Amodal Instance Segmentation via Prior-Guided Expansion

Junjie Chen[1], Li Niu[1]*, Jianfu Zhang[1], Jianlou Si[2], Chen Qian[2], Liqing Zhang[1]*

[1] The MoE Key Lab of AI, CSE department, Shanghai Jiao Tong University
[2] SenseTime Research, SenseTime
{chen.bys, ustcnewly, c.sis}@sjtu.edu.cn, {sijianlou,qianchen}@sensetime.com, zhang-lq@cs.sjtu.edu.cn

Abstract

Amodal instance segmentation aims to infer the amodal mask, including both the visible part and occluded part of each object instance. Predicting the occluded parts is challenging. Existing methods often produce incomplete amodal boxes and amodal masks, probably due to lacking visual evidences to expand the boxes and masks. To this end, we propose a prior-guided expansion framework, which builds on a two-stage segmentation model (i.e., Mask R-CNN) and performs box-level (resp., pixel-level) expansion for amodal box (resp., mask) prediction, by retrieving regression (resp., flow) transformations from a memory bank of expansion prior. We conduct extensive experiments on KINS, D2SA, and COCOA cls datasets, which show the effectiveness of our method.

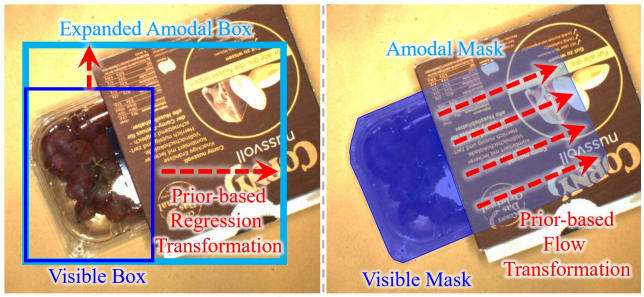
1 Introduction

Instance segmentation (*e.g.*, Mask R-CNN (He et al. 2017)) focuses on segmenting visible pixels for each object instance. In real-world images, the object instances usually partially occlude each other. To better parse the complex scene, amodal instance segmentation (Xiao et al. 2021; Li and Malik 2016) requires to infer the complete amodal mask, including both the visible region and the occluded region for each object instance. Such capacity could greatly benefit intelligent systems in extensive real-world applications, *e.g.*, facilitating the moving decision in complex traffic or living environment in autonomous driving (Qi et al. 2019) or robotics (Fang et al. 2020; Follmann et al. 2019).

Recent years have witnessed promising progress in the amodal instance segmentation area. Former methods (Qi et al. 2019; Zhu et al. 2017; Follmann et al. 2019) directly infer both the visible and the amodal regions from images, while recent methods infer depth order information (Zhang et al. 2019; Zhan et al. 2020) or introduce prior information (Xiao et al. 2021) to help amodal instance segmentation. Despite the great progress of previous works, predicting amodal masks is still challenging because of lacking visible evidences for the occluded regions.

In practice, we found that the inferred amodal box and amodal mask are often incomplete, probably due to lacking evidences for expanding to complete amodal region. Recently, Xiao et al. (2021) employs several ground-truth (GT) amodal masks of training instances (*i.e.*, instances with GT annotations in the training set) similar to the initially estimated amodal mask as shape prior to benefit the mask refinement. We also consider exploiting prior information to support amodal inference. In contrast, we propose to exploit the prior information of expanding visible region to amodal region, based on which prior-guided expansion is performed for amodal instance segmentation. Specifically, the amodal mask is represented by a box and a mask within it (He et al. 2017; Xiao et al. 2021). Undersize box inherently limits the subsequent amodal mask prediction, which remains ignored in recent works (Xiao et al. 2021; Follmann et al. 2019). Thus, we perform box-level (*resp.*, pixel-level) expansion for amodal box (*resp.*, mask) prediction.

Following (Xiao et al. 2021; Follmann et al. 2019), our framework is built upon Mask R-CNN and further includes an expansion prior memory bank, a prior-guided amodal box head, and a prior-guided amodal mask head. The general pipeline firstly employs Mask R-CNN to infer the object class, original amodal box, and visible mask. After that, we search expansion prior for box in the bank to help the prior-guided amodal box head obtain the expanded amodal box, as shown in Fig. 1 (a). Finally, we search expansion prior for mask in the bank to assist the prior-guided amodal mask



(a) Box-level Expansion. (b) Pixel-level Expansion.

Figure 1: Overview of our method, which performs box-level expansion (*resp.,* pixel-level expansion) guided by the prior-based regression transformations (*resp.,* flow transformations). Visually, the amodal mask in (b) consists of both the visible part (in blue) and occluded part (in light blue).

*Corresponding author

head in obtaining the amodal mask, as shown in Fig. 1 (b).

Specifically, we construct one sub-bank for each class, where each slot in bank stores the information of a training instance. For each slot, the key is GT visible mask and the value is a tuple of GT visible box, GT amodal box, and GT amodal mask. We use the estimated visible mask to query the class-specific sub-bank, considering that objects having similar visible parts are likely to have similar amodal parts. Then, we can derive regression/flow transformations according to the retrieved values. Based on the expansion prior that how GT visible boxes are expanded to GT amodal boxes, we derive regression transformations to guide the box-level expansion. Based on the expansion prior that how the estimated visible mask is expanded to GT amodal masks, we derive flow transformations to guide the pixel-level expansion. Thanks to the expansion prior, our model can produce more complete amodal boxes and amodal masks.

We conduct extensive experiments on three datasets: KINS (Qi et al. 2019), D2SA (Follmann et al. 2019), and COCOA cls (Zhu et al. 2017). The in-depth analysis could demonstrate the effectiveness of our framework by performing box-level expansion and pixel-level expansion guided by expansion prior. Our contributions can be summarized as: 1) we propose a prior-guided expansion framework for amodal instance segmentation to address the incomplete box/mask issue; 2) technically, we propose to exploit prior-based regression (*resp.*, flow) transformations to facilitate box-level (*resp.*, pixel-level) expansion; 3) extensive experiments on three benchmark datasets indicate the effectiveness of our method against state-of-the-art baselines.

2 Related Work

2.1 Visual Occlusion Learning

In practice, occlusion is an inevitable problem and dramatically increases the learning difficulty, which has been researched in extensive applications including image classification (Kortylewski et al. 2021; Xiao et al. 2020), object detection (Wang et al. 2020; Chu et al. 2020), tracking (Yang et al. 2014), and segmentation (Gao, Packer, and Koller 2011; Winn and Shotton 2006). For example, BANet (Chen et al. 2020b) introduced an occlusion handling algorithm to model the occlusion between object instances for panoptic segmentation. Huang *et al.* (Huang et al. 2020) proposed to leverage the less occluded visible parts for effectively removing the redundant boxes in crowded pedestrian detection. Zhan *et al.* (Zhan et al. 2020) conducted ordering recovery, amodal completion, and content completion subtasks in self-supervised manner for scene de-occlusion task. Recently, Yuan *et al.* (Yuan et al. 2021) proposed a generative model of multiple objects to reason about multi-object occlusion under box-level supervision for the robust instance segmentation task. In this paper, we focus on the occlusion problem in amodal instance segmentation.

2.2 Amodal Instance Segmentation

The standard instance segmentation has achieved prominent progress in recent years (Ghiasi et al. 2021; Tian et al. 2019; Chen et al. 2020a; Xie et al. 2020) and has derived various richer tasks, including efficient instance segmentation (Zhang et al. 2020a; Lee and Park 2020), high-resolution instance segmentation (Wei et al. 2020; Su et al. 2020), and so on (Gupta, Dollar, and Girshick 2019; Roscher et al. 2020). In this paper, we focus on amodal instance segmentation (Li and Malik 2016), which considers the problem of segmenting instances with the occluded region.

Recently, amodal instance segmentation has drawn increasing research interest, probably due to its practical application in extensive complex scenes. The earliest work on amodal instance segmentation was proposed by (Li and Malik 2016), which iteratively predicts the amodal bounding box based on amodal segmentation heatmap and trains the model using occlusion data synthesized by overlapping cropped image patches. Afterwards, Zhu *et al.* (Zhu et al. 2017) employed SharpMask (Pinheiro, Collobert, and Dollár 2015) to predict the object amodal mask from coarse to fine. Qi *et al.* (Qi et al. 2019) proposed to ensemble the features from box and class head by multi-level coding for the occluded instances, which are determined by an occlusion classifier. SLN (Zhang et al. 2019) introduced a depth order representation to facilitate the inference of amodal mask. Considering the relationship between the visible region and amodal region, ORCNN (Follmann et al. 2019) proposed to further predict the occlusion mask by subtracting the visible mask from the amodal mask. BCNet (Ke, Tai, and Tang 2021) added an occlusion perception branch parallel to the traditional instance segmentation pipeline to consider the interactions between objects. Xiao *et al.* (Xiao et al. 2021) proposed to use cross-task attention together with GT training masks similar to coarse prediction for refinement. Considering that using memory bank of prior knowledge (Xiao et al. 2021; Tu et al. 2020; Zhang et al. 2020b) is also an effective method, we exploit expansion transformations from prior knowledge to perform both box-level and pixel-level expansion for better amodal inference.

3 Method

For the input image $\mathbf{I}$, amodal instance segmentation aims to infer the object class y and amodal mask $\mathbf{M}^a$ for each object instance. As illustrated in Fig 1 (b), the amodal mask consists of visible mask and occluded mask.

Our overall framework is shown in Fig 2, which mainly consists of four modules: Mask R-CNN, expansion prior memory bank, prior-guided amodal box head, and prior-guided amodal mask head. Firstly, Mask R-CNN produces N target instances from the input image, and the i-th instance has three estimations: object class y_i, original amodal box $\mathbf{B}_i^o$, and visible mask $\mathbf{M}_i^v$. Based on the estimated class and visible mask, the memory bank searches and provides regression transformations $\mathbf{T}_i^r$ for subsequent box-level expansion. After that, the visible box $\mathbf{B}_i^v$ is derived by clipping the visible mask, and the prior-guided amodal box head estimates the expanded amodal box $\mathbf{B}_i^a$ according to $\mathbf{B}_i^v$ and $\mathbf{T}_i^r$. Based on $\mathbf{B}_i^a$, the original region feature map and estimated visible mask are also expanded accordingly via resampling (*e.g.*, ROIAlign (He et al. 2017)). Then, the expanded visible mask is employed to query the bank to retrieve the flow

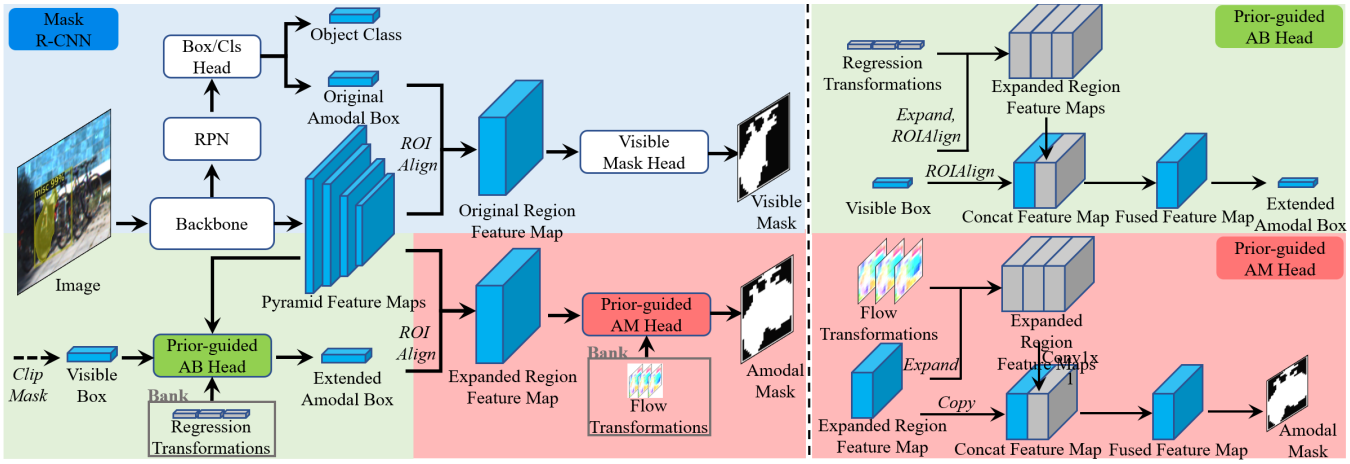


Figure 2: The framework of our method, mainly including Mask R-CNN (over blue background), prior-guided amodal box head (AB head, over green background), and prior-guided amodal mask head (AM head, over red background).

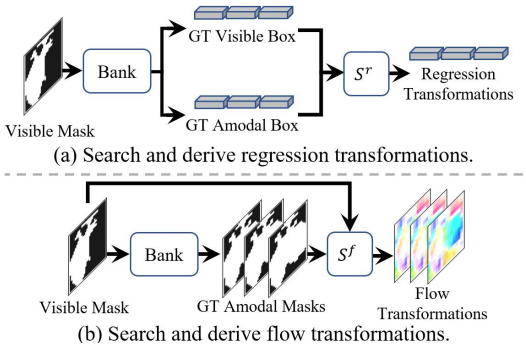


Figure 3: The procedure of searching and deriving regression transformations (a) and flow transformations (b), where the visible mask is employed to query and a solver is employed to derive transformations between targets.

transformations $\mathbf{T}_i^f$. Finally, the prior-guided amodal mask head predicts the amodal mask $\mathbf{M}_i^a$ on the expanded region feature map supported by $\mathbf{T}_i^f$.

For Mask R-CNN, we follow the setup in previous works (Xiao et al. 2021; Follmann et al. 2019), *e.g.*, predicting amodal box in its box head, using ResNet-50 (He et al. 2016) as backbone, and using channel size $C = 256$ and spatial size 14×14 for the region feature maps of mask heads. The training objective of Mask R-CNN is summarized as

$$\mathcal{L}_{mrcnn} = \mathcal{L}_{cls} + \mathcal{L}_{box}^o + \mathcal{L}_{mask}^v, \tag{1}$$

where $\mathcal{L}_{cls}$, $\mathcal{L}_{box}^o$, and $\mathcal{L}_{mask}^v$ are the training objectives for object class, original amodal box, and visible mask, consistent with these loss terms in (Xiao et al. 2021; Follmann et al. 2019; He et al. 2017). The architectures and procedures of other modules are introduced as follows.

3.1 Expansion Prior Memory Bank

The memory bank stores the expansion prior for box-level expansion and pixel-level expansion.

Construction The memory bank is constructed before the formal training according to training samples. For each class, we construct a sub-bank having K_s slots ($K_s = 1000$ in our experiments), with each slot corresponding to a training instance. The key of each slot is the visible mask of this instance, denoted as $\overline{\mathbf{M}}^v$. The value of each slot is a tuple of visible box $\overline{\mathbf{B}}^v$, amodal box $\overline{\mathbf{B}}^a$, and amodal mask $\overline{\mathbf{M}}^a$ of this instance. For the classes with fewer than K_s instances, we perform augmentation (*i.e.*, spatial transformation) to obtain K_s instances. For the classes with more than K_s instances, we perform K-Means and use K_s cluster centers.

Search For the i-th target instance, we employ its estimated visible mask $\mathbf{M}_i^v$ as query to search the sub-bank belonging to its predicted class, and find K ($K = 8$ in our experiments) nearest slots using the distance function $d(\mathbf{M}^{query}, \mathbf{M}^{key}) = \|E(\mathbf{M}^{query}) - E(\mathbf{M}^{key})\|_2$, where $E(\cdot)$ is a pre-trained encoder used in the distance computation (we reuse the mask encoder in (Xiao et al. 2021) and freeze it during training). After that, we can extract two types of expansion prior from the K retrieved values. Considering that the expansion prior is derived per value individually, we will take the k-th value as an example in the following description.

Deriving Regression Transformation To exploit expansion prior to guide the box-level expansion, we derive regression transformation according to the GT visible box $\overline{\mathbf{B}}_k^v$ and GT amodal box $\overline{\mathbf{B}}_k^a$, as shown in Fig. 3 (a). That is,

$$\mathbf{T}_k^r = S^r(\overline{\mathbf{B}}_k^v, \overline{\mathbf{B}}_k^a), \tag{2}$$

where the regression transformation $\mathbf{T}_k^r$ accounts for translating and scaling the box $\overline{\mathbf{B}}_k^v$ to match $\overline{\mathbf{B}}_k^a$. Function $S^r(\cdot,\cdot)$ is implemented based on the coordinates of two boxes as in (He et al. 2017; Ren et al. 2015). The details of $S^r(\cdot,\cdot)$ are trivial and omitted here.

Deriving Flow Transformation To exploit expansion prior

to guide the pixel-level expansion, we derive flow transformation according to the estimated visible mask $\mathbf{M}^v$ and GT amodal mask $\overline{\mathbf{M}}^a_k$, as shown in Fig. 3 (b). That is,

$$\mathbf{T}^f_k = S^f(\mathbf{M}^v, \overline{\mathbf{M}}^a_k), \tag{3}$$

where $\mathbf{T}^f_k$ is the derived flow transformation and $S^f(\cdot,\cdot)$ is a function solving the transformation from $\mathbf{M}^v$ to $\overline{\mathbf{M}}^a_k$, *i.e.*, 2-D spatial offsets for moving pixels in the visible mask to reconstruct the amodal mask. Unlike the close-form solver $S^r(\cdot,\cdot)$ for box, we have to employ a lightweight network (similar to FlowNet (Dosovitskiy et al. 2015)) as $S^f(\cdot,\cdot)$ to derive the flow transformations. $S^f(\cdot,\cdot)$ is pre-trained on paired GT visible masks and GT amodal masks of training instances and frozen during the formal training.

In the following, we introduce the details that how the prior regression (*resp.*, flow) transformations in memory bank guide the box-level (*resp.*, pixel-level) expansion.

3.2 Prior-guided Amodal Box Head

The original amodal box $\mathbf{B}^o$ predicted by Mask R-CNN is usually incomplete. Therefore, the prior-guided box head is proposed to perform box-level expansion guided by the searched regression transformations $\mathbf{T}^r$, as shown in the right-upper subfigure of Fig. 2.

Firstly, guided by regression transformations, we expand visible box $\mathbf{B}^v$ (derived via clipping visible mask $\mathbf{M}^v$) by

$$\mathbf{B}^p_k = \mathbf{T}^r_k(\mathbf{B}^v), \tag{4}$$

where $\mathbf{T}^r_k(\cdot)$ is the k-th regression transformation and $\mathbf{B}^p_k$ is the k-th prior-expanded box. Secondly, K prior-expanded boxes perform ROIAlign on the pyramid feature maps of Mask R-CNN to obtain K expanded region feature maps, in spatial size 4 considering computation complexity. After that, the channel size of concatenated expanded region feature map is squeezed from $K \times C$ to C. Meanwhile, the region feature map within visible box is obtained by ROIAlign. Then, the two feature maps are concatenated and fed into a 3×3 convolution layer outputting C channels. Finally, the flattened feature vector is fed into three fully-connected layers to predict the expanded amodal box $\mathbf{B}^a$. The loss term of this module *w.r.t* N target instances is

$$\mathcal{L}^a_{box} = \frac{1}{N}\sum_i^N \|\mathbf{B}^a_i - \mathbf{B}^{a*}_i\|_1, \tag{5}$$

where $\mathbf{B}^a_i$ is specifically a vector representing the 4 normalized coordinate (He et al. 2017; Ren et al. 2015) of the predicted amodal box for the i-th instance, and $\mathbf{B}^{a*}_i$ represents the corresponding ground-truth amodal box.

Overall, the expansion prior in the training instances is exploited as regression transformations and employed to expand visible box to facilitate the prediction of amodal box. In addition, we employ the expansion prior to expand the visible box derived by clipping visible mask, but it may be more intuitive to directly predict visible box by the box head in Mask R-CNN and then expand it. However, this intuitive manner degrades the performance of amodal box and amodal mask (see experiments in Sec. 4.4), probably because of lacking occlusion context which could have been exploited by backbone or region proposal network implicitly. Therefore, we predict amodal box in the box head of Mask R-CNN following (Xiao et al. 2021; Follmann et al. 2019), and obtain visible box by clipping visible mask.

3.3 Prior-guided Amodal Mask Head

Within the expanded amodal box $\mathbf{B}^a$, directly predicting the amodal mask $\mathbf{M}^a$ is still difficult. Therefore, the prior-guided amodal mask head is proposed to perform pixel-level expansion guided by the searched flow transformations $\mathbf{T}^f$, as shown in the bottom right subfigure of Fig. 2.

Firstly, guided by the flow transformations, we expand the feature map of visible region by

$$\mathbf{F}^p_k = \mathbf{T}^f_k(\mathbf{F}^e \cdot \mathbf{M}^{v,e}), \tag{6}$$

where $\cdot$ means dot-product, and $\mathbf{F}^e$ and $\mathbf{M}^{v,e}$ are the region feature map and estimated visible mask within the expanded amodal box respectively. $\mathbf{F}^p_k$ is the k-th expanded region feature map which spatially transforms visible region feature map via 2D offsets in $\mathbf{T}^f_k(\cdot)$. After that, the channel size of concatenated expanded region feature map is squeezed from $K \times C$ to C. The squeezed feature map is concatenated with $\mathbf{F}^e$ and then fed into a 1×1 convolution layer outputting C channels. Finally, 4 convolution layers, 1 deconvolution layer, and 1 convolution layer are employed to predict the amodal mask $\mathbf{M}^a$, following the mask head in (He et al. 2017; Xiao et al. 2021; Follmann et al. 2019). The loss term of this module *w.r.t* N instances could be formulated as

$$\mathcal{L}^a_{mask} = \frac{1}{N}\sum_i^N \mathcal{L}_{bce}(\mathbf{M}^a_i, \mathbf{M}^{a*}_i), \tag{7}$$

where $\mathbf{M}^{a*}_i$ is the associated ground-truth amodal mask of i-th target instance, and $\mathcal{L}_{bce}(\cdot,\cdot)$ is the binary cross-entropy loss used in (He et al. 2017; Xiao et al. 2021).

Overall, the expansion prior in the training instances is exploited as flow transformations, which are used to expand the feature map of visible region to facilitate the prediction of amodal mask. Compared with the directly concatenated GT amodal masks in (Xiao et al. 2021), the concatenated expanded region feature maps could implicitly encode more structural and contextual information, and thus better benefit the amodal inference (see experiments in Sec. 4.4).

3.4 The Total Training Objective

Overall, our total training objective can be formulated as

$$\mathcal{L} = \mathcal{L}_{mrcnn} + \mathcal{L}^a_{box} + \mathcal{L}^a_{mask}, \tag{8}$$

where $\mathcal{L}_{mrcnn}$, $\mathcal{L}^a_{box}$ and $\mathcal{L}^a_{mask}$ are the training objectives of Mask R-CNN in Sec. 3, prior-guided amodal box head in Sec. 3.2, and prior-guided amodal mask head in Sec. 3.3.

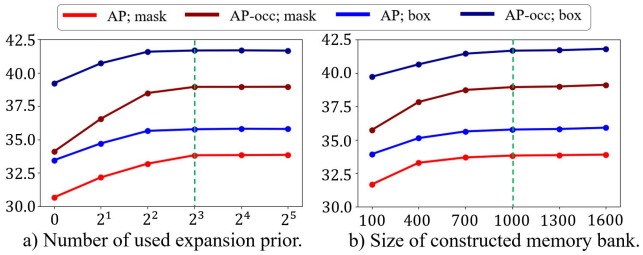
4 Experiments

4.1 Datasets and Implementation Details

As in (Xiao et al. 2021), we investigate the performance of our method on three public datasets: the KINS dataset (Qi

Set	AB Head		AM Head		Amodal Mask		Amodal Box	
	w/o	with	w/o	with	AP	AP-occ	AP	AP-occ
#1			✓		30.12	33.61	32.54	38.13
#2	✓		✓		30.66	34.10	33.45	39.23
#3		✓	✓		31.95	36.76	35.63	41.63
#4				✓	31.25	34.97	32.59	38.24
#5	✓			✓	31.87	36.32	33.48	39.34
#6		✓		✓	33.82	38.95	35.77	41.68

Table 1: Module contributions on KINS dataset. "AB head" (*resp.*, "AM head") is the abbreviation for amodal box (*resp.*, mask) head, while "with" and "w/o" indicate whether to enable the expansion prior or not.



a) Number of used expansion prior. b) Size of constructed memory bank.

Figure 4: The performances of amodal mask and amodal box *w.r.t* various numbers of used expansion prior. The green dotted lines indicate the default values.

et al. 2019), the D2SA (D2S amodal) dataset (Follmann et al. 2019), and the COCOA cls dataset (Zhu et al. 2017). We implement the proposed method on the codebase of previous work (Xiao et al. 2021), which builds on Detectron2 using Python 3.7 and PyTorch 1.4.0 framework. We conducted the experiments on Ubuntu 18.04 system with 32 GB Intel 9700K CPU and two NVIDIA 1080ti GPU cards.

4.2 Evaluation

We employ the mean average precision (AP) for the evaluation following previous works (Xiao et al. 2021; Zhu et al. 2017), and we evaluate the performance for both amodal box and amodal mask to investigate the effectiveness of box-level expansion and pixel-level expansion. We also follow (Xiao et al. 2021) to focus on the performance of occluded instances via AP-occ, which only computes the performance on the instances having visible rate (*i.e.*, IoU between visible mask and amodal mask) not larger than 85%. We employ the evaluation API in (Xiao et al. 2021) for fair comparisons, which inherits the API of COCO dataset (Lin et al. 2014).

4.3 Ablation Study

We conduct ablation study on the KINS dataset, considering that it is the largest real-world dataset for amodal instance segmentation. We investigate the performances of various combination sets of modules, and summarize the results in Tab. 1. The most basic set is Set#1, which totally obsoletes the prior-guided amodal box head and directly predicts amodal mask within the original amodal box (*i.e.*, Mask R-CNN with additional amodal mask head). Firstly, simply adding a box head without prior (*i.e.*, Set#1 *v.s.* Set#2) just slightly improves the performances of mask and box. Solely utilizing the prior-guided amodal box head (*i.e.*, Set#1 *v.s.* Set#3) dramatically promotes the performance of box and thus benefits the downstream amodal mask prediction. While solely utilizing the prior-guided amodal mask head (*i.e.*, Set#1 *v.s.* Set#4) dramatically promotes the performance of amodal mask. Finally, the full-fledged combination of prior-guided amodal box head and mask head (*i.e.*, Set#6) achieves the optimal performance indicating the effectiveness of prior-guided expansion.

Method	Amodal Mask				Amodal Box	
	AP	AP-50	AP-75	AP-occ	AP	AP-occ
MRCNN	30.01*	54.53*	30.11*	-	32.50	-
MRCNN^8	30.71*	54.36*	31.47*	-	32.57	-
ORCNN	30.64*	54.21*	31.29*	34.23*	32.65	38.58
Qi et al.	32.20*	55.45	33.21	37.47	33.40*	39.42
BCNet	31.61	55.02	32.86	36.72	32.66	38.71
Xiao et al.	32.08*	55.37*	33.34*	37.40*	32.70	39.00
PGExp	33.82	55.54	35.66	38.95	35.77	41.68

(a) Results on KINS dataset.

Method	Amodal Mask				Amodal Box	
	AP	AP-50	AP-75	AP-occ	AP	AP-occ
MRCNN	63.57*	83.85*	68.02*	-	64.01	-
MRCNN^8	64.85*	84.05*	70.72*	-	64.20	-
ORCNN	64.22*	83.55*	69.12*	45.27*	64.45	53.83
Qi et al.	66.67	84.04	72.56	47.66	65.03	53.95
BCNet	67.41	84.62	73.34	48.54	64.78	53.92
Xiao et al.	70.27*	85.11*	75.81*	51.17*	64.91	54.04
PGExp	71.79	85.23	76.77	53.75	71.23	55.38

(b) Results on D2SA dataset.

Method	Amodal Mask				Amodal Box	
	AP	AP-50	AP-75	AP-occ	AP	AP-occ
MRCNN	33.67*	56.50*	35.78*	-	39.41	-
MRCNN^8	34.72*	57.50*	36.93*	-	39.56	-
ORCNN	28.03*	53.68*	25.36*	17.40*	39.47	25.15
Qi et al.	34.82	56.12	37.31	19.22	39.74	25.32
BCNet	35.02	56.08	37.54	19.92	39.64	25.38
Xiao et al.	35.41*	56.03*	38.67*	22.17*	39.76	25.44
PGExp	37.55	57.74	41.41	23.31	43.01	26.31

(c) Results on COCOA dataset.

Table 2: The comparison of various methods on three benchmark datasets.

4.4 Expansion Prior Analysis

For the number of used expansion prior, we summarize the results in Fig. 4 (a) to show the effects. Guided by only 2 expansion prior, the performances of amodal box and amodal box both are obviously improved. The performances of box are gradually saturated around 4, while the performances of mask are gradually saturated around 8. Considering the uniformity, we employ $K = 8$ for both amodal box and amodal mask in Sec. 3.1. For the size of constructed memory bank, we summarize the results in Fig. 4 (b) to show the impacts. The overall performance is relatively saturated around the default value (*i.e.*, 1000). In addition, large bank size is still efficient relative to the dominated backbone, and the default value is a reasonable choice.

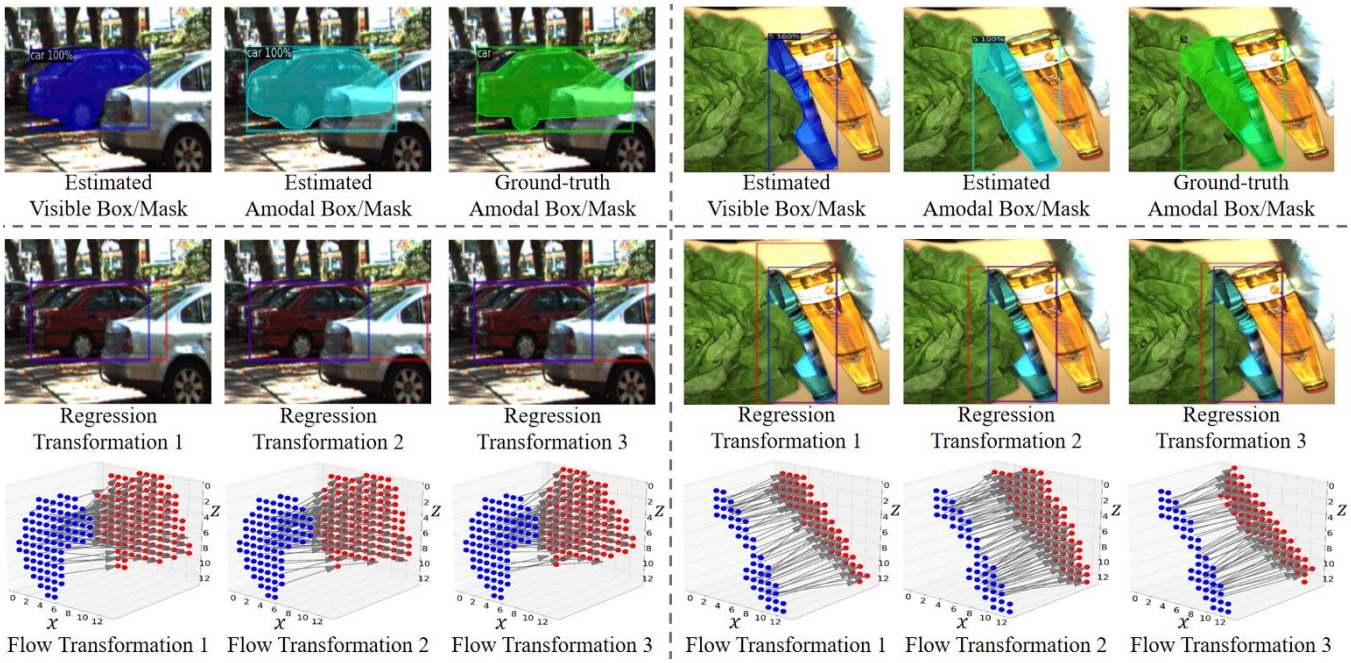


Figure 5: Visualizations for the expansion prior on KINS (left) and D2SA (right) datasets. The first row shows the estimations and ground-truth. The second row shows the top-3 (nearest in retrieving distance) regression transformations, in which the blue box indicates the visible box and the red box indicates the prior-expanded box. The third row shows the top-3 flow transformations, in which the blue points represent the estimated visible mask, the red points represent the searched GT amodal mask, and the grey arrows show the process of expanding visible region.

4.5 Qualitative Analysis

We visualize the regression transformations and flow transformations to better understand the process of box-level expansion and pixel-level expansion, as shown in Fig. 5. The regression transformations serve as prior knowledge indicating the expansion directions and scopes, and results in a more complete expanded amodal box. Within the expanded amodal box (zoomed to spatial size 14×14 via ROIAlign), the flow transformations serve as prior knowledge to guide the expanding of visible region feature map. Thanks to the expansion prior of regression transformations and flow transformations, our model is guided to produce more complete results.

4.6 Comparison with Previous Works

Comparative Baselines We compare our method (dubbed as PGExp) with the following state-of-the-art methods: 1) MRCNN (He et al. 2017), which uses the network architecture of Mask R-CNN to predict amodal box and amodal mask. 2) MRCNN^8, which is a deeper Mask R-CNN used in (Xiao et al. 2021). 3) ORCNN (Follmann et al. 2019), which parallelly predicts visible mask and amodal mask, and further predicts the occlusion mask by subtracting visible mask from amodal mask to model the relationship between them. 4) Qi et al. (2019), which estimates occluded parts by multi-task framework with multi-level coding. 5) BCNet (Ke, Tai, and Tang 2021), which employs bilayer structure to consider the interaction between occluding and occluded instances. 6) Xiao et al. (2021), which parallelly predicts coarse visible mask and amodal mask and employs cross-task attention and concatenating searched GT masks for refinement.

Results and Analysis All the results are summarized in Tab. 2, where the values marked by $*$ are directly copied from (Xiao et al. 2021) or the corresponding paper and the values without $*$ are obtained via reproductions with comparable and fair implementation. Firstly, simply employing a deeper mask head only slightly improves the mask performances, *e.g.*, 30.71 *v.s.* 30.01 *AP* on KINS dataset. Our method outperforms all baselines by a large margin in terms of amodal box (*e.g.*, 35.77 *v.s.* 33.40 *AP* on KINS dataset) by prior-guided box-level expansion, which also enables producing more complete amodal mask in the downstream process. Our method also shows superior performance against all baselines dramatically in term of amodal mask (*e.g.*, 33.82 *v.s.* 32.20 *AP* on KINS dataset), thanks to the prior-guided pixel-level expansion. In addition, the improvement in amodal box prediction of our method leads to generally less improvement in downstream amodal mask prediction, which may due to the neglecting of amodal box in previous works. We also conjecture that mask prediction is notably more difficult than box prediction, the improvement on box would be inevitably compromised on the downstream segmentation. Overall, our model achieves the optimal performances for both amodal box and mask, demonstrating the effectiveness of our prior-guided expansion framework.

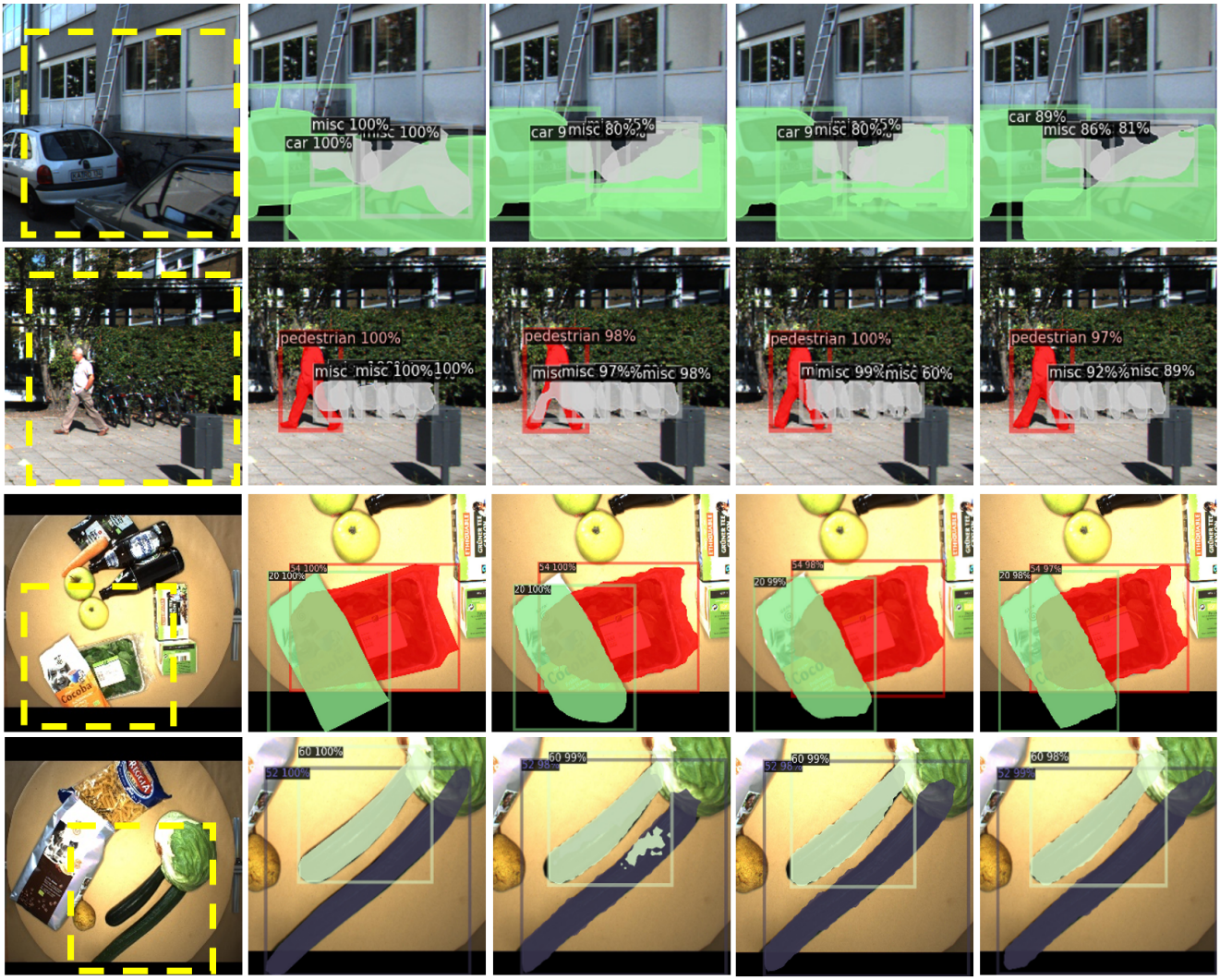


Figure 6: Qualitative comparison with competitive baselines. The column a) shows the source images, where the yellow dotted boxes indicate the visualized regions zoomed for clearer comparison. The column b-e) show the class, score, amodal box, and amodal mask for each target instance of ground-truth, BCNet (Ke, Tai, and Tang 2021), Xiao *et al* (Xiao et al. 2021), and ours.

4.7 Qualitative Comparison

We conduct the qualitative comparison with the two representative baselines, *i.e.*, BCNet (Ke, Tai, and Tang 2021) and Xiao *et al.* (Xiao et al. 2021). As shown in Fig. 6, our method could predict more complete amodal boxes and amodal masks, by performing prior-guided box-level expansion and prior-guided pixel-level expansion. For example, in the complex scene of the first row (*i.e.*, the two bicycles are occluded by cars), BCNet and Xiao *et al.* both fail to produce complete amodal boxes and amodal masks, and our method could estimate more precise results by virtue of the expansion priors. In the last row, for the cucumber occluded by the cabbage and border padding, our model could produce more favorable amodal mask, especially for the two ends.

5 Conclusion

In this paper, we have developed a prior-guided expansion framework for amodal instance segmentation. Specifically, we exploit expansion prior from training instances and derive regression (*resp.*, flow) transformations to facilitate box-level (*resp.*, pixel-level) expansion. Extensive experiments and in-depth analyses on KINS, D2SA, and COCOA cls datasets have demonstrated the effectiveness of our proposed framework for amodal instance segmentation.

Acknowledgements

The work was supported by the National Science Foundation of China (62076162), and the Shanghai Municipal Science and Technology Major/Key Project, China (2021SHZDZX0102, 20511100300).

References

Chen, H.; Sun, K.; Tian, Z.; Shen, C.; Huang, Y.; and Yan, Y. 2020a. BlendMask: Top-down meets bottom-up for instance segmentation. In *CVPR*.

Chen, Y.; Lin, G.; Li, S.; Bourahla, O.; Wu, Y.; Wang, F.; Feng, J.; Xu, M.; and Li, X. 2020b. BANet: Bidirectional aggregation network with occlusion handling for panoptic segmentation. In *CVPR*.

Chu, X.; Zheng, A.; Zhang, X.; and Sun, J. 2020. Detection in crowded scenes: One proposal, multiple predictions. In *CVPR*.

Dosovitskiy, A.; Fischer, P.; Ilg, E.; Hausser, P.; Hazirbas, C.; Golkov, V.; Van Der Smagt, P.; Cremers, D.; and Brox, T. 2015. FlowNet: Learning optical flow with convolutional networks. In *ICCV*.

Fang, Z.; Jain, A.; Sarch, G.; Harley, A. W.; and Fragkiadaki, K. 2020. Move to See Better: Self-Improving Embodied Object Detection. *arXiv preprint arXiv:2012.00057*.

Follmann, P.; König, R.; Härtinger, P.; Klostermann, M.; and Böttger, T. 2019. Learning to see the invisible: End-to-end trainable amodal instance segmentation. In *WACV*.

Gao, T.; Packer, B.; and Koller, D. 2011. A segmentation-aware object detection model with occlusion handling. In *CVPR*.

Ghiasi, G.; Cui, Y.; Srinivas, A.; Qian, R.; Lin, T.-Y.; Cubuk, E. D.; Le, Q. V.; and Zoph, B. 2021. Simple copy-paste is a strong data augmentation method for instance segmentation. In *CVPR*.

Gupta, A.; Dollar, P.; and Girshick, R. 2019. LVIS: A dataset for large vocabulary instance segmentation. In *CVPR*.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask R-CNN. In *ICCV*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*.

Huang, X.; Ge, Z.; Jie, Z.; and Yoshie, O. 2020. NMS by representative region: Towards crowded pedestrian detection by proposal pairing. In *CVPR*.

Ke, L.; Tai, Y.-W.; and Tang, C.-K. 2021. Deep Occlusion-Aware Instance Segmentation with Overlapping BiLayers. In *CVPR*.

Kortylewski, A.; Liu, Q.; Wang, A.; Sun, Y.; and Yuille, A. 2021. Compositional convolutional neural networks: A robust and interpretable model for object recognition under occlusion. *International Journal of Computer Vision*, 129(3): 736–760.

Lee, Y.; and Park, J. 2020. Centermask: Real-time anchor-free instance segmentation. In *CVPR*.

Li, K.; and Malik, J. 2016. Amodal instance segmentation. In *ECCV*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *ECCV*.

Pinheiro, P. O.; Collobert, R.; and Dollár, P. 2015. Learning to segment object candidates. In *NeurIPS*.

Qi, L.; Jiang, L.; Liu, S.; Shen, X.; and Jia, J. 2019. Amodal instance segmentation with KINS dataset. In *CVPR*.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster R-CNN: Towards real-time object detection with region proposal networks. *NeurIPS*.

Roscher, R.; Volpi, M.; Mallet, C.; Drees, L.; and Wegner, J. D. 2020. SemCity Toulouse: A benchmark for building instance segmentation in satellite images. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 5: 109–116.

Su, H.; Wei, S.; Liu, S.; Liang, J.; Wang, C.; Shi, J.; and Zhang, X. 2020. HQ-ISNet: High-quality instance segmentation for remote sensing imagery. *Remote Sensing*, 12(6): 989.

Tian, Z.; Shen, C.; Chen, H.; and He, T. 2019. FCOS: Fully convolutional one-stage object detection. In *ICCV*.

Tu, Y.; Niu, L.; Chen, J.; Cheng, D.; and Zhang, L. 2020. Learning from web data with self-organizing memory module. In *CVPR*.

Wang, A.; Sun, Y.; Kortylewski, A.; and Yuille, A. L. 2020. Robust object detection under occlusion with context-aware CompositionalNets. In *CVPR*.

Wei, S.; Zeng, X.; Qu, Q.; Wang, M.; Su, H.; and Shi, J. 2020. HRSID: A high-resolution SAR images dataset for ship detection and instance segmentation. *IEEE Access*, 8: 120234–120254.

Winn, J.; and Shotton, J. 2006. The layout consistent random field for recognizing and segmenting partially occluded objects. In *CVPR*.

Xiao, M.; Kortylewski, A.; Wu, R.; Qiao, S.; Shen, W.; and Yuille, A. 2020. TDMPNet: Prototype network with recurrent top-down modulation for robust object classification under partial occlusion. In *ECCV*.

Xiao, Y.; Xu, Y.; Zhong, Z.; Luo, W.; Li, J.; and Gao, S. 2021. Amodal Segmentation Based on Visible Region Segmentation and Shape Prior. *AAAI*.

Xie, E.; Sun, P.; Song, X.; Wang, W.; Liu, X.; Liang, D.; Shen, C.; and Luo, P. 2020. Polarmask: Single shot instance segmentation with polar representation. In *CVPR*.

Yang, M.; Liu, Y.; Wen, L.; You, Z.; and Li, S. Z. 2014. A probabilistic framework for multitarget tracking with mutual occlusions. In *CVPR*.

Yuan, X.; Kortylewski, A.; Sun, Y.; and Yuille, A. 2021. Robust Instance Segmentation through Reasoning about Multi-Object Occlusion. In *CVPR*.

Zhan, X.; Pan, X.; Dai, B.; Liu, Z.; Lin, D.; and Loy, C. C. 2020. Self-supervised scene de-occlusion. In *CVPR*.

Zhang, R.; Tian, Z.; Shen, C.; You, M.; and Yan, Y. 2020a. Mask encoding for single shot instance segmentation. In *CVPR*.

Zhang, Y.; Niu, L.; Pan, Z.; Luo, M.; Zhang, J.; Cheng, D.; and Zhang, L. 2020b. Exploiting motion information from unlabeled videos for static image action recognition. In *AAAI*.

Zhang, Z.; Chen, A.; Xie, L.; Yu, J.; and Gao, S. 2019. Learning semantics-aware distance map with semantics layering network for amodal instance segmentation. In *ACM MM*.

Zhu, Y.; Tian, Y.; Metaxas, D.; and Dollár, P. 2017. Semantic amodal segmentation. In *CVPR*.