Joint COCO and Mapillary Workshop at ICCV 2019: COCO Panoptic Segmentation Challenge Track

Technical Report: Scene Overlap Graph for Panoptic Segmentation

Yibo Yang[1,2], Xia Li[2,3], Hongyang Li[2], Tiancheng Shen[1,2], Yudong Liu[4], Zhouchen Lin[2]

[1]Academy for Advanced Interdisciplinary Studies, Peking University

[2]Key Laboratory of Machine Perception (MOE), School of EECS, Peking University

[3]Key Laboratory of Machine Perception, ShenZhen Graduate School, Peking University

[4]Wangxuan Institute of Computer Technology, Peking University

{ibo,ethanlee,lhy_ustb@pku.edu.cn,tianchengShen,bahuangliuhe,zlin}@pku.edu.cn

Abstract

The panoptic segmentation task requires a unified result from semantic and instance segmentation outputs that may contain overlaps. However, current methods widely ignore modeling overlaps. In this challenge, we model overlap relations among instances and resolve them for panoptic segmentation in a differentiable way. Inspired by scene graph representation, we formulate the overlapping problem as a simplified case, named scene overlap graph. We leverage each object's category, geometry and appearance features to perform relational embedding, and output a relation matrix that encodes overlap relations. In order to overcome the lack of supervision, we introduce a differentiable module to resolve the overlap between any pair of instances. The mask logits after removing overlaps are fed into per-pixel instance `id` *classification, which leverages the panoptic supervision to assist in the modeling of overlap relations. In experiments on MS-COCO, we demonstrate that our method is able to accurately predict overlap relations, and outperform the state-of-the-art method, UPSNet. Ablation studies are reported to show how we improve the performance. No external data is used.*

1. Introduction

Recently, the panoptic segmentation task introduced in [7] aims to unify the results of semantic and instance segmentation into a single pipeline. The system performs semantic segmentation for pixels that belong to amorphous background scenes, named *stuff*. For countable foreground objects, named *things*, the goal is to assign each object region with the right thing class, as well as an instance `id`, identifying which object it belongs to. As a result, panoptic segmentation cannot have overlapping segments. However, the outputs of semantic and instance segmentation may have conflicted predictions. Besides, most cutting-edge high-performance instance segmentation methods [2] adopt the region-based strategy [1], and output overlapping segments. As shown in Figure 1, the object pairs, such as *cup-dinning table*, *bottle-dinning table*, and *bowl-dinning table*, share overlapping regions from instance segmentation. Therefore, resolving overlaps and producing coherent segmentation results are the main challenge for panoptic segmentation [7].

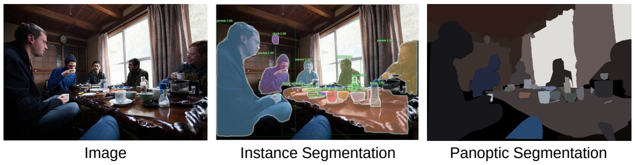


Figure 1: Instance segmentation has overlapping regions for objects, while panoptic segmentation requires a unified result for each pixel. Our method explicitly predicts overlap relations and resolve overlaps in a differentiable way.

In [7], the semantic and instance segmentation are trained separately, and their panoptic results are merged by heuristic post-processing steps. Later studies aim to unify the semantic and instance segmentation into an end-to-end training framework [6, 10, 12, 16, 14, 17, 9]. The panoptic results are usually produced by fusion strategies [6, 10], or predicted by a panoptic head [12, 16]. These studies do not explicitly model overlap relations of objects. However, modeling overlap is challenging without the supervision of object relations or depth information.

In this study, we introduce the scene overlap graph network (SOGNet) for panoptic segmentation. The SOGNet consists of four components: the joint segmentation, the relational embedding module, the overlap resolving module, and the panoptic head.

Similar to [6, 10, 12, 16, 14, 9], we also use ResNets [3] with feature pyramid network (FPN) [11] as the shared backbone of joint segmentation. Inspired by the relation classification in scene graph generation tasks [19, 15], we formulate the overlap problem in panoptic segmentation as a simplified scene graph with directed edges, in which there are only three relation types for instance i with respect to j: no overlap, covering as a subject, and being covered as an object. We name this representation as *scene overlap graph*. We leverage the category, geometry, and appearance information of objects to perform relational embedding, and output a matrix that explicitly encodes overlap relations. However, different from scene graph parsing tasks using the Visual Genome dataset that has relation annotations, the panoptic segmentation task does not offer annotations of object relations or depth information, so the overlap relations cannot be modeled with direct supervision.

In order to overcome this problem, we further develop the overlap resolving module, which resolves the overlaps between any pair of instances in a differentiable way. The mask logits after removing overlaps are then used for per-pixel instance id classification in the panoptic head. In doing so, the supervision from pixel-level classification helps the instance-level modeling of overlap relations.

2. Our SOGNet

In the scene graph generation task, objects in an image are constructed as a graph and their relations are directed edges. We formulate the overlapping problem in panoptic segmentation as a similar structure, named scene overlap graph (SOG). Our proposed SOGNet consists of four components. The joint segmentation connects semantic and instance segmentation in a unified network. The relational embedding module explicitly encodes overlap relations of objects. After the overlap resolving module, overlaps among instances are removed in a differentiable way. Finally, the panoptic head performs per-pixel instance id classification. An illustration of our SOGNet architecture is shown in Figure 2.

2.1. Joint Segmentation

Following current popular methods, we use ResNet with FPN as the shared backbone of semantic and instance segmentation branches. The Mask R-CNN structure is adopted for our instance segmentation head, which outputs the box regression, class prediction, and mask segmentation for foreground objects. As for semantic head, following [16], the FPN feature maps first go through three deformable 3×3 convolution layers, and then are up-sampled to the $1/4$ scale. Finally, they are concatenated to generate the per-pixel category prediction. This branch is supervised with both stuff and thing classes, and then the semantic logits of stuff classes are extracted into the panoptic head. We train our model using instance and panoptic annotation. The panoptic annotation that gives per-pixel category and instance id supervises the semantic and panoptic head, respectively. The instance annotation contains overlaps and is used for instance segmentation.

2.2. Relational Embedding Module

For any training image, we are given the ground truth $\{b_i, c_i, M_i\}_{i=1}^{N_{inst}}$, where b_i, c_i, and M_i refer to the bounding box, one-hot category, and corresponding mask for instance i, respectively, and N_{inst} is the number of instances in this image. As illustrated in Figure 2, we perform relational embedding using the ground truth in the training phase. During inference, we replace them with the prediction from Mask R-CNN branch. The $b_i \in \mathcal{R}^4$ and $c_i \in \mathcal{R}^{80}$ (there are 80 thing classes for MS-COCO) encode geometry and category information, respectively. In order to include appearance feature, we resize the values inside box b_i from M_i as 28×28, which is consistent with the size of Mask R-CNN's output. The resized mask is flattened to be a vector, denoted as $m_i \in \mathcal{R}^{784}$.

The bilinear pooling method learns joint representation for pair of features and is widely applied to visual question answering [5], and image recognition [18] tasks. We construct our category and appearance relation features using low-rank outer product in [5]. For a pair of instances i and j, their category relation feature is calculated as:

$$E_{i|j}^{(c)} = P^T \left(\sigma(V^T c_i) \circ \sigma(U^T c_j)\right), \tag{1}$$

where $\circ$ denotes the Hadamard product (element-wise multiplication), σ is the ReLU non-linear activation, V and U are two linear embeddings that project the input into subject and object features, respectively, and P maps the relation feature into output dimension d_c. We then have the category relation features as:

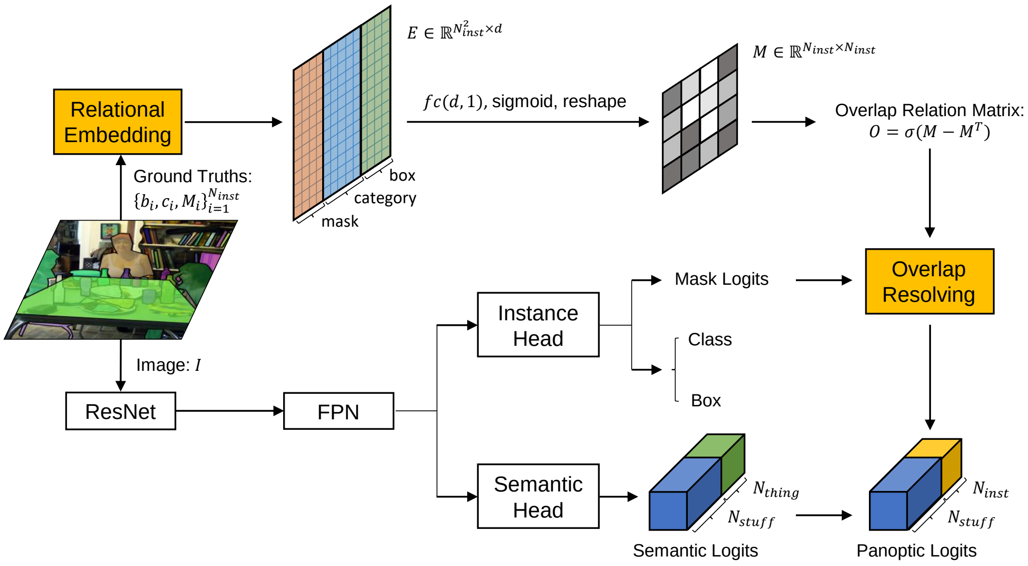
$$E^{(c)} = \left[E_{1|1}^{(c)}, E_{1|2}^{(c)}, \cdots, E_{N_{inst}|N_{inst}}^{(c)}\right]^T \in \mathcal{R}^{N_{inst}^2 \times d_c}, \tag{2}$$

where "[ ]" is the concatenation operation. In a similar way, using m_i as the input of Eq. (1), we can also construct the appearance relation features $E^{(m)} \in \mathcal{R}^{N_{inst}^2 \times d_m}$.

The relative geometry provides strong information to infer whether two objects have overlap or not. Following [4, 15], we have the translation- and scale-invariant relative geometry feature encoded as:

$$E_{i|j}^{(b)} = K^T \left(\frac{x_i - x_j}{w_j}, \frac{y_i - y_j}{h_j}, \log\left(\frac{w_i}{w_j}\right), \log\left(\frac{h_i}{h_j}\right)\right)^T, \tag{3}$$

where x_i, y_i, w_i, h_i are coordinates and scales extracted from b_i, and $K \in \mathcal{R}^{4 \times d_b}$ is a linear matrix that maps the 4-dimensional relative geometry feature into high-dimensional d_b. We can further have the geometry relation



Relational Embedding
Ground Truths: $\{b_i, c_i, M_i\}_{i=1}^{N_{inst}}$
$E \in \mathbb{R}^{N_{inst}^2 \times d}$
box
category
mask
$fc(d,1)$, sigmoid, reshape
$M \in \mathbb{R}^{N_{inst} \times N_{inst}}$
Overlap Relation Matrix: $O = \sigma(M - M^T)$
Image: I
ResNet
FPN
Instance Head
Mask Logits
Class
Box
Overlap Resolving
Semantic Head
N_{thing}
N_{stuff}
Semantic Logits
N_{inst}
N_{stuff}
Panoptic Logits

Figure 2: An illustration of the SOGNet for panoptic segmentation. The instance ground truths are input of our relational embedding module. During inference, they are replaced with the predictions from the instance segmentation head. The architecture is trained in an end-to-end manner. σ denotes the ReLU non-linear function.

features $E^{(b)} \in \mathcal{R}^{N_{inst}^2 \times d_b}$. We concatenate these edge representations about appearance, category, and geometry as:

$$E = [E^{(m)}, E^{(c)}, E^{(b)}] \in \mathcal{R}^{N_{inst}^2 \times d}, \tag{4}$$

where $d = d_m + d_c + d_b$. The relational embedding is then used to encode overlap relations.

2.3. Overlap Resolving Module

Based on relational embedding, we introduce the overlap resolving module to explicitly model overlap relations and resolve overlaps among instances in a differentiable way.

As illustrated in Figure 2, the relation features, $E \in \mathcal{R}^{N_{inst}^2 \times d}$, go through a $fc(d, 1)$ layer to have a single-channel output with the sigmoid activation to restrict the values within $(0, 1)$. We reshape the output as a square matrix, denoted as $M \in \mathcal{R}^{N_{inst} \times N_{inst}}$. The element M_{ij} has a physical meaning to represent the potential of instance i being covered by instance j. Because there can be only one overlap relation between instances i and j, we then introduce the overlap relation matrix defined as:

$$O = \sigma(M - M^T) \in \mathcal{R}^{N_{inst} \times N_{inst}}, \tag{5}$$

where σ denotes the ReLU activation that is used to filter out the negative differences between potentials on symmetric positions. In doing so, if $O_{ij} > 0$, it encodes that instance i is being covered by instance j, while on its symmetric position, $O_{ji} = 0$. When $O_{ij} = O_{ji} = 0$, the instances i and j do not have overlaps. Besides, all diagonal elements O_{ii} equals to 0. As explained later, the positive elements in O will be optimized towards 1 in implementations. We now show how to leverage the overlap relation matrix O to resolve overlaps.

For each bounding box, b_i, of the ground truth instances, we have its mask logits (the activations before sigmoid) of 28×28 from the Mask R-CNN output. We then interpolate these logits back to the image scale $H \times W$ by bilinear interpolation and padding outside the box. These logits, denoted as $\{A_i\}_{i=1}^{N_{inst}}$, may have overlaps because Mask R-CNN is region-based and operates on each region independently. Using the matrix O, we can deal with the overlaps between instances i and j as:

$$A'_i = A_i - A_i \circ [s(A_i) \circ s(A_j)]\, O_{ij}, \tag{6}$$

where A'_i is the output logit of instance i, and s represents the sigmoid activation that turns the logit A_i into a binary-like mask $s(A_i)$. The element-wise multiplication, $s(A_i) \circ s(A_j)$, calculates the intersecting region between instances i and j. The value O_{ij} decides whether the elements in intersecting region should be removed from the logit A_i. When O_{ij} approaches 1, O_{ji} equals to 0, thus the logit A_j will not be affected, and vice versa.

Considering the overlap relations of all the other instances on i, we have:

$$A'_i = A_i - A_i \circ s(A_i) \circ \sum_{j=1}^{N_{inst}} s(A_j) O_{ij}, \tag{7}$$

and then the computational steps of the overlap resolving module can be formulated as:

$$\mathcal{A}' = \mathcal{A} - \mathcal{A} \circ s(\mathcal{A}) \circ \left(s(\mathcal{A}) \times_3 O^T \right), \tag{8}$$

where $\mathcal{A} = [A_1, A_2, \cdots, A_{N_{inst}}] \in \mathcal{R}^{H \times W \times N_{inst}}$, and $\times_3$ denotes the Tucker product along the 3-rd dimension (reshape $s(\mathcal{A})$ as $\mathcal{R}^{HW \times N_{inst}}$ for inner product with O^T, and then return to $\mathcal{R}^{H \times W \times N_{inst}}$). We see that our module is friendly to tensor operations in current deep learning frameworks, and is differentiable for resolving overlaps, so that the SOGNet can be trained in an end-to-end fashion.

2.4. Panoptic Head

The overlap relation matrix, O, explicitly encodes whether there is intersection between any pair of instances, and if there is, the overlapping region should be removed from which instance. However, we are not provided with the supervision of overlap relations by the panoptic segmentation task. Because accurately resolving overlaps has a strong correlation with the quality of final panoptic output, we can exploit the pixel-level panoptic annotation to help modeling overlap relations encoded by O. As illustrated in Figure 2, the instance logits $\mathcal{A}'$ after the SOG module are then fed into the panoptic head.

Following UPSNet [16], we also incorporate the logits from semantic head into the mask logits $\mathcal{A}'$ from instance segmentation. We get the logits of i-th object from semantic output X_i by taking the values inside its ground truth box B_i from the channel corresponding to its ground truth category C_i, and padding zeros outside the box. In UPSNet, they are combined by addition, which is denoted as "Panoptic Head 1". Here we develop an improved combination denoted as "Panoptic Head 2". They are compared as:

$$\text{Panoptic Head 1}: \quad Z_i = X_i + A_i', \tag{9}$$

$$\text{Panoptic Head 2}: \quad Z_i = k \cdot X_i \circ s(A_i') + A_i', \tag{10}$$

where Z_i is the combined logit, s denotes the sigmoid function and k is a factor to balance the numerical difference between semantic output values and mask logits. We set k to be 2 in our experiments. Finally, we concatenate the combined instance logits $\mathcal{Z}_{inst} = [Z_1, ..., Z_{N_{inst}}]$ and the stuff logits $\mathcal{Z}_{stuff}$ from the semantic head to perform per-pixel instance `id` classification with the standard cross entropy loss function, $\mathcal{L}_{panoptic}$.

Despite we do not have the supervision to know which instance lies on the other one, we can leverage the ground truth binary masks, $\{M_i\}_{i=1}^{N_{inst}}$, to infer whether two instances have overlaps or not. We produce a symmetric matrix $R \in \mathcal{R}^{N_{inst} \times N_{inst}}$ defined as:

$$R_{ij} = \mathbb{1}\left[\frac{|M_i \circ M_j|}{\min\{|M_i|, |M_j|\}} \geq 0.1\right], \quad i \neq j, \tag{11}$$

where $|\cdot|$ calculates the area of a binary mask through sum operation, $\circ$ calculates the intersection mask through element-wise multiplication, and $\mathbb{1}$ denotes the indicator function that equals to 1 when the condition holds. All diagonal elements R_{ii} are filled with 0. When $R_{ij} = R_{ji} = 1$, it indicates that the overlapped intersection over the smaller object is larger than 0.1, which means there is a significant overlap between instances i and j. With the symmetric matrix R, we can introduce the relation loss as:

$$\mathcal{L}_R = \frac{1}{N_{inst}^2} \left\| O + O^T - R \right\|_F^2, \tag{12}$$

which calculates the mean squared error between $(O + O^T)$ and R. In doing so, when there is overlap between instances i and j, *i.e.*, $R_{ij} = R_{ji} = 1$, the overlap relation O_{ij} or O_{ji} is forced to approach 1, so that it will not contribute trivially when removing overlaps by Eq. (6).

In total, our SOGNet has the loss functions for semantic and instance segmentation, the panoptic loss $\mathcal{L}_{panoptic}$ for instance `id` classification, and the relation loss $\mathcal{L}_R$ to help optimizing the overlap relation matrix O.

3. Experiments

3.1. Ablation Studies

For our ablation studies, we conduct experiments on the COCO validation set to show how we improve the performance, and compare our SOGNet with UPSNet and other methods. We adopt the same training and inference schemes as UPSNet for fair comparison. The details is described in the corresponding paper [16].

As shown in Table 1, the performance of SOGNet is better than current state-of-the-art methods [6, 10, 8, 16]. All models in Table 1 use ResNet-50 as backbone. We also re-implement UPSNet in the same environment as SOGNet for fair comparison. Our reimplementation has nearly the same results as reported in [16]. UPSNet proposes to construct a void channel to predict the unknown class. When void prediction is enabled, SOGNet has a 1% PQ improvement over UPSNet. When void prediction is not used, SOGNet achieves a higher performance and has a 1.5% PQ improvement over UPSNet. We visualize the overlap relations encoded by the matrix O in Figure 3. It is shown that our method explicitly predicts overlap relations, such as *tie→person→bus*, and *spoon→cup→dinning table*, which are accurate as common sense.

3.2. Submitted Entry to Challenge

On *test-dev* set, we use ResNet-101 as backbone and adopt a longer training schedule and multi-scale training. As showin in Table 2, SOGNet has a performance of 48.2% PQ, which is the current single-model state of the art. For our submitted entry to challenge, we replace the instance segmentation results with CBNet [13] that uses Cascade Mask RCNN+ResNeXt-152 and has a mask mAP of 43.3. Given the overlap relations predicted by SOGNet, we fuse the instance segmentation from CBNet with the semantic segmentation from our SOGNet-Res101, to produce the panoptic result, which leads to a 50.0% PQ.

Models	PQ	SQ	RQ	PQth	PQst
Other Studies					
Panoptic FPN [6]	39.0	-	-	45.9	28.7
AUNet [10]	39.6	-	-	49.1	25.2
OCFusion [8]	41.2	77.1	50.6	49	29
Comparison with UPSNet (use void prediction)					
UPSNet	42.5	78.1	52.5	48.6	33.4
SOGNet (PH1)	43.1	78.6	53.2	49.3	**33.7**
SOGNet (PH2)	**43.5**	**79**	**53.4**	**50.1**	33.6
Comparison with UPSNet (no void prediction)					
UPSNet	42.2	78.3	52.2	48.0	**33.4**
SOGNet (PH 1)	43.0	78.1	53.1	49.3	33.3
SOGNet (PH 2)	**43.7**	**78.7**	**53.5**	**50.6**	33.2

Table 1: Compare SOGNet with UPSNet and other methods on *val* set. All models use ResNet-50 as backbone. SOGNet and UPSNet are implemented in the same environment for fair comparison. No augmentation schemes such as the multi-scale training and testing, flipping are used. "PH 1 / 2" denotes the "Panoptic Head 1 / 2", respectively.

Models	backbone	PQ	SQ	RQ
Megvii	ensemble	53.2	83.2	62.9
Caribbean	ensemble	46.8	80.5	57.1
PKU-360	ResNeXt-152	46.3	79.6	56.1
AUNet [10]	ResNeXt-152	46.5	81.0	56.1
UPSNet [16]	ResNet-101	46.6	80.5	56.9
SOGNet	ResNet-101	48.2	81.5	57.7
submitted entry	ResNeXt-152	50.0	81.8	60.0

Table 2: Performance on *test-dev*. The first block shows the results in Challenge 2018. The second block shows the single model performance of SOGNet and other methods. The last row shows our submitted entry.

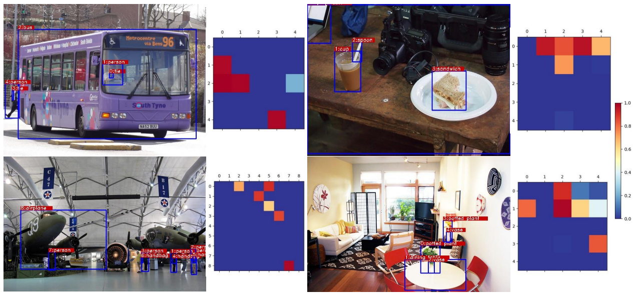


Figure 3: Overlap relations predicted by our method. The map in the right side of each figure is the overlap relation matrix O. Note that the activation on location (i, j) represents that the instance i is covered by (lies below) j. Our method accurately model overlap relations (zoom in to see).

References

[1] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 1

[2] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, pages 2961–2969, 2017. 1

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 2

[4] Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, and Yichen Wei. Relation networks for object detection. In *CVPR*, pages 3588–3597, 2018. 2

[5] Jin-Hwa Kim, Kyoung-Woon On, Woosang Lim, Jeonghee Kim, Jung-Woo Ha, and Byoung-Tak Zhang. Hadamard product for low-rank bilinear pooling. In *ICLR*, 2017. 2

[6] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *CVPR*, 2019. 1, 2, 4, 5

[7] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic segmentation. In *CVPR*, 2019. 1

[8] Justin Lazarow, Kwonjoon Lee, and Zhuowen Tu. Learning instance occlusion for panoptic segmentation. *arXiv preprint arXiv:1906.05896*, 2019. 4, 5

[9] Jie Li, Allan Raventos, Arjun Bhargava, Takaaki Tagawa, and Adrien Gaidon. Learning to fuse things and stuff. *arXiv preprint arXiv:1812.01192*, 2018. 1, 2

[10] Yanwei Li, Xinze Chen, Zheng Zhu, Lingxi Xie, Guan Huang, Dalong Du, and Xingang Wang. Attention-guided unified network for panoptic segmentation. In *CVPR*, 2019. 1, 2, 4, 5

[11] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, pages 2117–2125, 2017. 2

[12] Huanyu Liu, Chao Peng, Changqian Yu, Jingbo Wang, Xu Liu, Gang Yu, and Wei Jiang. An end-to-end network for panoptic segmentation. In *CVPR*, 2019. 1, 2

[13] Yudong Liu, Yongtao Wang, Siwei Wang, Tingting Liang, Qijie Zhao, Zhi Tang, and Haibin Ling. Cbnet: A novel composite backbone network architecture for object detection. *arXiv preprint arXiv:1909.03625v1*, 2019. 4

[14] Lorenzo Porzi, Samuel Rota Bulò, Aleksander Colovic, and Peter Kontschieder. Seamless scene segmentation. *arXiv preprint arXiv:1905.01220*, 2019. 1, 2

[15] Sanghyun Woo, Dahun Kim, Donghyeon Cho, and In So Kweon. Linknet: Relational embedding for scene graph. In *NeurIPS*, pages 560–570, 2018. 2

[16] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersin Yumer, and Raquel Urtasun. Upsnet: A unified panoptic segmentation network. In *CVPR*, 2019. 1, 2, 4, 5

[17] Tien-Ju Yang, Maxwell D Collins, Yukun Zhu, Jyh-Jing Hwang, Ting Liu, Xiao Zhang, Vivienne Sze, George Papandreou, and Liang-Chieh Chen. Deeperlab: Single-shot image parser. *arXiv preprint arXiv:1902.05093*, 2019. 1

[18] Chaojian Yu, Xinyi Zhao, Qi Zheng, Peng Zhang, and Xinge You. Hierarchical bilinear pooling for fine-grained visual recognition. In *ECCV*, pages 574–589, 2018. 2

[19] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *CVPR*, pages 5831–5840, 2018. 2