DEPTH-AWARE OBJECT INSTANCE SEGMENTATION

Linwei Ye[⋆] *Zhi Liu*[†] *Yang Wang*[⋆]

[⋆]Department of Computer Science, University of Manitoba, Winnipeg, MB, Canada
[†]School of Communication and Information Engineering, Shanghai University, Shanghai, China
{yel3, ywang}@cs.umanitoba.ca[⋆] liuzhi@staff.shu.edu.cn[†]

ABSTRACT

We consider the problem of object instance segmentation. The goal is to label each pixel in an image according to its object class as well as its object instance. The proposed approach consists of three steps including object instance detection, category-specific instance segmentation and depth-aware ordering. The novelty of the proposed approach is that it uses the depth information to resolve the ambiguity of pixel labels when two object instances are overlapping. Experimental results on the PASCAL VOC 2012 benchmark demonstrate the competitive performance of the proposed approach compared with other state-of-the-art methods.

Index Terms— instance segmentation, depth, occlusion reasoning

1. INTRODUCTION

We consider the problem of object instance segmentation. Given an image, the goal is to label each pixel according to its object class as well as its object instance. Instance segmentation is closely related to two important tasks in computer vision, namely *semantic segmentation* and *object detection*. The goal of semantic segmentation is to label each pixel according to its object class. However, semantic segmentation does not differentiate between two different object instances of the same class. For example, if there are two persons in an image, semantic segmentation will assign the same label to pixels belonging to either of these two persons. The goal of object detection is to predict the bounding box and the object class of each object instance in the image. However, object detection does not provide per-pixel labeling of the object instance. Compared with semantic segmentation and object detection, object instance segmentation is strictly more challenging, since it aims to identify object instance as well as provide per-pixel labeling of each object instance.

Object instance segmentation is a relatively new area in computer vision. Depending on how the instance segmentation results are represented, existing work on object instance segmentation can be classified into two categories: detection-level instance segmentation (e.g. [1, 2, 3]) and image-level instance segmentation (e.g. [4, 5]). Detection-level instance segmentation methods usually involve two stages, namely object detection and semantic segmentation. These methods consider all generated instances over the image and allow overlapping among different instances. In other words, a pixel in the image can belong to the segmentation masks of two different object instances. In contrast, image-level instance segmentation aims to assign each pixel to at most one object instance in the image. Since image-level instance segmentation needs to resolve the possible ambiguity of the pixel labels and assign each pixel to a unique object instance, it is more challenging than detection-level instance segmentation.

In this paper, we focus on image-level object instance segmentation. In other words, our goal is to assign each pixel in the image to at most one object instance. We propose a depth-aware object instance segmentation approach. The proposed approach consists of three steps: object instance detection, category-specific instance segmentation and depth-aware ordering. Our main contribution of our work is to introduce a novel depth-based occlusion reasoning that can resolve the ambiguity of pixel labels when two object instances are overlapping.

Related Work: Object instance segmentation is related to object detection and semantic segmentation. In recent years, convolutional neural networks (CNN) have been shown to be effective in solving both tasks. The state-of-the-art object detectors (e.g. [6, 7, 8]) work by generating object proposals [9] and then classifying each object proposal using CNN. Most recent semantic segmentation methods (e.g. [10, 11, 12]) use CNN with deconvolution or atrous convolution.

Recently, some efforts have been made for object instance segmentation. The detection-level instance segmentation methods generally focus on simultaneous detection and segmentation [1]. [2] proposes an intuitive energy minimization framework for category specific reasoning and shape prediction. A multi-task cascade network is designed by [3], which has a similar network as [7] for object proposals and adds an additional segmentation network for instance segmentation masks. For image-level instance segmentation, [13] introduces an associative embedding method using a tag label to identify the instance to obtain the segmentation result at one time. Affinity learning and boundary-based method are proposed in [5] to parse and separate instances directly from

semantic segmentation result.

2. OUR APPROACH

Figure 1 shows an overview of our approach. Given an input image, we first generate a set of candidate object instances using off-the-shelf object detectors. Each instance is represented as a bounding box. We also apply a depth estimation algorithm to estimate the depth of each pixel in the image. The result of the depth estimation is used to establish the depth ordering of the candidate instances according to their distances to the camera. We then apply a category-specific segmentation network to perform a pixel-wise labeling of the pixels in each candidate bounding box. Finally, the segmentation masks generated from candidate instances are placed within an output image in the order of their depth. The depth information can help resolve the ambiguity due to occlusions. For example, if two candidate object instances are overlapping, it is possible for a pixel to be claimed as foreground by both object instances. In this case, we can resolve the ambiguity using the depth information and assign the pixel to the object instance closer to the camera.

2.1. Object Instance Detection

Given an input image, the first step of our approach is to generate candidate object instances in the image. Each candidate object instance is represented as a bounding box. We can use any off-the-shelf object detectors for generating the candidate object instances. In this paper, we choose to use Faster R-CNN [8], since it is a state-of-the-art object detector and has been proved to be both effective and efficient. Faster R-CNN consists of two sub-networks, namely Region Proposal Network (RPN) for generating object proposals and Fast R-CNN [7] for detection. These two networks share features in their common convolutional layers resulting in the faster speed for object detection.

We use I_n^i to denote the n-th detection of the object category i, where $n = 1, ..., N_i$ and i denotes a specific object class (e.g., i is one of twenty object categories in PASCAL VOC dataset [14]), and N_i denotes the number of detected instances of the i-th object class. Figure 1(a) shows examples of four object instances (two instances of "people" and two instances of "horse") detected by Faster R-CNN. Due to occlusions, these instances are overlapping.

2.2. Category-Specific Instance Segmentation

The object detector in Sec. 2.1 provides a collection of detected object instances. For each detected object instance (i.e. bounding box), our next step is to produce a pixel-wise segmentation mask, where each pixel is labeled according to whether it belongs to the object or the background. Note that it is possible to produce this segmentation mask by applying a generic semantic segmentation method on each detected object instance. However, since we already know the object category of each detected object instance and only need to label each pixel in the bounding box as whether it belongs to the object category or not, our problem is easier than generic multi-class semantic segmentation. So instead of using a generic semantic segmentation model, we choose to learn a category-specific segmentation network for each object category. For each object instance, we apply the corresponding category-specific segmentation network to produce the segmentation mask.

We build our category-specific segmentation network based on DeepLab [12] to produce the segmentation mask for each detected object instance. DeepLab is a state-of-the-art method for semantic segmentation. The original DeepLab is designed for multi-class semantic segmentation, where the goal is to label each pixel as one of the several classes. But in our case, we would like to have a category-specific segmentation network that can produce a binary segmentation mask for a specific object category. We tailor category-specific segmentation network to each category i by fine-tuning the DeepLab network. Each category instance is cropped and re-scaled to the same size of 321x321 from the original image and used for the learning. At test time, each instance I_n^i obtained from Sec. 2.1 is passed to the corresponding category-specific segmentation network to generate the segmentation mask M_n^i for this instance. Some examples of generated instance segmentation masks are shown in Fig. 1(c).

Since the segmentation mask for each object instance is generated independently, some pixels can be claimed by two different object instances. This often happens when the two object instances are occluding each other. Figure 1(d) shows a simple fusion of all instance segmentation masks, where pixels claimed by more than one object instance are colored as white. We can see that a large block of regions are claimed by three potential instances, i.e. two horses and the person on the right. Since the goal of this paper is to do image-level instance segmentation, we would like to resolve this ambiguity caused by occlusion and assign each pixel to at most one instance.

2.3. Depth-Aware Ordering

In order to resolve the ambiguity caused by occlusion and assign each pixel to at most one object instance, we introduce depth-aware ordering based on relative depth to address this problem.

Consider two overlapping segmentation masks M_n^i and M_m^j, we would like to resolve the ambiguity in assigning the overlapping region. We consider two cases separately. The first case is that there is partial overlap between these two segmentation masks. We propose to assign the overlapping region to a specific instance segmentation mask in the image based on relative depth of these two object instances. In order to estimate the relative depth of object instances, we adopt

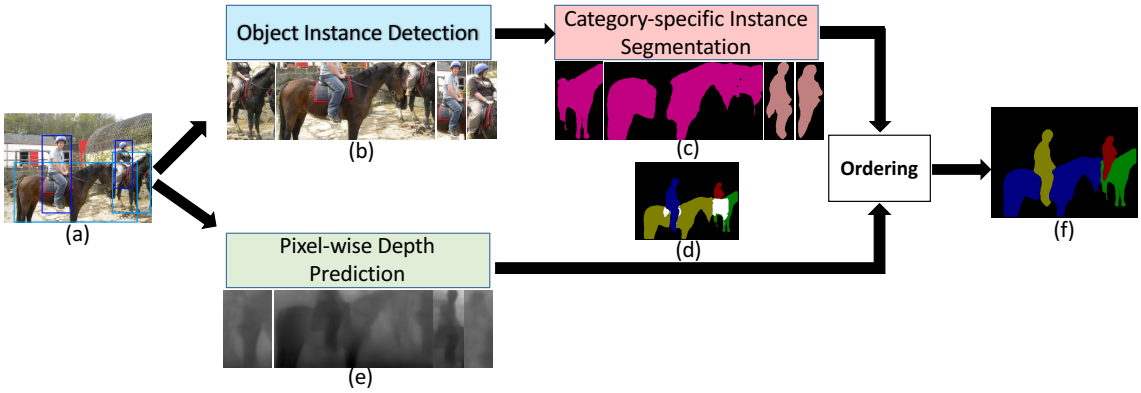

Fig. 1. Overview of the proposed instance segmentation approach. Given an input image (a), we produces a number of candidate object instances (b). Then these instances are passed to the category-specific segmentation network to generate their corresponding category-specific instance segmentation masks (c). (d) shows some overlapping regions colored as white between different instances. To resolve the ambiguity of the pixels in the overlapping regions, depth estimation is used to predict pixel-wise depth value (e) and followed by an depth-aware ordering strategy to generate the final instance segmentation result (f).

the hourglass network [15], which outputs pixel-wise relative depth value $D(p)$ for each pixel p in the image. Figure 1(e) shows an example of the estimated depth map where darker pixels correspond to regions closer to the camera. Then the relative depth for each instance segmentation mask is defined as $D(M_n^i) = \frac{\sum_{p \in M_n^i} D(p)}{num(p \in M_n^i)}$, where $num(\cdot)$ is the total number of pixels in the instance segmentation mask. For the occluded mask region $M_{n,m}$ between M_n^i and M_m^j, $M_{n,m}$ will be assigned to the mask that has a smaller depth value as follows:

$$M_{n,m} \in \begin{cases} M_n^i & \text{if } |D(M_n^i) - D(M_{n,m})| \\ & < |D(M_m^j) - D(M_{n,m})| \\ M_m^j & \text{otherwise} \end{cases} \quad (1)$$

The second case is that M_n^i (M_m^j) is completely covered by M_m^j (M_n^i). We adopt near-to-far strategy to place these two instance segmentation masks in the order of their distance to the camera. After all instance segmentation masks are processed, the final instance segmentation result is shown in Fig. 1(f) where the occluded regions shown in Fig. 1(d) are successfully assigned to people and horses respectively and the four individual instances are defined clearly with their boundaries.

3. EXPERIMENTAL RESULTS

Following [1, 5, 13], we evaluate our object instance segmentation approach on the PASCAL VOC 2012 validation dataset [14]. We use the segmentation dataset in [16] to train the category-specific segmentation network used in our approach. It contains 20 object categories and 10582 images. We compare our approach with several existing state-of-the-art instance segmentation methods, including both image-level instance segmentation methods [4, 5] and detection-level instance segmentation methods [1, 2, 3]. We also compare with two baselines based on the semantic segmentation results obtained from DeepLab [12]. The first baseline (DeepLab + connected components) directly extracts connected components from the semantic segmentation mask as individual instances. The second baseline (DeepLab + Faster R-CNN) uses the Faster R-CNN object detection results to obtain instances from the semantic segmentation masks. The semantic segmentation mask with the same category label as object detection result in the bounding box is labeled as an instance.

Table 1 shows the comparison result using mAP^r [1] under IoU threshold at 0.5. Our approach consistently outperforms all image-level instance segmentation methods [4, 5]. This clearly demonstrates the effectiveness of our proposed approach and the contribution of category-specific segmentation network and depth-aware ordering. In addition, the performance of our approach is either better than or comparable to other detection-level instance segmentation methods [1, 2, 3]. However, it should be noted that detection-level instance segmentation methods are not directly comparable to image-level instance segmentation methods under this metric, since detection-level instance segmentation ignores the occlusion between different instances and can assign the same pixel in the image to more than one instance. Compared with detection-level instance segmentation, image-level instance segmentation is a more challenging problem since we have to assign a pixel to at most one object instance.

Jin et al. [5] propose a metric called $AR@10$ to make the detection-level and image-level instance segmentation comparable. This metric measures the average recall between IoU overlap threshold from 0.5 to 1 and allows at most 10 instances used for instance segmentation evaluation over an

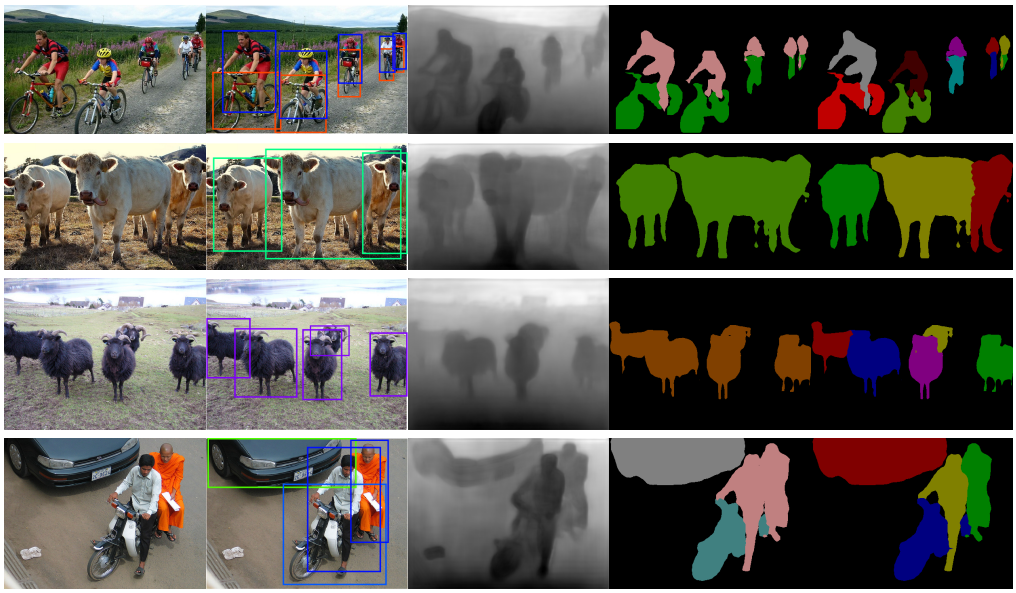

Fig. 2. Some qualitative examples of our approach on the PASCAL VOC 2012 validation dataset. For each input image, we show the results of object detection, depth estimation, semantic segmentation, and object instance segmentation.

Method	$mAP^r(\%)$
Detection-level instance segmentation	
SDS [1]	43.9
OH [2]	46.3
MNC [3]	63.5
DeepLab [12] + Faster R-CNN [8]	46.7
Image-level instance segmentation	
AE [4]	35.1
ODF [5]	49.9
DeepLab [12] + connected components	45.3
Ours	53.9

Table 1. Comparison of instance segmentation results on the PASCAL VOC 2012 validation dataset in terms of mAP^r. Note that detection-level instance segmentation and image-level instance segmentation are not directly comparable under this metric, since the former allows a pixel to be assigned to more than one instances.

image. As shown in Table 2, our proposed approach performs either better than or comparable to other state-of-the-art object instance segmentation methods under this metric.

Figure 2 shows some qualitative examples. We can see that the proposed approach is capable of distinguishing multiple overlapping instances of different categories (e.g. people and bicycle/motorbike in the 1st and 4th images). It can also identify partially occluded instances of same categories (e.g. cow and sheep in the 2nd and 3rd images).

Method	$AR@10(\%)$
Detection-level instance segmentation	
SDS [1]	7.0
MNC [3]	33.4
Image-level instance segmentation	
ODF [5]	38.8
Ours	38.7

Table 2. Comparison of instance segmentation results on the PASCAL VOC 2012 validation dataset in terms of $AR@10$ defined in [5].

4. CONCLUSION

In this work, we have proposed a novel depth-aware object instance segmentation approach. Our approach consists of three steps: object instance detection, category-specific instance segmentation and depth-aware ordering. The novelty of our approach is that it uses the estimated depth information to resolve the ambiguity of pixel labels when object instances are overlapping. The experimental results have demonstrated that the effectiveness of the proposed approach.

Acknowledgement: LY and YW are supported by NSERC. ZL is supported by the National Natural Science Foundation of China under Grant No. 61471230, and by the Program for Professor of Special Appointment (Eastern Scholar) at Shanghai Institutions of Higher Learning. LY is also supported by the University of Manitoba GETS funding program. We also thank NVIDIA for the GPU donations.

5. REFERENCES

[1] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik, "Simultaneous detection and segmentation," in *European Conference on Computer Vision*, 2014, pp. 297–312.

[2] Yi-Ting Chen, Xiaokai Liu, and Ming-Hsuan Yang, "Multi-instance object segmentation with occlusion handling," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3470–3478.

[3] Jifeng Dai, Kaiming He, and Jian Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3150–3158.

[4] Alejandro Newell, Kaiyu Yang, and Jia Deng, "Stacked hourglass networks for human pose estimation," in *European Conference on Computer Vision*, 2016, pp. 483–499.

[5] Long Jin, Zeyu Chen, and Zhuowen Tu, "Object detection free instance segmentation with labeling transformations," *arXiv preprint arXiv:1611.08991*, 2016.

[6] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580–587.

[7] Ross Girshick, "Fast r-cnn," in *IEEE International Conference on Computer Vision*, 2015, pp. 1440–1448.

[8] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 91–99.

[9] Jasper R.R. Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, 2013.

[10] Jonathan Long, Evan Shelhamer, and Trevor Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[11] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han, "Learning deconvolution network for semantic segmentation," in *IEEE International Conference on Computer Vision*, 2015, pp. 1520–1528.

[12] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *arXiv preprint arXiv:1606.00915*, 2016.

[13] Alejandro Newell and Jia Deng, "Associative embedding: End-to-end learning for joint detection and grouping," *arXiv preprint arXiv:1611.05424*, 2016.

[14] Mark Everingham, Luc Van Gool, Christopher K.I. Williams, John Winn, and Andrew Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.

[15] Weifeng Chen, Zhao Fu, Dawei Yang, and Jia Deng, "Single-image depth perception in the wild," in *Advances in Neural Information Processing Systems*, 2016, pp. 730–738.

[16] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik, "Semantic contours from inverse detectors," in *International Conference on Computer Vision*, 2011, pp. 991–998.