AISFormer: Amodal Instance Segmentation with Transformer

Minh Tran[1]
minht@uark.edu
Khoa Vo[1]
khoavoho@uark.edu
Kashu Yamazaki[1]
kyamazak@uark.edu
Arthur Fernandes[2]
Arthur.Fernandes@cobbvantress.com
Michael Kidd[3]
mkidd@uark.edu
Ngan Le[1]
thile@uark.edu

[1] Department of CSCE, University of Arkansas
[2] Cobb-Vantress, Inc
[3] Poultry Science, University of Arkansas

Abstract

Amodal Instance Segmentation (AIS) aims to segment the region of both visible and possible occluded parts of an object instance. While Mask R-CNN-based AIS approaches have shown promising results, they are unable to model high-level features coherence due to the limited receptive field. The most recent transformer-based models show impressive performance on vision tasks, even better than Convolution Neural Networks (CNN). In this work, we present AISFormer, an AIS framework, with a Transformer-based mask head. AISFormer explicitly models the complex coherence between occluder, visible, amodal, and invisible masks within an object's regions of interest by treating them as learnable queries. Specifically, AISFormer contains four modules: (i) feature encoding: extract ROI and learn both short-range and long-range visual features. (ii) mask transformer decoding: generate the occluder, visible, and amodal mask query embeddings by a transformer decoder (iii) invisible mask embedding: model the coherence between the amodal and visible masks, and (iv) mask predicting: estimate output masks including occluder, visible, amodal and invisible. We conduct extensive experiments and ablation studies on three challenging benchmarks i.e. KINS, D2SA, and COCOA-cls to evaluate the effectiveness of AISFormer. The code is available at: https://github.com/UARK-AICV/AISFormer

1 Introduction

Instance segmentation (IS) is a fundamental task in computer vision, which predicts each object instance and its corresponding per-pixel segmentation mask. Modern IS approaches [18, 25, 42, 45] are typically built on CNN and follow Mask R-CNN [19] paradigm with

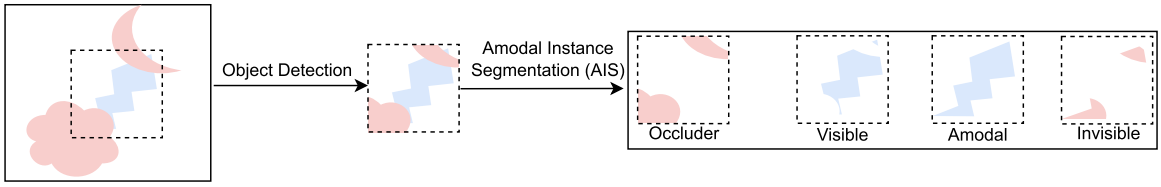

Figure 1: An explanation of different mask instances in Amodal Instance Segmentation (AIS). Given a region of interest (ROI) extracted by an object detector, AIS aims to extract both visible and invisible mask instances including occluder, visible, amodal, and invisible.

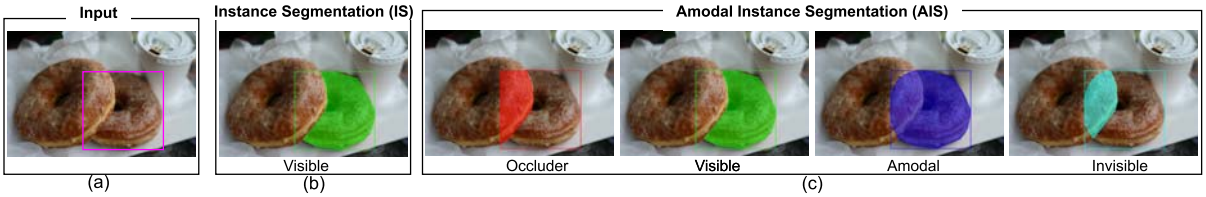

Figure 2: A comparison between Instance Segmentation (IS) and Amodal Instance Segmentation (AIS). Given an image with ROI (a), IS aims to extract the visible mask instance (b) whereas AIS aims to extract both the visible mask and occluded parts (c).

two stages of first detecting bounding boxes, and then segmenting instance masks. Mask R-CNN has been improved by various backbone architecture designs to obtain high accuracy IS performance. Despite demonstrating impressive performance, those Mask R-CNN-based approaches are limited when dealing with occlusion, in which a lot of segmentation errors happen due to overlapping objects, especially among object instances belonging to the same class [23]. The capability of perceiving/segmenting the entirety of an object instance regardless of partial occlusion is known as amodal instance perception/segmentation (AIS). In AIS, there are various mask instances taken into consideration including occluder, visible, amodal, and invisible as shown in Fig.1.

A comparison between IS and AIS is given in Fig.2. While IS aims to segment the visible region of the donut, as shown in the middle panel, AIS aims to extract both visible and occluded regions of the donut as shown in the right panel of Fig.2. Humans are capable of amodal perception, but computers are limited to modal perception [37], which has played an important role in theoretical study and real-world applications. In the literature, various AIS approaches [16, 23, 26, 32, 41] have been based on the IS pipeline of Mask R-CNN [19] and attempted to predict the amodal mask of objects given the pooling region of interest (ROI) feature. In such networks, the main contribution comes from segmentation head designs, with attention paid to the amodal instance mask regression after obtaining the ROI. For example, [16] performs AIS through an amodal branch and invisible mask loss, [32] reasons invisible parts via a multi-task framework with Multi-Level Coding (MLC), [40] proposes a cross-task attention based refinement strategy to refine the amodal mask and the visible mask, [23] introduces a bilayer decoupling layer extract to occluder and infer occludee. The Mask R-CNN-based AIS approaches have shown promising AIS results, however, they are unable to model high-level or long-range features coherence due to the limited receptive field. That means the complex relationship between ROIs has been not modeled in such Mask R-CNN-based AIS approaches.

Recently, the DETR[6]-based approaches [12, 14, 20] have shown their advantage and achieved on-par performances compared with CNN-based ones on object detection and instance segmentation. Indeed, they model the long-range relationships between objects across visual features by reformulating objects as learnable query embeddings and take advantage

of the transformer attention mechanism. Inspired by that advantage, we propose AISFormer, an AIS framework, with a Transformer-based mask head. Our AISFormer consists of four modules: feature encoding, mask transformer decoder, invisible mask embedding, and segmentation. In the first module, after obtaining the region of interest (ROI) feature from the backbone and ROIAlign algorithm, CNN-based layers and a transformer encoder are applied to learn both short-range and long-range features of the given ROI. The second module aims to generate the occluder, visible and amodal mask query embeddings by a transformer decoder. The next module extracts invisible mask embeddings to model the coherence between the amodal mask and visible mask. The last module, the segmentation module, estimates the output masks including occluder, visible, amodal, and invisible. Our overall network is shown in Fig.3. Our contribution can be summarized as follows:

- We propose AISFormer, an amodal instance segmentation framework, with a Transformer-based mask head. Our AISFormer can explicitly model the complex coherence between occluder, visible, amodal, and invisible masks within an object's regions of interest by treating them as learnable queries. AISFormer also models the relationship between these embeddings and the corresponding region of interest.
- We empirically validate the usefulness of our proposed method by showing that it achieves superior performance to most of the current state-of-the-art methods benchmarked on three amodal datasets, i.e., KINS [32], COCOA-cls [46], and D2SA [15].

2 Related Work

2.1 Instance Segmentation (IS)

The existing IS approaches can be summarized into two categories of two-stage and one-stage. The former category, two-stage IS approaches, is divided into two groups: top-down and bottom-up methods based on the priority of detection and segmentation. The top-down two-stage approaches [5, 8, 10, 11] follow the pipeline of Mask R-CNN [19] by first detecting bounding boxes and then performing segmentation in each RoI region. The bottom-up two-stage approaches [3, 9, 30] first generate semantic segmentation and then group pixels through clustering or metric learning techniques. While the two-stage category has not taken the correlation between detection and segmentation into consideration, the latter category, single-stage, addresses this limitation with less time consumption [4]. There are two common groups in the single-stage category i.e., anchor-based and anchor-free methods. The anchor-based one-stage approaches [4, 10, 27] simultaneously generate a set of class-agnostic candidate masks on the candidate region and extract instances from a semantic branch. To cope with the drawback of anchor-based approaches that rely heavily on predefined anchors sensitive to hyper-parameters, anchor-free one-stage approaches [7, 25, 43] are proposed by eliminating anchor boxes and using corner/center points.

2.2 Amodal Instance Segmentation (AIS)

Unlike IS, which only focuses on visible regions, AIS predicts both the visible and amodal masks of each object instance. One of the first works in AIS is proposed by Li *et al.* [26], which relies on the directions of high heatmap values computed for each object to iteratively enlarge the corresponding object modal bounding box. Generally, AIS can be categorized into two groups of weakly supervised and fully supervised methods.

Fully supervised AIS: The majority of AIS approaches follow a fully supervised approach, where the amodal annotation is required. The amodal annotation can come from human

labeling [16, 32, 40] or synthetic occlusions [44]. Follmann *et al.* [16] propose ORCNN, in which they have two main instances mask heads, amodal and inmodal. In addition, they build another mask head on top of these two for invisible mask prediction. Based on ORCNN architecture, Qi *et al.* [32] introduce ASN, where they propose the multi-level coding module. The module aims to learn global information, thus obtaining better segmentation for the invisible part. Different from ORCNN and ASN, VQ-VAE [21] replaces the fully convolutional instance mask heads with variational autoencoders. Specifically, the input feature is firstly classified into an intermediate shape code. Then, the shape code is reconstructed into a complete object shape. Xiao *et al.* [41] also uses a pre-trained autoencoder to refine the initial mask prediction from the amodal mask head. The pre-trained autoencoder encodes the mask to a shape prior memory codebook and decodes it into a refined amodal mask. More recently, BCNet[23] attempts to decouple the mixing features by extending another branch that predicts the mask of occluders inside the bounding box of the occludee. Thus, the model can be aware of occluders and more precisely predict the amodal mask.

Weakly supervised AIS (WAIS): [31, 44] consider this problem under weak supervision where they attempt to generate amodal mask ground-truths from datasets using the available visible ground truths. The reason is that human annotators sometimes could not provide reliable ground truth on amodal masks. The procedure of [31, 44] on WAIS can be described as follows. First, data augmentation is applied to generate occlusion on top of objects in the dataset. Then, a UNet model [34] is trained as a segmenter. The input of the UNet is the available visible mask ground truth of the object. However, with the occlusion augmentation added to the object, the visible ground truth is treated as the amodal ground truth. In the second training stage, the amodal segmentations outputs from the trained UNet in the first phase are taken as a pseudo-ground truth for learning a standard instance segmenter – Mask R-CNN [24]. In the inference phase, Mask R-CNN trained on the generated amodal ground truth is expected to yield the correct AIS. Another approach in WAIS uses object bounding boxes as inputs [35, 36], where shape priors and probabilistic models are applied to classify visible, invisible, and background parts within the bounding box.

Our ASIFormer belongs to the first group of fully supervised learning, in which visible, amodal, and occluder masks are represented by queries in a unified manner.

2.3 Query-based Image Segmentation

Inspired by DETR [6], query-based IS methods [12, 14, 20] have achieved impressive IS performance by treating segmentation as a set prediction problem. In these methods, queries are used to represent the objects and jointly learn multiple tasks of classification, detection, and segmentation. In [12], each query represents one object with multiple representations of class, location, and mask. For the IS branch, the mask encoding vectors are learnt by the compression coding (DCT [2], PCA [1], Sparse Coding [13]) of raw spatial masks. Instead of high-dimensional masks, ISTR [20] predicts low-dimensional mask embeddings and is trained by a set loss based on bipartite matching. Meanwhile, QueryInst [14] addresses IS problem by using parallel dynamic mask heads [22]. In QueryInst, queries are shared between tasks of detection and segmentation in each stage via dynamic convolutions. Unlike the previous works which process on regular dense tensors, Mask Tranfiner [24] first decomposes and represents an image region as a hierarchical quadtree. Then, all points on the quadtree are transformed into a query sequence to predict labels. By doing that, Mask Tranfiner only processes detected error-prone tree nodes, which are mostly along object boundaries or in high-frequency regions, and self-corrects their errors in parallel. Similar to

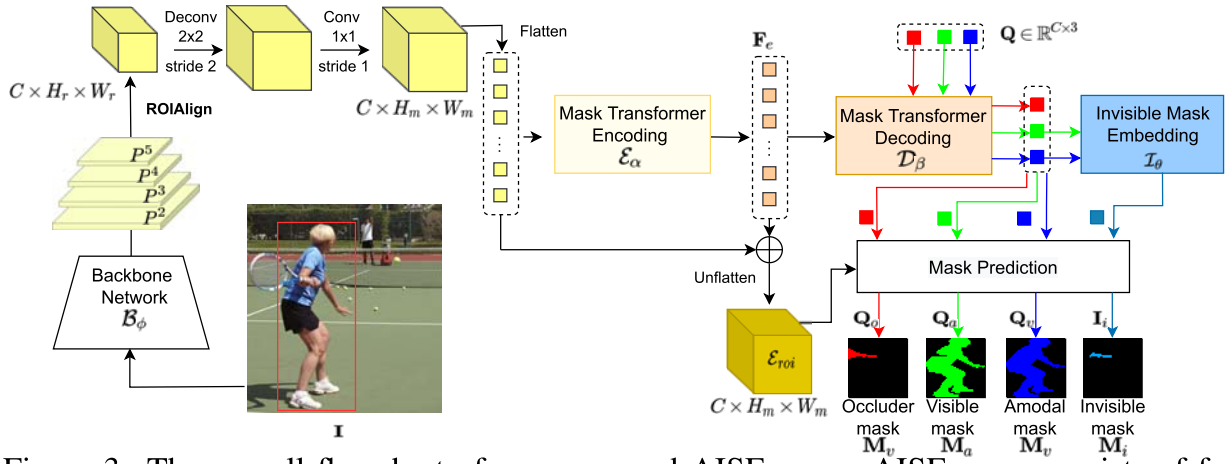

Figure 3: The overall flowchart of our proposed AISFormer. AISFormer consists of four modules corresponding to (i) feature encoding: after obtaining the region of interest (ROI) feature from the backbone $\mathcal{B}_\phi$ and ROIAlign algorithm φ, CNN-based layers and a transformer encoder are applied to learn both short-range and long-range features of the given ROI.(ii) mask transformer decoding $\mathcal{D}_\beta$: generate the occluder, visible, and amodal mask query embeddings by a transformer decoder (iii) invisible mask embedding $\mathcal{I}_\theta$ to model the coherence between the amodal and visible masks by computing the invisible mask embedding, and (iv) segmentation to estimate output masks including occluder ($\mathbf{M}_o$), visible ($\mathbf{M}_v$), amodal ($\mathbf{M}_a$)and invisible ($\mathbf{M}_i$).

[12, 14, 20], our proposed AISFormer is inspired by DETR. Indeed, our learnable queries represent visible, amodal, and occluder masks. Our transformer-based approach aims to model the coherence between them.

3 Methods

We propose an approach, termed AISFormer, to efficiently tackle AIS. The overall architecture of AISFormer is illustrated in Fig 3. Our AISFormer consists of four modules corresponding to (i) feature encoding: in this first module, after obtaining the region of interest (ROI) feature from the backbone $\mathcal{B}_\phi$ and ROIAlign algorithm φ, CNN-based layers and a transformer encoder are applied to learn both short-range and long-range features of the given ROI. (ii) mask transformer decoding $\mathcal{D}_\beta$: generate the occluder, visible, and amodal mask query embeddings by a transformer decoder (iii) invisible mask embedding $\mathcal{I}_\theta$ to model the coherence between the amodal and visible masks by computing the invisible mask embedding, and (iv) segmentation to estimate output masks including occluder, visible, amodal and invisible.

Generally, given an image $\mathbf{I}$, AIS aims to predict the amodal mask $\mathbf{M}_a$ and visible mask $\mathbf{M}_v$ within ROIs. To better model the mask coherence between amodal mask $\mathbf{M}_a$ and visible mask $\mathbf{M}_v$, our AISFormer estimates other relevant masks such as occluder mask $\mathbf{M}_o$ and invisible mask $\mathbf{M}_i$. In such relationships, the amodal mask $\mathbf{M}_a$ consists of the visible part $\mathbf{M}_v$ and the invisible mask $\mathbf{M}_i$, which is a part of occluder mask $\mathbf{M}_o$.

3.1 Feature Encoding

Given an image $\mathbf{I}$, we first adopt a pre-trained image recognition network $\mathcal{B}_\phi$ (e.g. ResNet [19], RegNet[33]) to extract the spatial visual features, where ϕ is backbone network weights. After backbone feature extraction, we apply ROIAlign algorithm φ to obtain ROIs corresponding to the bounding boxes in the image $\mathbf{F}_{roi} = \varphi(\mathcal{B}_\phi(\mathbf{I}))$, where $\mathbf{F}_{roi} \in \mathbb{R}^{C \times H_r \times W_r}$. To

represent $\mathbf{F}_{roi}$ in higher resolution without increasing computational cost, we apply a deconvolution layer with a stride of 2 into the spatial domain. As a result, the feature maps are upsampled, i.e., $\mathbf{F}_{roi} \in \mathbb{R}^{C\times H_m \times W_m}$, whereas $H_m = 2 \times H_r$ and $W_m = 2 \times W_r$. The higher-resolution feature maps $\mathbf{F}_{roi}$ is passed through a 1×1 stride 1 convolutional layer and then flattened from the spatial dimensions into one dimension and obtain $H_r W_r$ embeddings with the dimension of C. The flattened feature maps with their positional encodings [38] are fed into the mask transformer encoder $\mathcal{E}_\alpha$ to extract long-range dependency across spatial domains, where α is transformer encoder network weights. The mask transformer encoder is designed as a self-attention model and described in Fig.4(a). The output of this module is denoted as $\mathbf{F}_e$ and computed as:

$$\mathbf{F}_e = \mathcal{E}_\alpha(\text{Flatten}(\mathbf{F}_{roi})); \text{ where } \mathbf{F}_{roi} = \varphi(\mathcal{B}_\phi(\mathbf{I})) \tag{1}$$

3.2 Mask transformer decoding

Inspired by DETR[6], our mask transformer decoder transforms 3 learnable mask query-embeddings $\mathbf{Q}_o, \mathbf{Q}_v, \mathbf{Q}_a$, corresponding to occluder, visible and amodal masks, respectively. This mask transformer decoder $\mathcal{D}_\beta$ is designed as in the Fig.4 (b) which combines both self-attention ($\mathcal{A}_{self}$) and cross-attention($\mathcal{A}_{cross}$). The self-attention takes three learnable mask query-embeddings $\mathbf{Q} = [\mathbf{Q}_o, \mathbf{Q}_v, \mathbf{Q}_a] \in \mathbb{R}^{C\times 3}$ as input to model the correlation between queries. Meanwhile, the cross attention takes the output from the self-attention model $\mathcal{A}_{self}$, encoding feature $\mathbf{F}_e$ (i.e. the output of the mask transformer encoder) and their positional encodings as input to learn the coherence between individual query and feature maps $\mathbf{F}_e$. The mask transformer decoding $\mathcal{D}_\beta$ is described as follows.

$$\mathbf{Q} = \mathcal{D}_\beta(\mathbf{Q}, \mathbf{F}_e) = \mathcal{A}_{cross}\left(\mathbf{F}_e, (\mathcal{A}_{self}(\mathbf{Q}) \oplus \mathbf{Q})\right) \tag{2}$$

3.3 Invisible mask embedding

Our observation is that there are correlations between the visible mask and amodal mask as given in Fig.1. Specifically, the amodal mask is considered as the union of the visible mask (i.e modal mask) and the invisible mask (i.e. occluded parts). In order to model that relationship, we propose an invisible mask embedding module $\mathcal{I}_\theta$, which takes visible and amodal mask query instances (i.e. $\mathbf{Q}_v, \mathbf{Q}_a$) from the previous module $\mathcal{D}_\beta$ into consideration. The mask instances $\mathbf{Q}_v, \mathbf{Q}_a$) are concatenated into a long vector in $\mathbb{R}^{2C}$. The invisible mask embedding module $\mathcal{I}_\theta$ is designed as in Fig.4(c) with an MLP consisting of hidden layers. The output $\mathbf{I}_i \in \mathbb{R}^C$ is computed as follows:

$$\mathbf{I}_i = MLP([\mathbf{Q}_v; \mathbf{Q}_a]) \tag{3}$$

3.4 Mask predicting

The mask prediction module aims to predict four relevant mask instances as defined in AIS, i.e. occluder ($\mathbf{M}_o$), visible ($\mathbf{M}_v$), amodal ($\mathbf{M}_a$), and invisible ($\mathbf{M}_i$) masks. In this module, we first compute per-pixel ROI embeddings $\mathcal{E}_{roi} \in \mathbb{R}^{C\times H_m \times W_m}$. Indeed, $\mathcal{E}_{roi}$ represents the features of each pixel in the set of output masks $\mathbf{M} = [\mathbf{M}_o, \mathbf{M}_v, \mathbf{M}_a, \mathbf{M}_i]$. We define the per-pixel ROI embeddings as in the following equation.

$$\mathcal{E}_{roi} = \text{Unflatten}\left(\text{Flatten}(\mathbf{F}_{roi}) \oplus \mathbf{F}_e\right) \tag{4}$$

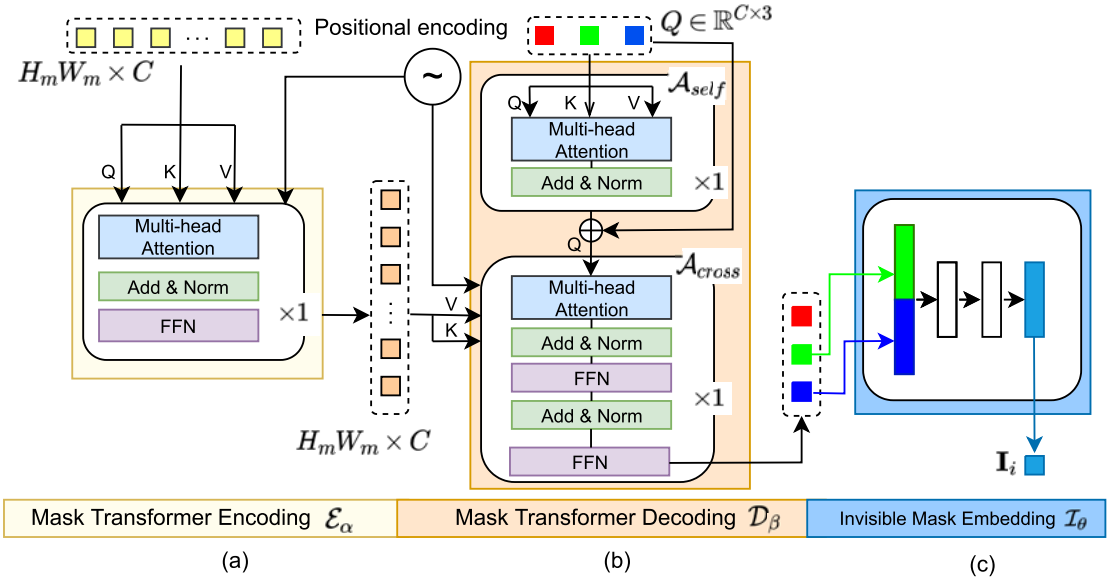

Figure 4: Illustration network architecture of AISFormer. (a): mask transformer encoder $\mathcal{E}_\alpha$ is designed as one block of self-attention, (b): mask transformer decoder $\mathcal{D}_\beta$ is designed as a combination of one block of self-attention ($\mathcal{A}_{self}$) and one block of cross-attention ($\mathcal{A}_{cross}$) and (c): invisible embedding $\mathcal{I}_\theta$ is designed as an MLP with two hidden layers.

We then apply a dot product ($\otimes$) on mask query embeddings $\mathbf{Q} = [\mathbf{Q}_o, \mathbf{Q}_v, \mathbf{Q}_a]$, invisible mask embedding $\mathbf{I}_i$ and per-pixel ROI embeddings $\mathcal{E}_{roi}$ to obtain the output masks $\mathbf{M}$ as in Eq.5. Our AISFormer is designed as an end-to-end framework and trained using the cross-entropy loss.

$$[\mathbf{M}_o, \mathbf{M}_v, \mathbf{M}_a, \mathbf{M}_i] = \mathcal{E}_{roi} \otimes [\mathbf{Q}_o, \mathbf{Q}_v, \mathbf{Q}_a, \mathbf{I}_i] \tag{5}$$

4 Experiments

4.1 Datasets, Metrics and Implementation Details

Datasets: We benchmark our AISFormer and conduct the comparison with other SOTA methods on three amodal segmentation datasets, namely KINS [32], COCOA-cls [46], and D2SA [15]. KINS is a large-scale traffic dataset with modal and amodal annotation masks of instances and is created based on the KITTI dataset [17]. Our AISFormer is trained on the training split with 7,474 images (95,311 instances) and tested on the testing split with 7,517 images (92,492 instances). D2SA is built based on the D2S(Densely Segmented Supermarket) [15] dataset with 60 categories of instances related to supermarket items. There are 2,000 training images and 3,600 testing images. The annotations are generated by overlapping objects on others. COCOA-cls is created based on COCO dataset [28]. In COCOA-cls, there are 2,476 training images and 1,223 testing images. It also contains 80 instance classes about common objects in context.

Metrics: In order to conduct the evaluation of AISFormer and comparison with other existing approaches, we adopt mean average precision (AP) and mean average recall (AR), commonly used evaluation metrics in AIS tasks.

Implementation Details: We implement our AISFormer based on Detectron2 [39], a Pytorch-based detection framework. For the KINS dataset, we use an SGD optimizer with a learning rate of 0.0025 and a batch size of 1 on 48000 iterations. For D2SA datasets, we also train with an SGD optimizer but with a learning rate of 0.005 and a batch size of 2 on 70000

Table 1: Performance comparison on KINS dataset. † indicates our reproduced results. On each backbone, the best scores are in **bold** and the second best scores are in underlines.

Sup.	Backbone	Model	Venue	Shape Prior	*AP*	AP_{50}	AP_{75}	*AR*
Weakly	ResNet-50	PCNet [44]	CVPR'20	✗	29.1	51.8	29.6	18.3
	ResNet-50	ABSU [31]	ICCV'21	✓	29.3	52.1	29.7	18.4
Fully	ResNet-50	VQVAE [21]	IEEE TMI'20	✓	30.3	–	–	–
	ResNet-50	VRSP-Net [41]	AAAI'21	✓	32.1	55.4	33.3	20.9
	ResNet-50	Mask R-CNN [19]	ICCV'17	✗	30.0	54.5	30.1	19.4
	ResNet-50	ORCNN [16]	WACV'19	✗	30.6	54.2	31.3	19.7
	ResNet-50	ASN [32]	CVPR'19	✗	32.2	–	–	–
	ResNet-50	**AISFormer**	–	✗	**33.8**	**57.8**	**35.3**	**21.1**
	ResNet-101	Mask R-CNN [19]†	ICCV'17	✗	30.2	54.3	30.4	19.5
	ResNet-101	BCNet [23]	CVPR'21	✗	28.9	–	–	–
	ResNet-101	BCNet [23] †	CVPR'21	✗	32.6	57.2	35.4	21.5
	ResNet-101	**AISFormer**	–	✗	**34.6**	**58.2**	**36.7**	**21.9**
	RegNet	APSNet [29]	CVPR'22	✗	**35.6**	–	–	–
	RegNet	**AISFormer**	–	✗	**35.6**	**59.9**	**37.0**	**22.5**

iterations. For COCOA-cls, since it has a smaller amount of training images, we train it on 10000 iterations with the learning rate of 0.0005 and a batch size of 2. The experiments have been conducted using an Intel(R) Core(TM) i9-10980XE 3.00GHz CPU and a Quadro RTX 8000 GPU.

4.2 Performance Comparison

Table 1 compares our AISFormer with SOTA AIS methods on the KINS dataset. AISFormer obtains steady improvement on distinct backbones (i.e. ResNet-50, ResNet-101, and RegNet). In particular, in comparison with the existing methods evaluated on KINS using ResNet-50 as the backbones, our method outperforms both SOTA methods with shape prior (e.g., VQVAE [21] by 3.5%*AP* and VRSP-Net [41] by 1.7%*AP*) and without using shape priors (e.g., Mask R-CNN [19] by 3.8%*AP* and ASN [32] by 1.6%*AP*), respectively. In addition, our method outperforms previous methods using ResNet-101 as the backbone by a large margin from $2\% - 4.4\%AP$ and has a comparable result with the current SOTA APSNet [29], using RegNet [33] as the backbone.

We also compare our model with all existing SOTA methods on the D2SA and COCOA-cls datasets, using ResNet-50 as the backbone as shown in Table 2. Compare with other existing shape-free approaches, our AISFormer achieves the SOTA performance on both datasets with a large margin. Specifically, compared with the second best i.e. BCNet [23], our AISFormer gains 2.39%*AP* and 1.74%*AR* on D2SA and 2.13%*AP* and 1.6%*AR* on COCOA-cls. Notably, our AISFormer outperforms shape prior VRSP-Net [41] on the COCOA-cls dataset while obtaining high performance and comparable results with VRSP-Net [41] on the D2SA dataset. Our observation and hypothesis on the D2SA dataset are that the majority of objects in D2SA are supermarket items and their shapes are standardized and unvarying (e.g. items are in the shape of a rectangle or tube). Thus, with standardized shape objects, the shape prior approach i.e. VRSP-Net [41] outperforms other shape-free approaches; however, it does not work well on other datasets which contain more variety of shapes. Generally, AIS performance on the D2SA dataset is higher than KINS and COCOA-cls datasets, which contain more variety of objects and shapes.

Table 2: Performance comparison on the D2SA and COCOA-cls datasets with ResNet-50 as the backbone. † indicates our reproduced results. In the category of without shape prior, the best scores are in **bold** and the second best scores are in underlines.

Model	Venue	Shape Prior	D2SA				COCOA-cls			
			AP	AP_{50}	AP_{75}	*AR*	*AP*	AP_{50}	AP_{75}	*AR*
VRSP-Net [41]	AAAI'21	✓	70.27	85.11	75.81	69.17	35.41	56.03	38.67	37.11
Mask R-CNN [19]	ICCV'17	✗	63.57	83.85	68.02	65.18	28.03	53.68	25.36	29.83
ORCNN [16]	WACV'19	✗	64.22	83.55	69.12	65.25	28.03	53.68	25.36	29.83
ASN [32] †	CVPR'19	✗	63.94	84.35	69.57	65.20	35.33	58.82	37.10	35.50
BCNet [23] †	CVPR'21	✗	65.97	84.23	72.74	66.90	35.14	58.84	36.65	35.80
AISFormer	–	✗	**68.36**	**85.08**	**74.58**	**68.64**	**37.27**	**59.69**	**40.70**	**37.40**

4.3 Ablations

We further investigate the effectiveness of our proposed AISFormer. We conduct six experiments as shown in Table 3 as follows:

• **Exp #1**: AISFormer with only one amodal mask query embeddings (i.e. $\mathbf{Q} = [\mathbf{Q}_a]$).

• **Exp #2**: AISFormer with amodal mask query and occluder query embeddings (i.e. $\mathbf{Q} = [\mathbf{Q}_a, \mathbf{Q}_o]$). By adding occluder query embedding, the performance improves 2.35% AP on D2SA and 1.9% AP on KINS. This shows the effectiveness of occluder query embedding.

• **Exp #3**: AISFormer with amodal mask query and visible query embeddings (i.e. $\mathbf{Q} = [\mathbf{Q}_a, \mathbf{Q}_v]$). By adding visible query embedding, the performance improves 1.82% AP on D2SA and 1.69% AP on KINS. This shows the effectiveness of visible query embedding.

• **Exp #4**: AISFormer with amodal mask query, visible query embeddings (i.e. $\mathbf{Q} = [\mathbf{Q}_a, \mathbf{Q}_v]$), and invisible query embedding. This shows the effectiveness of invisible mask embedding, which helps to gain 1.08% AP on D2SA and 0.23% AP on KINS.

• **Exp #5**: Both amodal mask query, visible query and occluder embeddings (i.e. $\mathbf{Q} = [\mathbf{Q}_o, \mathbf{Q}_a, \mathbf{Q}_v]$) are used and it obtains better performance on each individual embedding. This shows the effectiveness of learnable query embeddings.

• **Exp #6**: This is our proposed network architecture, AISFormer, which obtains the best performance. This shows the effectiveness of both learnable query embeddings and invisible mask embedding when combining them into an end-to-end framework.

We further visualize the attention maps of learnable queries as in Fig.5. More qualitative results are included in the supplementary.

Table 3: Effectiveness of occluder mask embeddings (Exps. #1 vs. #2, #3 vs. #5), visible mask embeddings (Exps. #1 vs. #3, #2 vs. #5), and invisible mask embeddings (Exps. #3 v.s.#4, #5 v.s.#6) with ResNet-50 as the backbone on the D2SA and KINS datasets.

Exp.	Amodal	Occluder	Visible	Invisible	D2SA		KINS	
					AP	AP_{50}	*AP*	AP_{50}
#1	✓	✗	✗	✗	64.66	81.30	31.01	54.28
#2	✓	✓	✗	✗	67.01	83.92	32.91	55.39
#3	✓	✗	✓	✗	66.48	83.01	32.70	55.16
#4	✓	✗	✓	✓	67.56	84.72	32.93	55.45
#5	✓	✓	✓	✗	68.17	84.63	33.21	56.20
#6	✓	✓	✓	✓	68.36	85.06	33.78	57.80

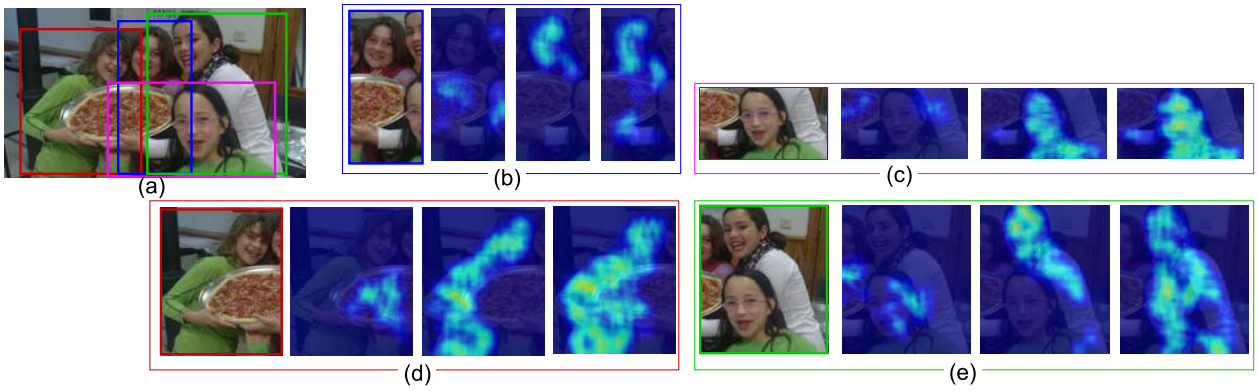

Figure 5: Attention visualization of query embeddings. (a): Input image with four ROIs. (b), (c), (d), (e): attention feature maps of queries in each ROI. For each ROI, from left-right: ROI, occluder query embedding, visible query embedding, and amodal query embedding.

Conclusion

We have proposed AISFormer, an AIS framework, with Transformer-based mask heads. The proposed is designed as an end-to-end framework with four modules: feature encoding, mask transformer decoding, invisible mask embedding, and mask predicting. Our AISFormer can explicitly model the complex coherence between occluder, visible, amodal, and invisible mask instances within ROI by treating them as learnable queries. Results and ablation study on three benchmarks, namely KINS, D2SA, and COCOA-cls, show the effectiveness of our proposed learnable queries and indicate that our AISFormer obtains the SOTA AIS performance compared to the existing methods. Our future research will integrate the shape prior into AISFormer as well as investigate AISFormer on other modalities such as time-lapse, and videos.

Acknowledgments

This material is based upon work supported by the National Science Foundation (NSF) under Award No OIA-1946391, NSF 1920920, and NSF 2223793.

References

[1] Hervé Abdi and Lynne J Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.

[2] Nasir Ahmed, T_ Natarajan, and Kamisetty R Rao. Discrete cosine transform. *IEEE transactions on Computers*, 100(1):90–93, 1974.

[3] Anurag Arnab and Philip H. S. Torr. Bottom-up instance segmentation using deep higher-order crfs. In Richard C. Wilson, Edwin R. Hancock, and William A. P. Smith, editors, *Proceedings of the British Machine Vision Conference 2016, BMVC 2016, York, UK, September 19-22, 2016*. BMVA Press, 2016. URL http://www.bmva.org/bmvc/2016/papers/paper019/index.html.

[4] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. YOLACT: real-time instance segmentation. In *2019 IEEE/CVF International Conference on Computer Vi-*

sion, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019, pages 9156–9165. IEEE, 2019. doi: 10.1109/ICCV.2019.00925. URL https://doi.org/10.1109/ICCV.2019.00925.

[5] Zhaowei Cai and Nuno Vasconcelos. Cascade R-CNN: delving into high quality object detection. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 6154–6162. IEEE Computer Society, 2018. doi: 10.1109/CVPR.2018.00644. URL http://openaccess.thecvf.com/content_cvpr_2018/html/Cai_Cascade_R-CNN_Delving_CVPR_2018_paper.html.

[6] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[7] Hao Chen, Kunyang Sun, Zhi Tian, Chunhua Shen, Yongming Huang, and Youliang Yan. Blendmask: Top-down meets bottom-up for instance segmentation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 8570–8578. IEEE, 2020. doi: 10.1109/CVPR42600.2020.00860. URL https://doi.org/10.1109/CVPR42600.2020.00860.

[8] 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. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 4974–4983. Computer Vision Foundation / IEEE, 2019. doi: 10.1109/CVPR.2019.00511. URL http://openaccess.thecvf.com/content_CVPR_2019/html/Chen_Hybrid_Task_Cascade_for_Instance_Segmentation_CVPR_2019_paper.html.

[9] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017.

[10] Xinlei Chen, Ross B. Girshick, Kaiming He, and Piotr Dollár. Tensormask: A foundation for dense object segmentation. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 2061–2069. IEEE, 2019. doi: 10.1109/ICCV.2019.00215. URL https://doi.org/10.1109/ICCV.2019.00215.

[11] Tianheng Cheng, Xinggang Wang, Lichao Huang, and Wenyu Liu. Boundary-preserving mask r-cnn. In *European conference on computer vision*, pages 660–676. Springer, 2020.

[12] Bin Dong, Fangao Zeng, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. Solq: Segmenting objects by learning queries. *Advances in Neural Information Processing Systems*, 34:21898–21909, 2021.

[13] David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52 (4):1289–1306, 2006.

[14] Yuxin Fang, Shusheng Yang, Xinggang Wang, Yu Li, Chen Fang, Ying Shan, Bin Feng, and Wenyu Liu. Instances as queries. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6910–6919, 2021.

[15] Patrick Follmann, Tobias Bottger, Philipp Hartinger, Rebecca Konig, and Markus Ulrich. Mvtec d2s: densely segmented supermarket dataset. In *Proceedings of the European conference on computer vision (ECCV)*, pages 569–585, 2018.

[16] 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.

[17] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11): 1231–1237, 2013.

[18] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D Cubuk, Quoc V Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2918–2928, 2021.

[19] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2980–2988. IEEE Computer Society, 2017. doi: 10.1109/ICCV.2017.322. URL https://doi.org/10.1109/ICCV.2017.322.

[20] Jie Hu, Liujuan Cao, Yao Lu, ShengChuan Zhang, Yan Wang, Ke Li, Feiyue Huang, Ling Shao, and Rongrong Ji. Istr: End-to-end instance segmentation with transformers. *ArXiv preprint*, abs/2105.00637, 2021. URL https://arxiv.org/abs/2105.00637.

[21] Won-Dong Jang, Donglai Wei, Xingxuan Zhang, Brian Leahy, Helen Yang, James Tompkin, Dalit Ben-Yosef, Daniel Needleman, and Hanspeter Pfister. Learning vector quantized shape code for amodal blastomere instance segmentation. *IEEE TRANSACTIONS ON MEDICAL IMAGING*, 2020.

[22] Xu Jia, Bert De Brabandere, Tinne Tuytelaars, and Luc Van Gool. Dynamic filter networks. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 667–675, 2016. URL https://proceedings.neurips.cc/paper/2016/hash/8bf1211fd4b7b94528899de0a43b9fb3-Abstract.html.

[23] 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.

[24] Lei Ke, Martin Danelljan, Xia Li, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Mask transfiner for high-quality instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4412–4421, 2022.

[25] Youngwan Lee and Jongyoul Park. Centermask: Real-time anchor-free instance segmentation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 13903–13912. IEEE, 2020. doi: 10.1109/CVPR42600.2020.01392. URL https://doi.org/10.1109/CVPR42600.2020.01392.

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

[27] Yi Li, Haozhi Qi, Jifeng Dai, Xiangyang Ji, and Yichen Wei. Fully convolutional instance-aware semantic segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 4438–4446. IEEE Computer Society, 2017. doi: 10.1109/CVPR.2017.472. URL https://doi.org/10.1109/CVPR.2017.472.

[28] 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.

[29] Rohit Mohan and Abhinav Valada. Amodal panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21023–21032, 2022.

[30] Alejandro Newell, Zhiao Huang, and Jia Deng. Associative embedding: End-to-end learning for joint detection and grouping. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 2277–2287, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/8edd72158ccd2a879f79cb2538568fdc-Abstract.html.

[31] Khoi Nguyen and Sinisa Todorovic. A weakly supervised amodal segmenter with boundary uncertainty estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7396–7405, 2021.

[32] Lu Qi, Li Jiang, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Amodal instance segmentation with KINS dataset. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 3014–3023. Computer Vision Foundation / IEEE, 2019. doi: 10.1109/CVPR.2019.00313. URL http://openaccess.thecvf.com/content_CVPR_2019/html/Qi_Amodal_Instance_Segmentation_With_KINS_Dataset_CVPR_2019_paper.html.

[33] Ilija Radosavovic, Raj Prateek Kosaraju, Ross B. Girshick, Kaiming He, and Piotr Dollár. Designing network design spaces. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19,*

2020, pages 10425–10433. IEEE, 2020. doi: 10.1109/CVPR42600.2020.01044. URL https://doi.org/10.1109/CVPR42600.2020.01044.

[34] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[35] Yihong Sun, Adam Kortylewski, and Alan Yuille. Weakly-supervised amodal instance segmentation with compositional priors. *ArXiv preprint*, abs/2010.13175, 2020. URL https://arxiv.org/abs/2010.13175.

[36] Yihong Sun, Adam Kortylewski, and Alan Yuille. Amodal segmentation through out-of-task and out-of-distribution generalization with a bayesian model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1215–1224, 2022.

[37] Abhinav Valada, Ankit Dhall, and Wolfram Burgard. Convoluted mixture of deep experts for robust semantic segmentation. In *IEEE/RSJ International conference on intelligent robots and systems (IROS) workshop, state estimation and terrain perception for all terrain mobile robots*, volume 2, 2016.

[38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

[39] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019.

[40] Yuting Xiao, Yanyu Xu, Ziming Zhong, Weixin Luo, Jiawei Li, and Shenghua Gao. Amodal segmentation based on visible region segmentation and shape prior. *AAAI Conference on Artificial Intelligence*, 2020.

[41] Yuting Xiao, Yanyu Xu, Ziming Zhong, Weixin Luo, Jiawei Li, and Shenghua Gao. Amodal segmentation based on visible region segmentation and shape prior. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 2995–3003, 2021.

[42] 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. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 12190–12199. IEEE, 2020. doi: 10.1109/CVPR42600.2020.01221. URL https://doi.org/10.1109/CVPR42600.2020.01221.

[43] Hui Ying, Zhaojin Huang, Shu Liu, Tianjia Shao, and Kun Zhou. Embedmask: Embedding coupling for one-stage instance segmentation. *ArXiv preprint*, abs/1912.01954, 2019. URL https://arxiv.org/abs/1912.01954.

[44] Xiaohang Zhan, Xingang Pan, Bo Dai, Ziwei Liu, Dahua Lin, and Chen Change Loy. Self-supervised scene de-occlusion. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 3783–3791. IEEE, 2020. doi: 10.1109/CVPR42600.2020.00384. URL https://doi.org/10.1109/CVPR42600.2020.00384.

[45] Gang Zhang, Xin Lu, Jingru Tan, Jianmin Li, Zhaoxiang Zhang, Quanquan Li, and Xiaolin Hu. Refinemask: Towards high-quality instance segmentation with fine-grained features. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6861–6869, 2021.

[46] Yan Zhu, Yuandong Tian, Dimitris N. Metaxas, and Piotr Dollár. Semantic amodal segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 3001–3009. IEEE Computer Society, 2017. doi: 10.1109/CVPR.2017.320. URL https://doi.org/10.1109/CVPR.2017.320.