Rooftop [57]. Even without fine-tuning, our method already outperforms Curve-GCN with fine-tuning, showing the strong generalization of our approach.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Train</th>
<th>Finetune</th>
<th>Backbone</th>
<th>#Clicks</th>
<th>IoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Curve-GCN [42]</td>
<td>Cityscapes</td>
<td>×</td>
<td>ResNet-50</td>
<td>2</td>
<td>76.1</td>
</tr>
<tr>
<td>Curve-GCN [42]</td>
<td>Cityscapes</td>
<td>×</td>
<td>ResNet-50</td>
<td>2.4</td>
<td>77.6</td>
</tr>
<tr>
<td>Curve-GCN [42]</td>
<td>Cityscapes</td>
<td>×</td>
<td>ResNet-50</td>
<td>3.6</td>
<td>80.2</td>
</tr>
<tr>
<td>DEXTR [42]</td>
<td>Cityscapes</td>
<td>N.A.</td>
<td>ResNet-101</td>
<td>4</td>
<td>79.4</td>
</tr>
<tr>
<td>IOG (ours)</td>
<td>PASCAL</td>
<td>✓</td>
<td>ResNet-50</td>
<td>3</td>
<td>77.9</td>
</tr>
<tr>
<td>IOG (ours)</td>
<td>PASCAL</td>
<td>✓</td>
<td>ResNet-50</td>
<td>3</td>
<td>82.2</td>
</tr>
<tr>
<td>IOG (ours)</td>
<td>COCO</td>
<td>✓</td>
<td>ResNet-101</td>
<td>3</td>
<td>83.8</td>
</tr>
</tbody>
</table>

Table 6. Cross domain analysis on ssTEM [22]. Note that ssTEM does not have a training split, therefore we do not perform fine-tuning on this dataset.

Figure 8. Cross-domain performance. Qualitative results of our IOG on Cityscapes, Agricultural-Vision, Rooftop, and ssTEM.
the inside points annotations on all instances in PASCAL
ion to serve as the inside point for training. To validate
gin for the proposed modification, we collected inputs between training and testing will often have a neg-
outside points. The inconsistent choices in annotating the outside points. The inconsistent
To alleviate the negative effect caused by user variance in selecting the inside points, 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 7, we first notice a large performance degradation when 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.

Extension to datasets with box annotations only: Many existing off-the-shelf datasets such as ImageNet and Open Images, have provided bounding box annotations. Here, we explore how to quickly harvest high-quality instance segmentation masks using our IOG for 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.

Table 7. Manual vs. simulated inside points. r denotes the radius of perturbation applied during training. All the models take ResNet-101 as backbone and are trained on PASCAL-10k.

<table>
<thead>
<tr>
<th>Method</th>
<th>Backbone</th>
<th>Train</th>
<th>IoU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sim</td>
<td>ResNet-50</td>
<td>PASCAL-1k</td>
<td>89.5</td>
</tr>
<tr>
<td>Geo</td>
<td>ResNet-50</td>
<td>PASCAL-1k</td>
<td>89.5</td>
</tr>
<tr>
<td>Sim</td>
<td>ResNet-50</td>
<td>PASCAL-1k</td>
<td>89.5</td>
</tr>
<tr>
<td>Outside only</td>
<td>ResNet-101</td>
<td>PASCAL-10k</td>
<td>90.9</td>
</tr>
<tr>
<td>Outside only</td>
<td>ResNet-101</td>
<td>PASCAL-10k</td>
<td>91.1</td>
</tr>
</tbody>
</table>

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

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 9. 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.

5. Conclusion

We propose a simple yet effective Inside-Outside Guidance (IOG) approach for minimizing the 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. In addition, our method 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, demonstrating its superiority as an annotation tool.

Acknowledgments This work is in part supported by National Key Research and Development of China (2017YFC1703503), National Natural Science Foundation of China (No.61532005, 61972022), Grant U1936212, and ARC DECRA DE190101315.
References


