Amodal Instance Segmentation for Occlusion Handling with Balanced Overlapping Trilayer

Hyunjin Kim Jangho Park Jungmin Han Jubi Hwang

Abstract

*Segmentation is one of the essential parts of scene understanding, and most recent research in instance segmentation has been conducted in the visible area. However, scenes mostly feature overlapped multiple objects in the actual world situation, and previous works still have the overlapping problem because of its simple computational segmentation method with ROI box extraction or extra post-processing. We present an advanced amodal instance segmentation method called **T**rilayer **C**onvolutional **Net**work (**TCNet**), which was based on BCNet and included an additional GCN layer for the occludee for an effective mask prediction in two-stage instance segmentation. For amodal segmentation, TCNet extracts the ROI feature and gives the ROI feature as an input for occluder and occludee prediction. After that, it combines all features and gives it to the last layer to get the result. We conducted experiments on MS-COCO and COCO-OCC for modal and KINS for amodal and evaluated the model's performance with the standard metrics. Even though our occludee-aware TCNet achieves similar results on most segmentation performance, this method also shows the best performance for large objects among SOTA methods using different backbones and object detectors.*

1. Introduction

For a long time, lots of efforts have been made for applying a human vision system to the machine. Traditional algorithm provides basic information and understanding of images based on image feature and computational method in scene [2, 25, 27]. In recent years, the advent of deep neural networks derives the renaissance in computer vision. Despite the rapid development of machine perception, most studies only have been meaningful in the visible area from before to now. However, real world scenes mostly feature multiple objects and appear overlapping each other. In addition, in various situations, a machine may have to determine a specific judgment through a scene in which overlapping objects exist, e.g., in the military field, when a vessel under surveillance by the Navy overlaps, segmentation information on the hidden part is required. Making inference for these invisible areas is called 'Amodal Instance Segmentation'. Early works [8, 20, 21] approach the amodal instance segmentation by using a simple computational segmentation method with ROI box extraction or extra post-processing. Some advanced work has proposed a method for simultaneously regressing both occluder and occludee information in ROI, but allows it to be learned biased against the information of one object. Therefore, segmentation results were smoothed or were not normally completed in the overlapping part between object and object. So, how can we teach the machine to infer well a portion of the objects occluded? The key point is to catch the relationship between the overlapping objects in the image and to maintain a good balance without bias in the part during the training process. If we don't get the relationship between the two objects, problems will arise in the occlusion handling process, resulting in over smooth segmentation than natural object edges. Also, when using mask head architecture, which is biased toward either Occluder or Occludee, misclassified pixels will appear, or it will be difficult to reflect the entire part of the object normally in a network that is segmenting the hidden part.

We proposed the Trilayer Convolutional Network (TCNet). For utilizing the interacted information between occluder and occludee, we use the decoupling image layer method from BCNet [15]. This approach showed higher instance segmentation performance in overlapped images than other methods that regress a single image from ROI. However, the information of the occludee was inferred after the element-wise addition of the ROI feature and occluder segmentation each other. In the inference process, we add an occludee only prediction layer to maintain the information balance of occludee and occluder. Also, by using Graph Convolution Network (GCN) in segmentation prediction process, we can consider non-local relation among object pixels.

We validate our method by using COCO dataset. In modal segmentation, our model achieves comparable results among SOTA methods in large object cases but not in small ones. In Amodal segmentation, we utlize KINS dataset for test out methods. our proposed model can pre-

dict the invisible part of the objects and produce the whole segmentation area of an object even in a situation where the overlapping is severe.

2. Related Work

2.1. Instance Segmentation

There have been a lot of pioneering studies in object detection, establishing a bounding box around all objects within an image. One-stage instance segmentation methods perform segmentation and detection of ROI region simultaneously. PolarMask [30] solves the instance segmentation problem as instance center classification and dense distance regression in a polar coordinate. YOLACT [1] generates prototype masks and predicts per-instance mask coefficients. SOLO [28] introduces the "instance categories" which assign categories to each pixel within an instance according to the instance's location and size. These one-stage methods are efficient because of their simple procedures but are less accurate than the two-stage methods. Two-stage instance segmentation methods detect bounding boxes first and then complete segmentation for each ROI region. FCIS [22] utilizes FCN for semantic segmentation and instance mask proposal to perform instance mask prediction and classification together. Mask R-CNN [12] adds a branch for predicting an object mask parallel with the existing branch for bounding box recognition. MS R-CNN [14] contains a network block to learn the quality of the predicted instance masks. BCNet [16] is a bilayer mask prediction network for addressing the issues of heavy occlusion and overlapping objects in two-stage instance segmentation.

2.2. Amodal Instance Segmentation

The problem of object invisible part prediction and segmentation is instinctively possible for humans, but rather complicated for computers. Nevertheless, amodal instance segmentation has advanced by leaps and bounds. Before neural networks are applied to the instance segmentation task, many computational methods are implemented for the occluded object detection and instance segmentation. John Winn et al. [29] suggest that layout consistent random field using pairwise potentials and instance potential for partially occluded object segmentation. Tianshi Gao et al. in [11] introduce a binary variable for each cell in the bounding box indicating whether the pixels inside it belong to the object when performing detection is reflected by occlusion. After deep learning methods are applied to instance segmentation, Li and Malik [21] firstly propose a method that uses amodal segmentation heatmap while iteration of Iterative Bounding Box proceeds. Since the lack of an amodal instance segmentation dataset, Zhu et al. [32] create a novel datset COCO amodal for amodal instance segmentation, based on the original COCO [23] dataset. According to Kiana Ehsani et al., [9], if the segmentation task is sufficiently trained through amodal and modal masks in a virtual realistic image, it is shown that the object where occlusion occurred can be reconstructed using the GAN network as well as amodal segmentation. Furthermore, the authors show that the trained corresponding network is applicable to the natural image and has produced meaningful results. Zhan et al. [31] propose a self-supervised network that predicts amodal mask and content completion without occlusion ordering and amodal masks. Unlike the previous amodal segmentation model, [10] focuses on the occluded object and infer the occlusion mask by subtracting the visible mask prediction from the amodal mask prediction. Also, it suggests the first end-to-end trainable model for predicting amodal instance masks and provides new datasets with ground truth for semantic amodal segmentation. Lei Ke et al. [16] provide a novel approach for catching the interactions between occluder and occludee. Based on backbone [13] with ROI features, the proposed method utilizes Graph Convolutional Network(GCN) [17] for considering the non-local relationship between pixels and decoupling overlapping objects into two image layers for considering the interactions between them.

3. Method

3.1. Overview

In order to solve the occlusion-aware instance segmentation problem, especially in an amodal setting, we propose the novel trilayer structured pipeline. Figure 1 shows our overall pipeline which consists of two parts; ROI feature extraction part and triple-layered mask prediction parts. Since mask prediction includes three GCN-based layers, we called it a trilayer structured network, the **TCNet**. This method is inspired by the previous work BCNet [15], which consists of a bilayer GCN structure where each layer predicts occluder and occludee, respectively. In detail, we adopt the same ROI feature extraction part from BCNet that uses FPN as backbone and FCOS [26] as bounding box detector. The difference between BCNet is the mask prediction parts. Though BCNet achieves the *state-of-the-art* performance in instance segmentation with occlusion, it is still not performed well in an *amodal* setting. This is because it only uses occluder information as prior knowledge of the occludee prediction. In an amodal instance segmentation problem, the most important part is the prediction accurate boundary of the occluded part. However, only using occluder information is not enough to predict that part. Thus, we modify the layer from bilayer to trilayer.

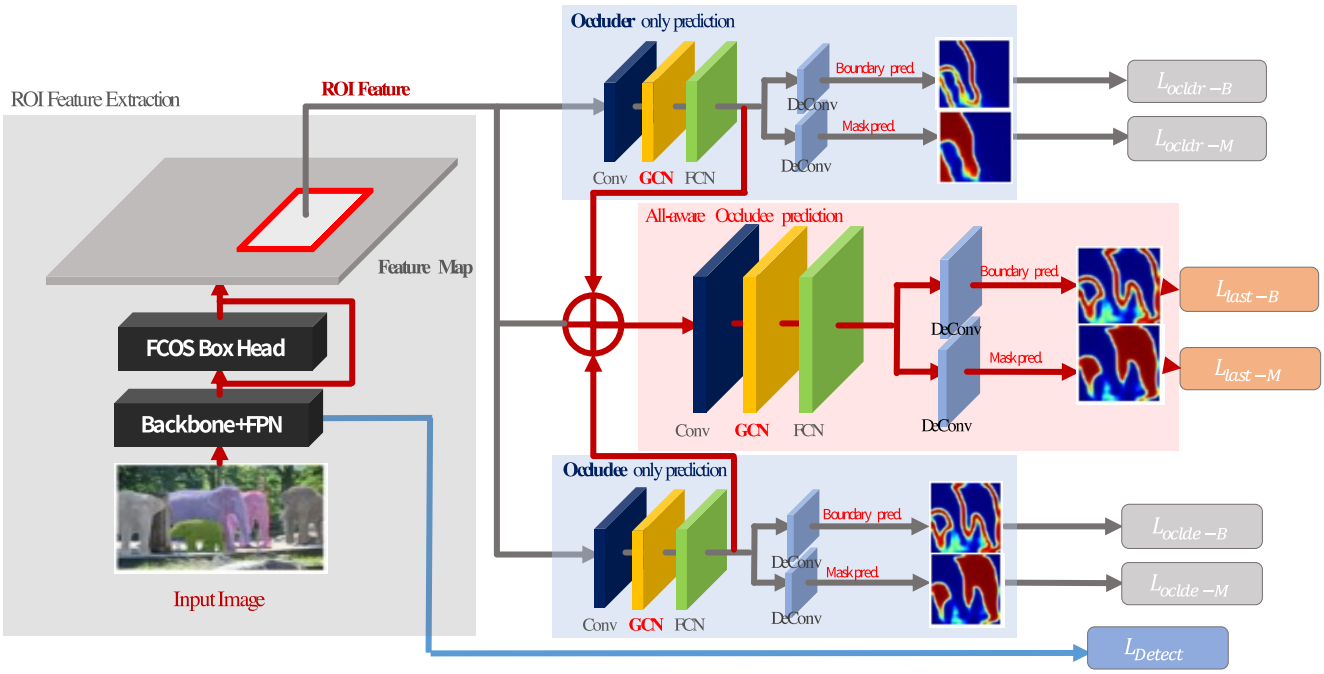

Figure 1. **Overall pipeline of TCNet** In order to solve occlusion-aware instance segmentation, especially amodal setting, we proposed a novel network that consists of ROI feature extraction part and triple-layered prediction parts.

3.2. Trilayer Occlusion Aware Modeling

Trilayer Structure We proposed a trilayer structure in order to consider not only occluder-occludee interaction but also concentrate occludee itself. Layers are divided into two parts. The first part is the occluder-occludee prediction part. Two separated GCN layers get ROI features as an input and compute both the boundary map and mask map. In this step, since these two layers respectively produce the occluder feature and occludee feature, and send this information to the final decision layer (All-aware occludee layer) by adding them to the ROI feature. It enables the final layer to predict the occluded area much better by considering both occluder and occludee. Also, we compute losses for each layer. This makes each layer perform its role as we designed it.

GCN-based Prediction Layer Each prediction layer is based on GCN structure which is first proposed in BCNet. The output of GCN layer is computed by the following formula,

$$Z = \mathrm{ReLU}(\sigma(AXW_g)) + X$$

where X is the input, W_g is the learnable weight matrix, σ is the normalization function, and A is the adjacency matrix which represents pairwise similarity which is defined as,

$$A_{ij} = sigmoid(\theta(x_i)^T \phi(x_j))$$

where θ and ϕ are 1×1 convolution layer, and x_i, x_j are graph nodes.

In our complete model, the output of GCN of the all-aware occludee layer is computed as below formula:

$$\begin{aligned} Z^{ocldr} &= \mathrm{ReLU}(\sigma(A^{ocldr} X^{roi} W_g^{ocldr})) + X^{roi} \\ Z^{oclde} &= \mathrm{ReLU}(\sigma(A^{oclde} X^{roi} W_g^{oclde})) + X^{roi} \\ X^{last} &= X^{roi} + F^{ocldr} Z^{ocldr} + F^{oclde} Z^{oclde} \\ Z^{last} &= \mathrm{ReLU}(\sigma(A^{last} X^{last} W_g^{last})) + X^{last} \end{aligned}$$

Where X^{roi} is the input ROI feature of the layers and $A^{ocldr}, W_g^{ocldr}, A^{oclde}, W_g^{oclde}$ are GCN components of occluder only prediction layer and occludee only prediction layer, respectively. Also, F^{ocldr}, F^{oclde} are FCN of each block. Since we send all features (ROI, occluder, occludee) to the final decision layer (all-aware occludee layer) by summing them.

Loss Function In our model, loss is computed for each part, each layer. We first compute *Detect Loss* in the ROI feature extraction part. *Detect Loss* is computed by the following formula which is borrowed from FCOS [26]:

$$\mathcal{L}_{Detect} = \mathcal{L}_{Regression} + \mathcal{L}_{Centerness} + \mathcal{L}_{Class}$$

For prediction layer, we compute boundary map loss $\mathcal{L}_{ocldr-B}, \mathcal{L}_{oclde-B}, \mathcal{L}_{last-B}$ and mask map loss

Table 1. Effect of the proposed trilayer composed of the occludee GCN, occluder GCN, and occlusion-aware GCN layer.

Occludee GCN Layer		COCO		COCO-OCC	
Contour	Mask	AP	AP_{50}	AP	AP_{50}
		39.6	61.2	30.7	50.6
✓		31.9	47.8	28.9	44.1
	✓	31.6	47.1	25.0	41.3
✓	✓	39.0	56.1	36.1	53.0

$\mathcal{L}_{ocldr-M}, \mathcal{L}_{oclde-M}, \mathcal{L}_{last-M}$ each by comparing with ground truth map. We use binary cross-entropy loss for them. Thus, the total loss can be written as follows:

$$\mathcal{L}_{Total} = \lambda_1\mathcal{L}_{Detect} + \lambda_2\mathcal{L}_{ocldr-B} + \lambda_3\mathcal{L}_{ocldr-M} + \lambda_4\mathcal{L}_{oclde-B} + \lambda_5\mathcal{L}_{oclde-M} + \lambda_6\mathcal{L}_{last-B} + \lambda_7\mathcal{L}_{last-M}$$

4. Experiments

4.1. Setup

Datasets. We utilize the Microsoft COCO dataset, which is a large-scale object detection and segmentation dataset, to train and evaluate the proposed TCNet. The dataset used in the experiments consists of 2017train (118k images), 2017val (5k images), and 2017test-dev (40k images). Also, we use additional dataset, called COCO-OCC, to evaluate segmentation performance that can better deal with occlusion handling. It is a subset split of Microsoft COCO dataset proposed by the authors of baseline model. The COCO-OCC dataset contains a total of 1,005 images extracted from the 2017val where the overlapping ratio between the bounding boxes of objects is at least 0.2. We ultimately aim to propose an amodal instance segmentation model that is robust for occlusion between multiple objects. Therefore, it is extremely important to conduct experiments on COCO-OCC dataset, which is more difficult to perform segmentation task with highly overlapping objects. Actually, it is observed that a performance gap around 3.0AP for the same model exists.

Metrics. We evaluate results on 2017val, 2017test-dev, and 2017occ with the standard metrics used in object detection and segmentation tasks. The precision value inevitably decreases when the recall is increased by 0.1 units from 0 to 1. The AP (Average Precision) is determined by calculating the precision value for each unit and averaging it. It can be computed for each class, and the average value obtained by calculating APs for the total number of classes is mAP. However, mAP is called AP in the Microsoft COCO dataset.

Implementation details. For training, the Detectron2 platform is used and basically, it is recommended to allocate one GPU per batch size 2 in Detectron2, so we use a total of four TITAN Xp GPUs and the batch size is set to 8. SGD with momentum 0.1 is employed as the optimizer for training 180K iterations with 1K constant warm-up iterations and initial learning rate is 0.01. The occluder GCN layer explicitly models occluding regions by simultaneously detecting occlusion contours and masks. However, since there are not many cases in which the partial occlusion occurs in the dataset used for model training, it is not possible to learn the occluder GCN layer well. So, the process of filtering out part of the non-occluded ROI proposals is used to make occlusion cases account for 50% for balance sampling. For inference, the mask head of occlusion-aware GCN layer predicts masks for occludee in the high-score box proposals generated by the faster R-CNN 2-stage or FCOS 1-stage detector, where the occluder and occludee GCN layers produce features considering occlusion relationship between the occluder and occludee as input for the occlusion-aware GCN layer. For training/inference, Detectron2. which is a platform for pytorch-based object detection and semantic segmentation created by Facebook Artificial Intelligence Research (FAIR). Depending on the research and the type of experiment, the batch size recommended to be assigned per GPU is different. We use 4 TITAN Xp GPUs and 4 GeForce RTX 3090 GPUs and each batch size is set to 8 and 4. Since the difference in the model learning speed is very large depending on the batch size allocated per GPU, it is important for the user to set the appropriate value considering the situation.

4.2. Ablation Study

In the ablation study, we validate the effect of trilayer occluder-occludee modeling. The goal of the ablation study is to understand how the proposed trilayer structure helps performance improvement and how the contour and mask of the occludee layer affect model performance. Table 1 shows that the bilayer, baseline model, performs better for COCO dataset that contain images of various situations. However, the proposed trilayer focusing on occludee has the best performance in COCO-OCC dataset containing images of heavy occlusion. In addition, the performance of the model decreases more when the contour prediction is not used than mask prediction. That is, using the contour information is more useful than predicting the mask prediction. The last row of Table 1 demonstrates the importance of the complete trilayer. Compared to the baseline, joint occlusion modeling produces the most apparent improvement, especially for the heavy occlusion cases. Our ablation study leaves much to be desired since all the datasets shown in Table 1 have modal masks, we do not know what the tendency is for the amodal dataset. Due to the lack of time and re-

Table 2. Comparison with SOTA methods on COCO test-dev set.

Method	Backbone	Detector	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Mask R-CNN [12]	ResNet-50	Faster R-CNN	35.6	57.6	38.1	18.7	38.3	46.6
PANet [24]	ResNet-50	Faster R-CNN	36.6	58.0	39.3	16.3	38.1	52.4
BCNet [15]	ResNet-50	Faster R-CNN	38.4	59.6	41.5	21.9	40.9	49.3
Mask R-CNN [12]	ResNet-101	Faster R-CNN	37.0	59.2	39.5	17.1	39.3	52.9
MaskLab [5]	ResNet-101	Faster R-CNN	37.3	59.8	39.6	19.1	40.5	50.6
Mask Scoring R-CNN [14]	ResNet-101	Faster R-CNN	38.3	58.8	41.5	17.8	40.4	54.4
BMask R-CNN [7]	ResNet-101	Faster R-CNN	37.7	59.3	40.6	16.8	39.9	**54.6**
HTC [4]	ResNet-101	Faster R-CNN	39.7	**61.8**	**43.1**	21.0	42.2	53.5
BCNet [15]	ResNet-101	Faster R-CNN	**39.8**	61.5	**43.1**	**22.7**	**42.4**	51.1
YOLACT [1]	ResNet-101	FCOS	31.2	50.6	32.8	12.1	33.3	47.1
TensorMask [6]	ResNet-101	FCOS	37.1	59.3	39.4	17.4	39.1	51.6
ShapeMask [18]	ResNet-101	FCOS	37.4	58.1	40.0	16.1	40.1	53.8
CenterMask [19]	ResNet-101	FCOS	38.3	-	-	17.7	40.8	54.5
BlendMask [3]	ResNet-101	FCOS	38.4	60.7	41.3	18.2	41.5	53.3
BCNet [15]	ResNet-101	FCOS	**39.6**	**61.2**	**42.7**	**22.3**	42.3	51.0
Ours	ResNet-101	FCOS	39.0	56.1	42.2	20.5	**43.4**	**54.9**

sources, the results of the ablation study conducted with the KINS dataset were not completed by the submission date.

4.3. Performance Comparison with SOTA Methods

Table 2 compares TCNet with state-of-the-art instance segmentation methods on COCO dataset. Unfortunately, Table 2 shows that from an AP perspective, BCNet performs better than the proposed TCNet. However, as for larger objects, our model achieves comparable result among SOTA methods. Compared to BCNet, AP_M and AP_L have higher segmentation performance by 1.1 and 3.9, respectively. Presumably, our model focuses on occludee hidden by other objects, we conclude that the larger the object, the more information about occludee, so that the model shows better performance. One thing to note is that based on the experience of conducting the experiment through the Detectron2 platform, the difference in iterations defined in Detectron2 leads to a large performance difference. We believe that some performance differences with BCNet can be sufficiently covered by hyperparameter fine-tuning.

4.4. Visualization of Occlusion Handling

Figure 2 and Figure 5 show the qualitative comparison on COCO dataset, which has model mask ground truth. The modal instance segmentation results have been performed neatly despite the presence of many and various objects in the image and to some extent occluded by multiple objects. And Figure 3 shows the results of the amodal instance segmentation, which is our most main task. Although several bicycles and cars are parked in a state of overlap in images, our proposed model predicts even the invisible part of the objects and creates the shape of them well. The visualization results discussed above show only the successful aspects of TCNet. However, failure cases must also be examined. However, we are noteworthy about the failure cases like Figure 4. Previous studies have shown that the prediction for occluder, where the shape of the object can be almost completely identified, is good, but the prediction for occludee varies widely. On the other hand, our model is good at predicting occludee and rather the prediction results for occluder, which is easy to predict, are strange.

We reconsider our initial intentions to determine the cause of these results. We think it important to focus on occludee in order to ultimately make occludee prediction with occlusion handling. Therefore, it is pointed out as a problem that there is no branch for learning only occludee in BCNet. Unfortunately, excessive weights are assigned to occludee due to the insufficient hyperparameter fine-tuning. And strangely, the prediction result for occludee seems to be better than occluder. If we think about the relationship between trilayer's components, in the end, two branches out of three are structures for just occludee. It is presumed that the results such as Figure 4 is obtained by such biased weight allocation.

5. Conclusion

This study presents an advanced amodal instance segmentation method based on BCNet, including an additional

Figure 2. Modal Instance Segmentation on COCO validation set by TCNet. The visualization results show that model instance segmentation has been conducted neatly in spite of the presence of various objects with overlapping.

Figure 3. Success case of amodal instance segmentation on KINS test set. Although several bicycles and cars are parked in an overlapping form, our model predicts even the invisible part of the objects and create the shape of them well.

GCN layer for the occludee for an effective mask prediction in two-stage instance segmentation. Our occludee-aware TCNet achieves the best performance for large objects among SOTA methods using same backbones and object detectors, which are used in other SOTA methods. The experimental results reveal that the precise prediction of amodal and, in particular, invisible masks is a difficult task, so further work in this research to predict occluded areas more precisely will be required. Reconstruction of the loss function and hyperparameter optimization and applying a generative model from GAN or diffusion model would be the next steps of further research to predict occluded areas more precisely. The suggested model's analysis of the occluded layer using an attention map might be needed to validate its work. There is no consideration for uncertainty in this study. If we can find an optimal hyperparameter for balancing the components of trilayer, we can make our research more meaningful by adding uncertainty modeling to the mask prediction process. Since the proposed model is

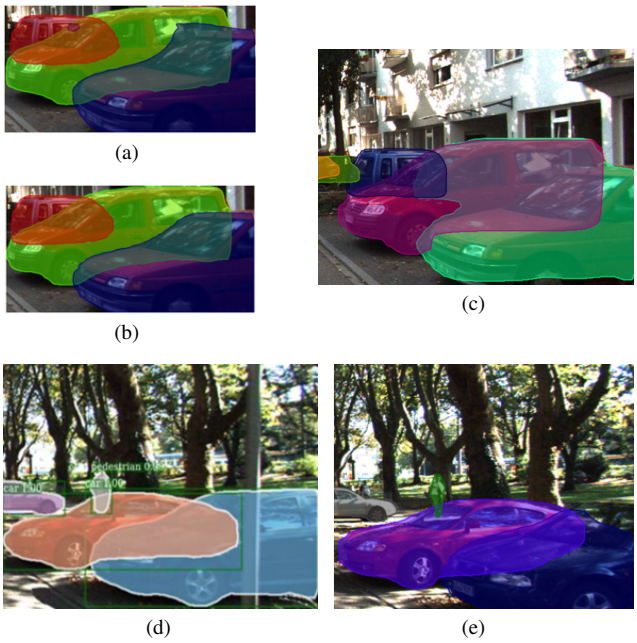

Figure 4. (a) Mask R-CNN + ASN (b) BCNet (c) Ours: TCNet (d) Mask R-CNN (e) Ours: TCNet. Compared to the mask for red car and pedestrian predicted by (d), our model certainly generates occludee masks well.

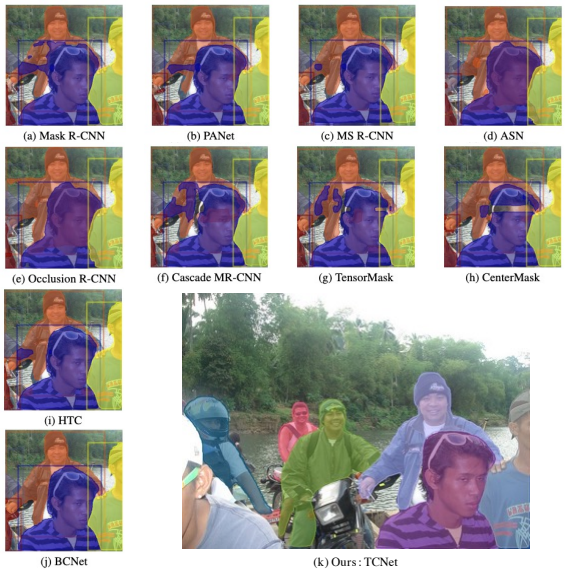

Figure 5. Modal Instance Segmentation on COCO validation set by SOTA methods and TCNet. Compared to other SOTA methods, our proposed model performs segmentation almost perfectly.

good at predicting occludee and robust to occlusion, it will be used as an effective approach for occlusion handling. We expect our proposed method will bring benefits in both occlusion handling and instance segmentation for occludee.

References

[1] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. YOLACT: real-time instance segmentation. *Proceedings of the IEEE/CVF international conference on computer vision*, 2019. 2, 5

[2] John Canny. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6):679–698, 1986. 1

[3] Hao Chen, Kunyang Sun, Zhi Tian, Chunhua Shen, Yongming Huang, and Youliang Yan. Blendmask: Top-down meets bottom-up for instance segmentation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 5

[4] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. Hybrid task cascade for instance segmentation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 5

[5] Liang-Chieh Chen, Alexander Hermans, George Papandreou, Florian Schroff, Peng Wang, and Hartwig Adam. Masklab: Instance segmentation by refining object detection with semantic and direction features. *CoRR*, abs/1712.04837, 2017. 5

[6] Xinlei Chen, Ross Girshick, Kaiming He, and Piotr Dollar. Tensormask: A foundation for dense object segmentation. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 5

[7] Tianheng Cheng, Xinggang Wang, Lichao Huang, and Wenyu Liu. Boundary-preserving mask R-CNN. *European Conference on Computer Vision*, 2019. 5

[8] Jifeng Dai, Kaiming He, and Jian Sun. Instance-aware semantic segmentation via multi-task network cascades. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3150–3158, 2016. 1

[9] Kiana Ehsani, Roozbeh Mottaghi, and Ali Farhadi. Segan: Segmenting and generating the invisible. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6144–6153, 2018. 2

[10] Patrick Follmann, Rebecca König, Philipp Härtinger, Michael Klostermann, and Tobias Böttger. Learning to see the invisible: End-to-end trainable amodal instance segmentation. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1328–1336. IEEE, 2019. 2

[11] Tianshi Gao, Benjamin Packer, and Daphne Koller. A segmentation-aware object detection model with occlusion handling. In *CVPR 2011*, pages 1361–1368. IEEE Computer Society, 2011. 2

[12] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. *Proceedings of the IEEE international conference on computer vision*, 2017. 2, 5

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2

[14] Zhaojin Huang, Lichao Huang, Yongchao Gong, Chang Huang, and Xinggang Wang. Mask scoring R-CNN. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019. 2, 5

[15] Lei Ke, Yu-Wing Tai, and Chi-Keung Tang. Deep occlusion-aware instance segmentation with overlapping bilayers. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 1, 2, 5

[16] Lei Ke, Yu-Wing Tai, and Chi-Keung Tang. Deep occlusion-aware instance segmentation with overlapping bilayers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4019–4028, 2021. 2

[17] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016. 2

[18] Weicheng Kuo, Anelia Angelova, Jitendra Malik, and Tsung-Yi Lin. Shapemask: Learning to segment novel objects by refining shape priors. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 5

[19] Youngwan Lee and Jongyoul Park. Centermask: Real-time anchor-free instance segmentation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 5

[20] Ke Li, Bharath Hariharan, and Jitendra Malik. Iterative instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3659–3667, 2016. 1

[21] Ke Li and Jitendra Malik. Amodal instance segmentation. In *European Conference on Computer Vision*, pages 677–693. Springer, 2016. 1, 2

[22] Yi Li, Haozhi Qi, Jifeng Dai, Xiangyang Ji, and Yichen Wei. Fully convolutional instance-aware semantic segmentation. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016. 2

[23] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 2

[24] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. *CoRR*, abs/1803.01534, 2018. 5

[25] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 1

[26] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. FCOS: Fully convolutional one-stage object detection. In *Proc. Int. Conf. Computer Vision (ICCV)*, 2019. 2, 3

[27] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, volume 1, pages I–I. Ieee, 2001. 1

[28] Xinlong Wang, Tao Kong, Chunhua Shen, Yuning Jiang, and Lei Li. SOLO: segmenting objects by locations. *European Conference on Computer Vision*, 2019. 2

[29] John Winn and Jamie Shotton. The layout consistent random field for recognizing and segmenting partially occluded ob-

jects. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 1, pages 37–44. IEEE, 2006. 2

[30] Enze Xie, Peize Sun, Xiaoge Song, Wenhai Wang, Xuebo Liu, Ding Liang, Chunhua Shen, and Ping Luo. Polarmask: Single shot instance segmentation with polar representation. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019. 2

[31] Xiaohang Zhan, Xingang Pan, Bo Dai, Ziwei Liu, Dahua Lin, and Chen Change Loy. Self-supervised scene deocclusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3784–3792, 2020. 2

[32] Yan Zhu, Yuandong Tian, Dimitris Metaxas, and Piotr Dollár. Semantic amodal segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1464–1472, 2017. 2