End-to-end Amodal Video Instance Segmentation

Jasmin Breitenstein
j.breitenstein@tu-bs.de
Kangdong Jin
k.jin@tu-bs.de
Aziz Hakiri
a.hakiri@tu-bs.de
Marvin Klingner
m.klingner@tu-bs.de
Tim Fingscheidt
t.fingscheidt@tu-bs.de

Institute for Communications Technology,
Technische Universität Braunschweig
Schleinitzstraße 22, 38106 Braunschweig,
Germany

Abstract

Amodal perception is the important ability of humans to imagine the entire shape of occluded objects. This ability is crucial for safety-relevant perception tasks such as autonomous movement of robots and vehicles. Existing methods mostly focus on amodal perception in single images. However, video understanding is important for amodal perception as it provides additional cues for perceiving occlusions. In this paper, we are the first to present an end-to-end trainable amodal video instance segmentation method. Specifically, we present a strategy to extend existing instance segmentation models by an amodal mask branch as well as a tracking branch, inspired by video instance segmentation (VIS) methods. The tracking branch allows to not only predict amodal and visible masks at the same time, but also to connect them over time by predicting video-based instance IDs. Our video-based method `VATrack` outperforms the existing image-based state-of-the-art methods on the commonly used SAIL-VOS dataset's benchmarks in all amodal metrics, while also improving most modal (i.e., visible) metrics. Additionally, we introduce a novel video-based evaluation where our method may serve as a baseline for future research on amodal VIS. Code for `VATrack` can be found on github[1].

1 Introduction

Many computer vision tasks aim at reaching at least human-level perception performance, leading to many advances towards high-performing recognition tasks [2, 12, 13, 21, 22, 25, 32, 33, 35, 43, 44, 48, 52] with a wide range of applications, e.g., automated driving [6, 18], and video surveillance [46]. In all applications, detecting and segmenting the present instances, as well as monitoring their movement throughout a sequence is crucial for safety, i.e., it is critical to understand the whereabouts of all actors or objects throughout a relevant

[1] https://github.com/ifnspaml/vatrack

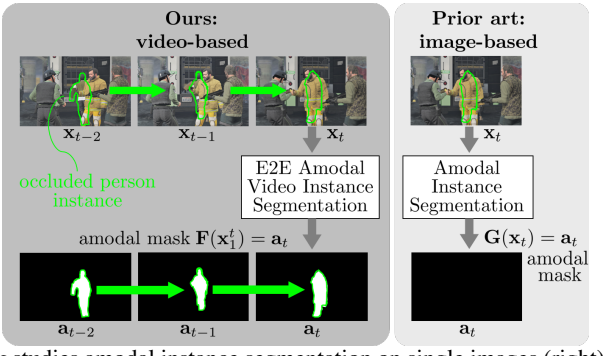

Figure 1: Prior work studies amodal instance segmentation on single images (right). However, to track heavy and total occlusions, video-based methods are required (left): Our end-to-end trainable amodal video instance segmentation method `VATrack` recovers the totally occluded person (green) at time t, as the method has seen earlier inputs $\mathbf{x}_1, \ldots, \mathbf{x}_{t-1}$ and memorizes internal states. Any single image-based method can only fail for total occlusion.

time span [9, 10, 17, 30]. This is especially important for *occlusions of instances* in a scenario, either due to a temporal full occlusion, or a partial occlusion, which should be both perceived, e.g., to avoid accidents due to such occlusion-related corner cases [3, 4, 7, 23].

Humans perceive the full shape of occluded instances. In the frames $\mathbf{x}_{t-2}, \mathbf{x}_{t-1}, \mathbf{x}_t$ in Fig. 1, they understand the full shape of the occluded person (green). This amodal perception is a crucial ability for human video understanding [34, 36]. Perception systems only perceive what is visible, and cannot recognize instances under heavy occlusion, especially when characteristic parts remain unseen [40, 58], In Fig. 1 the image-based prior art fails to detect the fully occluded person in frame $\mathbf{x}_t$. Hence, amodal perception in perception systems will impact many applications suffering from poor perception and improve video understanding.

End-to-end video instance segmentation (VIS) methods provide temporal context to instance segmentation and have profited from advances in both instance segmentation and tracking [27, 49, 50, 54]. However, the performance weakens when faced with occlusions [40]. *While methods have been investigated to improve detection of occluded objects, we are the first to bring together end-to-end VIS with amodal segmentation techniques to provide temporally consistent amodal instance masks for video sequences.* Our concept can be seen on the left in Fig. 1. It can predict the green instance at time instant t, while the image-based amodal instance segmentation on the right of Fig. 1 fails due to the missing temporal context.

To achieve end-to-end trainable amodal VIS, we extend standard instance segmentation by two architectural contributions: First, we add an amodal mask head for amodal instance segmentation. It outputs an amodal mask $\mathbf{a}_t$ as in Fig. 1. Joint training with the visible instance segmentation head enables implicitly learned cues for mutual benefit of visible and amodal segmentation. Second, we introduce an additional tracking head to provide video-level instance IDs to associate different amodal masks over time and to improve amodal mask quality for heavy and total occlusions. For the tracking method, we adapt well-known VIS methods [38, 50] to our purposes. *Our proposed novel approach for a jointly trained visible and amodal VIS termed* `VATrack` *leads to visible and amodal performance gains.*

While there are benchmarks for image-based amodal segmentation [24, 35, 41, 59], currently there exists no video-based evaluation. Hence, we introduce a video-based evaluation as well as video-based metrics on the SAIL-VOS dataset [24] for amodal VIS. We present `VATrack` variants as baseline for future research on amodal VIS. *Additionally, we show that our video-based model significantly outperforms existing approaches in a single image-*

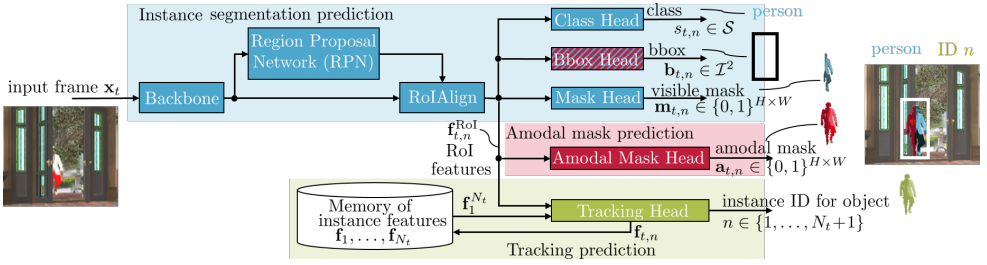

Figure 2: Inference of our *proposed jointly trained visible and amodal video instance segmentation* (VATrack): Given an input frame $\mathbf{x}_t$, the network outputs the corresponding instance class $s_{t,n}$, its amodal bounding box $\mathbf{b}_{t,n}$, and both the visible and amodal masks $\mathbf{m}_{t,n}, \mathbf{a}_{t,n}$. The tracking head extracts features $\mathbf{f}_{t,n}$ from the RoI features $\mathbf{f}^{\text{RoI}}_{t,n}$ for comparison with the existing memory of instance features $\mathbf{f}_1, \ldots, \mathbf{f}_{N_t}$ to identify a predicted instance with either a known instance ID $n \in \mathcal{N}_t = \{1, \ldots, N_t\}$ or to assign a new instance ID $n = N_t + 1$. The memory of instance features is updated with the extracted features $\mathbf{f}_{t,n}$, hence we denote the memory features without the frame index t.

based evaluation, thereby setting a new state-of-the-art performance on SAIL-VOS.

Our contributions can be summarized as follows: First, we propose the first end-to-end trainable method for amodal video instance segmentation. Second, our joint approach for visible and amodal VIS VATrack improves both visible and amodal performance, setting a new state of the art. Finally, we introduce the first video-based evaluation for amodal VIS.

2 Related Work

We review works from both *amodal* segmentation and *video* instance segmentation.

Amodal instance segmentation: In the last years, amodal segmentation has gained a lot of attention for semantic segmentation [6, 8, 39] and instance segmentation [31, 37, 42, 45]. First methods for amodal segmentation predicted the amodal instance mask given an input image, or given both the image and the visible mask [59], or by post-processing [29]. Many methods since have adapted the instance segmentation network Mask R-CNN [22] to predict amodal masks using multiple head configurations for the network: Follmann et al. [19] employ an invisible mask head that predicts the occluded instance parts. Qi et al. [41] add a classifier to decide whether a region of interest (RoI) proposal is occluded. Other works use distance information in the learning process [55], include occlusion reasoning [53], model occlusions via hierarchical fusion [1] or use graph convolutional networks as occluder segmentation and occludee segmentation head [28]. Hu et al. [24] train Mask R-CNN with an amodal and visible mask head. In our work, we adapt this strategy to the amodal VIS task. SAIL-VOS [24] is a synthetic dataset comprised of scenes from GTA V and to our knowledge so far the only publicly available dataset that allows amodal VIS, meaning that it does not only provide amodal and visible instance masks but also annotations to track those throughout a video sequence. This is our reason for employing SAIL-VOS. While amodal instance segmentation can already provide valuable amodal masks for single images, they lack temporal context, which is, however, crucial to resolve total occlusions.

Video instance segmentation: The VIS task was introduced by Yang et al. [50], who extended the Mask R-CNN by a tracking head to MaskTrack R-CNN. Many VIS methods built upon their work, mainly differing in tracking [27, 38]. Here, we build upon the tracking methods of [38, 50] for amodal VIS. For more information, we refer to surveys [20, 57].

While SAIL-VOS [24] is the first dataset to enable amodal VIS, all methods evaluated on this dataset are so far purely image-based. Yao et al. [51] concurrently developed a method

for self-supervised amodal VIS, which is not end-to-end trainable, but takes as input the current frame, optical flow and visible instance masks to predict the amodal masks. The dependence on the visible masks means that this method also cannot predict the amodal mask for video frame $\mathbf{x}_t$ in Fig. 1. In contrast, we propose `VATrack` for *end-to-end* amodal VIS that, given only a video sequence provides *both the visible and amodal instance masks*. Additionally, we report results on a more comprehensive video level as well as image level, while in [51] only image-level results are supplied.

3 New Amodal Video Instance Segmentation

For amodal VIS, we build upon an instance segmentation (blue in Fig. 2). This can be extended by a tracking method (+green) to perform VIS, see Fig. 2. We investigate two types of tracking methods, the `MaskTrack R-CNN` [50] and the `QDTrack` [38] method. We first explain the instance segmentation blocks and provide details of the tracking methods. Then, we describe the additional amodal mask head (red block) for joint amodal and visible VIS of our proposed `VATrack`, and outline the training approaches for amodal and visible VIS.

3.1 Instance Segmentation Prediction

Fig. 2 shows the inference for our proposed visible and amodal VIS method `VATrack`. The blue part corresponds to an instance segmentation prediction based on `Mask R-CNN` [22]. Input to our method in Fig. 2 is a normalized input frame $\mathbf{x}_t \in [0,1]^{H\times W\times C}$ of a video $\mathbf{x}_1^T$, with H,W being the image height and width, $C=3$ the channel size, and frame indices $t \in \mathcal{T} = \{1,\dots,T\}$ of the video of length T. The previous frames $\mathbf{x}_1^{t-1}$ have been processed by the network consisting of a backbone, region proposal network (RPN) and region of interest align (RoIAlign) to extract features $\mathbf{f}_{t,n}^{\text{RoI}}$ of regions of interest (RoI) with $n \in \mathcal{N}_t = \{1,\dots,N_t\}$ being one instance ID out of the N_t uniquely observed instances until frame t. From these features the instance segmentation prediction, consisting of class, bounding box and mask head, predicts the class $s_{t,n} \in \mathcal{S} = \{1,\dots,S\}$, S being the number of instance classes, per instance $n \in \mathcal{N}_t$, the bounding box $\mathbf{b}_{t,n} \in \mathcal{I}^2$ with the set $\mathcal{I} = \{(1,1),\dots,(H,W)\}$ of pixel indices, and the corresponding instance mask $\mathbf{m}_{t,n} \in \{0,1\}^{H\times W}$. We denote the set of all instances in video $\mathbf{x}_1^T$ as $\mathcal{N} = \{1,\dots,N\}$, with N being the number of unique instances in the entire video sequence $\mathbf{x}_1^T$, where $\mathcal{N}_t \subset \mathcal{N}$. The mask, bounding box and class head are trained using the loss functions of `Mask R-CNN` [22].

3.2 Tracking Method

In Fig. 2, the instance segmentation (blue) is extended by a tracking method (+green). During inference, a memory storing instance features $\mathbf{f}_n \in \mathbb{R}^{h\times w\times c}$ is updated throughout the video, where h,w,c are the tracking-method-specific height, width, and channel number of the instance features extracted in the tracking head. Instance ID $n \in \mathcal{N}_t = \{1,\dots,N_t\}$ is one out of N_t uniquely observed instances until frame index t. The memory at a current frame index t stores instance features $\mathbf{f}_1,\dots,\mathbf{f}_{N_t}$. The number of instances N_t at time index t is equal to or smaller than the total number N of instances. Given a predicted instance of ID n, there are two options: First the predicted instance is identified as an already known instance, i.e., $n \in \{1,\dots,N_t\}$. Then, the feature vector $\mathbf{f}_{t,n}$ updates the instance feature vector $\mathbf{f}_n \leftarrow \mathbf{f}_{t,n}$ of instance n in the memory. Second, ID n is assigned as $n = N_t + 1$, i.e., a new instance. In this case, new instance features $\mathbf{f}_{N_t+1} \leftarrow \mathbf{f}_{t,n}$ are saved in the instance memory. Afterwards, both $N_t \leftarrow N_t + 1$ and $\mathcal{N}_t \leftarrow \mathcal{N}_t \cup \{n\}$ are updated. When the final video frame $\mathbf{x}_T$ is processed by

the network, $\mathcal{N}_t = \mathcal{N}$ contains all unique instances N of the video sequence[2] $\mathbf{x}_1^T$.

For the `MaskTrack R-CNN` [50] tracking method, a Softmax-based probability is calculated given the current RoI features $\mathbf{f}_{t,n}^{\text{RoI}}$ and the memory instance features $\mathbf{f}_1, \dots, \mathbf{f}_{N_t}$ for assigning a known instance ID $n \in \mathcal{N}_t$ or a new instance ID $n = N_t + 1$ to the features of a predicted instance. For the `QDTrack` [38] method, in the tracking head in Fig. 2, a bi-directional Softmax is applied to identify instance proposals $\mathbf{f}_{t,n}^{\text{RoI}}$ with the memory instance features $\mathbf{f}_1, \dots, \mathbf{f}_{N_t}$, resulting in a known instance ID $n \in \mathcal{N}_t$ or a new instance ID $n = N_t + 1$.

3.3 Amodal Mask Prediction

The proposed *joint prediction of visible and amodal VIS* of `VATrack` is shown in Fig. 2. Now we describe the amodal mask prediction. The amodal mask head has the same architecture as the (visible) mask head, taking as input the RoI features $\mathbf{f}_{t,n}^{\text{RoI}}$ for predicted instance n at frame index t and predicting the amodal mask $\mathbf{a}_{t,n} \in \{0,1\}^{H \times W}$. The amodal mask head is trained with a cross-entropy loss $J^{\text{amodal}} = J^{\text{CE}}$ same as the visible mask head.

With the amodal mask head, the question arises whether the bounding box (bbox) head should output the visible or the amodal bbox. In Fig. 2 this can be seen by the mixed colorization. Naturally, it is not possible to use the visible bounding boxes, as masks are only predicted *inside* the bbox, hence making amodal mask prediction impossible. Per default, we use only the amodal bbox for training. In the supplementary, we investigate whether using only an amodal bbox head, or including an additional amodal bbox head is more beneficial.

3.4 Training Approaches

Separate training: Here, we omit the amodal mask head of the `VATrack` framework in Fig. 2. When training with just one mask head, the total loss is given as,

$$J^{\text{separate}} = J^{\text{RPN}} + J^{\text{bbox}} + J^{\text{cls}} + J^{\text{mask}} + J^{\text{track}}, \tag{1}$$

where J^{RPN} is the RPN loss, J^{bbox} is the bbox head loss, and J^{cls} is the class head loss. The mask head is trained using the cross-entropy loss $J^{\text{mask}} = J^{\text{CE}}$.

The ground truth input to the loss functions depends on whether we train on the visible or on the amodal masks. If the method is trained on the visible masks $\overline{\mathbf{m}}_{t,n}$, we refer to it by the name of the tracking approach. If the amodal masks $\overline{\mathbf{a}}_{t,n}$ are used, we refer to it as `AmodalTrack`, which can be either `MT`- or `QD`-based, depending on the tracking method.

To learn meaningful embeddings of the memory instance features $\mathbf{f}_1, \dots, \mathbf{f}_{N_\tau}$, those features are also extracted from a reference frame $\mathbf{x}_\tau \in [0,1]^{H \times W \times C}$, where τ is selected from the range $\tau \in \{t - \Delta t, \dots, t + \Delta t\}$. Typically Δt is chosen such that this interval allows for instances to appear in both frames, so similar representations can be learned for them. Naturally, we need a video length $T \geq 2$ for this training approach. Then, both the RoI features from the input frame $\mathbf{x}_t$ and from the reference frame $\mathbf{x}_\tau$ are fed into the tracking head to produce feature embeddings $\mathbf{f}_{t,n}, \mathbf{f}_{\tau,n} \in \mathbb{R}^{h \times w \times c}$ per proposed instance n.

The tracking loss J^{track} depends on the chosen tracking method. For the `MaskTrack R-CNN` tracking method, J^{track} is also a cross-entropy loss calculated between the ground truth instance ID $\overline{n}$ and the tracking branch's predicted probability that the extracted features $\mathbf{f}_{t,n}$ belong to the already known instances with memory features $\mathbf{f}_1, \dots, \mathbf{f}_{N_t}$ (extracted from the ground truth instances of the reference frame $\mathbf{x}_\tau$ in training), or to a new instance [50].

[2] Note that in practice, the instance feature memory would be restricted by the available hardware. This is relevant, e.g., if an instance vanishes from the field of view. Its features are stored in memory until the memory's limit is reached. Then features of the instance with longest absence would be discarded. Such limitations are typically disregarded in (amodal) VIS research [38, 50, 51].

For the `QDTrack` tracking method, quasi-dense similarity learning leads to $J^{\text{track}} = J^{\text{emb}} + \lambda J^{\text{aux}}$: A contrastive loss J^{emb} between the feature embeddings of the RoI features of the reference and input frame is used to learn the feature embedding of the tracking branch so that for the same instance n from the input and the reference frame, feature embeddings $\mathbf{f}_{t,n}, \mathbf{f}_{\tau,n}$ are encouraged to be close to each other, while the embeddings of different instances are encouraged to be far away from each other, for details see [38]. An auxiliary loss term J^{aux} between the same feature embeddings is needed to stabilize the training of the tracking branch of `QDTrack`. We refer to [38] and [50] for details about the losses and the more detailed operation of the two VIS methods and their respective tracking branches.

Joint training (`VATrack`): The amodal mask head is also trained using the cross-entropy loss $J^{\text{mask}} = J^{\text{CE}}$. To distinguish between the heads, we use the notation J^{visible} or J^{amodal}. While in the separate training case, the loss term J^{mask} has a weight of 1.0 in (1), we found that if we incorporate a second mask head, results differ depending on the weighting of both loss terms. Instead we weigh with pre-selected hyperparameters λ_1, λ_2 and report an ablation study in the supplementary. Thus we obtain the total loss for joint training by

$$\begin{aligned} J^{\text{joint}} = & J^{\text{RPN}} + J^{\text{bbox}} + J^{\text{cls}} + J^{\text{track}} \\ & + \lambda_1 \cdot J^{\text{visible}} + \lambda_2 \cdot J^{\text{amodal}}, \end{aligned} \tag{2}$$

where we distinguish the loss terms of the visible (J^{visible}) and the amodal (J^{amodal}) mask prediction, even though they denote the same loss function and just differ in their targets.

The bbox loss J^{bbox} is trained using the amodal ground truth bbox $\overline{\mathbf{b}}_{t,n}$. As the mask is predicted inside the bbox, this allows to also learn the prediction of the visible instance masks since the amodal bbox is at least the same size of the visible bbox. For an ablation using two bbox heads in the supplementary, we also have two loss terms for each bbox head, i.e., $J^{\text{bbox}}_{\text{amodal}}$ and $J^{\text{bbox}}_{\text{visible}}$, which we each weigh for simplicity by the canonical value of 1.0.

We use the visible ground truth $\overline{\mathbf{m}}_{t,n}$ to train the (visible) mask head, and the amodal ground truth $\overline{\mathbf{a}}_{t,n}$ to train the amodal mask head. The ground truth for the tracking head and the class head are not affected by the visibility of an instance, as neither the instance ID n nor its instance class change depending on the visibility of the instance in a frame $\mathbf{x}_t$.

4 Experimental Evaluation and Discussion

In the following, we provide a description of our used datasets and metrics. Afterwards, we provide an image-level and video-level evaluation including a state-of-the-art comparison.

4.1 Dataset

We use the SAIL-VOS dataset [24]. To our knowledge, it is up to date the only available dataset providing amodal VIS labels. The amodal ground truth in SAIL-VOS is obtained from the underlying game engine by specifically toggling the visibility of all objects in a scene on and off [24]. The dataset contains 201 videos split into training data (160 videos), and validation data (41 videos). The frames have a resolution of 1280×800. While the dataset has 162 annotated instance classes, experiments typically consider a fixed subset of $S = 24$ classes [24]. We aim to predict the correct class and mask per instance, using the same evaluation setup as Hu et al. [24], whose method also serves as a baseline. For more details on the SAIL-VOS dataset, we refer to [24].

The SAIL-VOS dataset contains videos with jump cuts. Tracking is one of the core methods within VIS, but naturally does not work across jump cuts. Accordingly, we also investigate on a derivative of SAIL-VOS for training and evaluation, which we term SAIL-VOS-cut. During training on the SAIL-VOS data, we observe loss irregularities whenever

	Method	Backbone	V	A	TC	AP	AP_{50}	AP^{P}_{50}	AP^{H}_{50}	AP^{L}_{50}	AP^{M}_{50}	AP^{S}_{50}
image-based	MaskAmodal [24]	?		✓		13.0	23.0	24.3	16.7	36.6	21.5	6.1
	MaskJoint [24]	?	✓	✓		14.1	24.8	24.3	18.9	37.8	21.5	5.7
	MaskAmodal*	RX-101		✓		16.3	25.6	27.4	17.1	35.2	24.2	10.1
	MaskJoint *	RX-101	✓	✓		16.7	25.6	26.9	17.3	33.0	22.3	9.0
video-based	AmodalTrack (MT-based)	RX-101		✓	✓	15.9	25.7	24.9	17.8	36.8	22.8	11.2
	Ours: VATrack (MT-based)	RX-101	✓	✓	✓	16.4	26.0	24.9	18.0	**38.6**	22.5	10.6
	AmodalTrack (QD-based)	RX-101		✓	✓	17.8	27.4	29.2	18.6	34.7	26.8	11.4
	Ours: VATrack (QD-based)	RX-101	✓	✓	✓	**18.3**	**28.6**	**29.7**	**20.1**	38.1	**26.9**	**15.7**

Table 1: **Amodal** instance segmentation **image-level** performance (%) on validation data for image-based methods (blue, results on SAIL-VOS = SAIL-VOS-cut dataset), and for video-based methods (orange, results on SAIL-VOS-cut dataset). Marker $*$ denotes resimulated results. Checkmarks indicate whether the method predicts visible (V) and/or amodal (A) masks. The video-based methods take temporal context (TC) into account, shown by checkmarks. Results have been produced using the ResNext-101 (RX-101) backbone, Hu et al. [24] do not report their backbone (marked by: ?). Best results are in **bold**, second best are underlined.

	Method	Backbone	V	A	TC	AP	AP_{50}	AP^{P}_{50}	AP^{H}_{50}	AP^{L}_{50}	AP^{M}_{50}	AP^{S}_{50}
image-based	Mask R-CNN[24]	?	✓			14.3	24.1	24.7	17.2	**42.8**	21.3	4.9
	MaskJoint[24]	?	✓	✓		14.2	24.5	24.1	17.6	38.9	21.0	5.1
	Mask R-CNN*	RX-101	✓			16.1	25.8	26.9	16.5	37.5	25.1	11.2
	MaskJoint*	RX-101	✓	✓		15.9	25.1	26.3	16.0	36.6	23.9	8.3
video-based	MaskTrack R-CNN[50] °	RX-101	✓		✓	15.3	24.3	23.3	17.0	39.3	23.7	9.0
	Ours: VATrack (MT-based)	RX-101	✓	✓	✓	15.9	25.7	24.4	16.9	41.0	24.9	9.7
	QDTrack[38]°	RX-101	✓		✓	17.0	27.1	27.7	18.2	37.2	26.1	12.6
	Ours: VATrack (QD-based)	RX-101	✓	✓	✓	**17.3**	**27.9**	**29.1**	**18.3**	38.6	**28.9**	**12.7**

Table 2: **Visible** instance segmentation **image-level** performance (%) on validation data for image-based methods (blue, results on SAIL-VOS = SAIL-VOS-cut dataset), and for video-based methods (orange, results on SAIL-VOS-cut dataset). Marker $*$ denotes resimulated results. Marker $\circ$ denotes results by adapting the VIS method to the SAIL-VOS-cut dataset. Checkmarks indicate whether the method predicts visible (V) and/or amodal (A) masks. The video-based methods take temporal context (TC) into account, shown by checkmarks. Results have been produced using the ResNext-101 (RX-101) backbone, Hu et al. [24] do not report their backbone (marked by: ?). Best results are in **bold**, second best are underlined.

there is a jump cut between the key frame and the reference frame. Hence, SAIL-VOS-cut consists of shorter video clips from the original SAIL-VOS dataset starting and ending at a jump cut. Jump cuts are identified by an automatic algorithm. This does neither change content nor size of the dataset, but just the video file composition. We report results of all video-based methods on SAIL-VOS-cut, which, due to the design of SAIL-VOS-cut, can be compared to image-based methods. This is because for image-based methods, there is no difference between SAIL-VOS and SAIL-VOS-cut in training and evaluation. For completeness, results on SAIL-VOS of our proposed video evaluation are reported in the supplementary. SAIL-VOS-cut and the underlying generation scripts will be published to ensure reproducibility. Note that as common in the (video) instance segmentation research field, all datasets are split in disjoint training and validation datasets, the latter being used for evaluation [14, 16, 24, 26, 27, 40, 49, 50, 56].

4.2 Metrics

We report metrics on image and video level. As previous methods only evaluate image-based amodal instance segmentation, comparison to the state of the art is only possible with image-level metrics. The supplementary contains a mathematical description of the metrics.
Image level: For image-level evaluation, we use the proposed metrics for SAIL-VOS [24], i.e., average precision (AP) and AP at IoU threshold 50% (AP_{50}). We also distinguish between large (AP^{L}_{50}), medium (AP^{M}_{50}), and small (AP^{S}_{50}) objects, heavily (AP^{H}_{50}) and partially (AP^{P}_{50}) occluded objects. We also report the visible and the amodal set of AP metrics.

Method	Backbone	V	A	vAP	vAP_{50}	vAP^{P}_{50}	vAP^{H}_{50}	vAP^{L}_{50}	vAP^{M}_{50}	vAP^{S}_{50}
AmodalTrack (MT-based)	RX-101		✓	2.4	3.1	3.8	1.7	3.8	1.4	0.4
Ours: VATrack (MT-based)	RX-101	✓	✓	2.3	3.1	3.8	1.7	3.7	1.5	0.3
AmodalTrack (QD-based)	RX-101		✓	13.1	20.5	21.0	10.7	29.4	14.7	**8.9**
Ours: VATrack (QD-based)	RX-101	✓	✓	**14.1**	**22.3**	**22.0**	**12.8**	**32.8**	**15.6**	8.8

Table 3: **Amodal** instance segmentation **video-level** performance (%) on the **SAIL-VOS-cut** validation set. Checkmarks indicate whether the method predicts visible (V) and/or amodal (A) masks. Results have been produced using the ResNext-101 (RX-101) backbone. Best results are in **bold**, second best are underlined.

Method	Backbone	V	A	vAP	vAP_{50}	vAP^{P}_{50}	vAP^{H}_{50}	vAP^{L}_{50}	vAP^{M}_{50}	vAP^{S}_{50}
MaskTrack R-CNN [50]°	RX-101	✓		2.5	3.2	3.8	1.9	4.5	1.7	0.3
Ours: VATrack (MT-based)	RX-101	✓	✓	2.3	3.1	3.8	1.6	4.4	1.4	0.3
QDTrack [38]°	RX-101	✓		13.0	22.4	21.2	14.3	**40.6**	19.0	**9.1**
Ours: VATrack (QD-based)	RX-101	✓	✓	**14.0**	**23.0**	**21.9**	**14.6**	36.4	**21.5**	8.6

Table 4: **Visible** instance segmentation **video-level** performance (%) on the **SAIL-VOS-cut** validation set. Marker ∘ denotes results by adapting the VIS method to the SAIL-VOS-cut dataset. Checkmarks indicate whether the method predicts visible (V) and/or amodal (A) masks. Results have been produced using the ResNext-101 (RX-101) backbone. Best results are in **bold**, second best are underlined.

Video level: For video evaluation, we use the standard metrics from VIS literature [27, 47, 50]: video average precision (vAP) and video average precision at IoU threshold 50% (vAP_{50}). Here, instance predictions are considered over the entire video. More precisely, without sufficient overlap of all predicted masks of an instance with the ground truth in a video, its prediction is not counted as correct. Naturally, high vAP is harder to achieve. For video evaluation we differentiate again between large (AP^{L}_{50}), medium (AP^{M}_{50}), and small (AP^{S}_{50}) objects, as well as heavily (AP^{H}_{50}) and partially (AP^{P}_{50}) occluded objects [24].

4.3 Image-Level Evaluation Results

Here, we report *image-level* results for amodal (Tab. 1) and visible (Tab. 2) instance segmentation, comparing our results to the state of the art on the SAIL-VOS validation data [24]. All methods are trained for 12 epochs on an NVIDIA A100 GPU, while adopting hyperparameters from prior art [11, 15, 38, 50] without further tuning. We set $\lambda_1 = \lambda_2 = 1.0$ in (2) as it is the standard in mmdetection [11] and mmtracking [15]. An ablation on the loss weights as well as details on training and implementation can be found in the supplementary. In Tab. 1 and 2, we distinguish between the blue-highlighted image-based methods (results on SAIL-VOS = SAIL-VOS-cut dataset) and the orange-highlighted video-based methods (results on SAIL-VOS-cut dataset), reporting results for the images of the SAIL-VOS validation dataset, taking into account the temporal context preventing jump cuts.

Amodal instance segmentation: Tab. 1 reports the *amodal* instance segmentation *image-level* performance. We report the image-based state-of-the-art results from Hu et al. [24] for MaskAmodal and MaskJoint. MaskAmodal is Mask R-CNN predicting only amodal masks, while MaskJoint is a Mask R-CNN with an additional amodal mask head to predict both mask types. We report our resimulated results for both methods (∗ in Tab. 1, 2). Hu et al. [24] did not report their backbone. For fair comparison, we chose ResNext-101. We denote the separately trained methods (loss (1)) predicting only the amodal masks in the notation of [24]: AmodalTrack MT-based and QD-based, distinguishing the two tracking methods, i.e., MaskTrack R-CNN (MT-based) and QDTrack (QD-based) tracking methods. The MT-based AmodalTrack performs better than the image-based methods in four of seven metrics (e.g., AP_{50}=25.7% vs. current SOTA 25.6% and AP^{L}_{50}=36.8% vs. current SOTA 35.2%) showing the value of video-based tracking for amodal instance segmentation. Due to the improved tracking method, QD-based AmodalTrack outperforms the MT-

based one in almost all metrics, additionally outperforming the best image-based method MaskJoint*, where AP (AP_{50}) improves by 1.1% (1.8%) absolute.

Finally, still in Tab. 1, VATrack yields further improvements to AmodalTrack. While the MT-based VATrack improves four metrics, QD-based VATrack outperforms its corresponding AmodalTrack in all metrics. *QD-based VATrack provides strongest results over all image-based and video-based methods*, with an AP of 18.3% (MaskJoint*: 16.7%) and an AP_{50} of 28.6% (25.6%), showing the benefit of joint training and video-based tracking. MT-based VATrack performs slightly better than the QD-based VATrack in AP^L_{50}, likely due to the better results of the MT-based AmodalTrack.

Visible instance segmentation: Results for the *image-level visible* evaluation are shown in Tab. 2. Instead of the AmodalMask method, we report results of the image-based method Mask R-CNN as well as the video-based methods MaskTrack R-CNN, and QDTrack for visible segmentation. When comparing the video-based MaskTrack R-CNN to the image-based baselines (top four rows in Tab. 2), we observe that MaskTrack R-CNN only achieves similar results as some metrics are improved while others are slightly worse. Interestingly, this is in contrast to the amodal results in Tab. 1, where we observed clearer improvements by its amodal counterpart MT-based AmodalTrack. We believe that a reason for this result is that the temporal context has much more value for guidance w.r.t. occluded instances when predicting amodal masks than for the prediction of visible masks. Another reason might also be the known trade-off between segmentation and tracking performance [40]. QDTrack outperforms the MaskTrack R-CNN in most metrics due to the improved tracking head and also surpasses the image-based baselines.

Another important observation from Tab. 2 is that the additional prediction of amodal masks in our joint training with VATrack also benefits the visible instance segmentation. More specifically, MT-based VATrack outperforms MaskTrack R-CNN in all but one metric. Also, we observe that the video-based QD-based VATrack as our best (proposed) method improves upon the QDTrack results in all metrics and, additionally, outperforms the image-based baselines in almost all metrics, e.g., AP of 17.3% vs. 16.1% (Mask R-CNN*). Most notably, the VATrack performance for partially (and heavily) occluded objects reaches 29.1% (18.3%) in AP^P_{50} (AP^H_{50}), exhibiting a strong improvement. *The QD-based VATrack method outperforms the image-based state of the art on the SAIL-VOS-cut validation set in all metrics in the image-level amodal evaluation, while providing competitive image-level visible performance leading in almost all metrics.*

4.4 Video-Level Evaluation Results

In this section, we report results on video level for amodal and visible VIS. Here, we cannot present results for single-image approaches, i.e., methods in blue in Tab. 1, 2.

Amodal VIS: Tab. 3 shows the amodal video-level results. MT-based VATrack performs similar to the MT-based AmodalTrack, with quite low performance in both cases due to the more challenging video-level metrics. Further, we note a discrepancy between the results of the MT-based and the QD-based methods, e.g., the $vAP_{50} = 3.1\%$ for MT-based VATrack and MT-based AmodalTrack, while QD-based AmodalTrack achieves a $vAP_{50} = 20.5\%$, and QD-based VATrack even reaches 22.3% due to the improved tracking. Finally, we can confirm the merit of joint training for amodal instance segmentation as QD-based VATrack outperforms the QD-based AmodalTrack in six out of seven metrics.

Visible VIS: Tab. 4 shows the video-level visible results. We observe the same discrepancy between the results of the MT-based methods and the QD-based ones, e.g., the vAP for MaskTrack R-CNN (MT-based VATrack) is 2.5% (2.3%), while QDTrack (QD-based

VATrack) achieves a vAP of 13.0% (14.0%). We can also confirm the value of joint amodal and visible VIS training as both MT-based methods yield a similar performance, while the QD-based VATrack again performs best in five out of seven metrics.

Interestingly, for both amodal and visible video-level evaluation, QD-based VATrack improves performance compared to the respective AmodalTrack (QDTrack) on partially occluded objects (vAP^P_{50}) by 1.0% (0.7%) absolute, and for heavily occluded objects (vAP^H_{50}) by 2.1% (0.3%) absolute, which confirms our results in Tab. 1, 2. *To summarize, our VATrack methods achieve current state-of-the-art results in the video-based evaluation on the SAIL-VOS-cut dataset and can be used for future reference.*

5 Conclusions

In this work, we investigate the task of end-to-end amodal video instance segmentation (VIS) on the SAIL-VOS dataset. Our VATrack method is based on the Mask R-CNN instance segmentation in which we include an amodal mask head to predict amodal and visible masks at the same time. Additionally, we adapt two different tracking heads from VIS methods. The resulting AmodalTrack methods already perform competitively with the state of the art on the SAIL-VOS dataset. However, *the proposed end-to-end QD-based VATrack outperforms the current state of the art on all amodal metrics (e.g., $AP_{50} = 28.6\%$ vs. 25.6%), while also improving almost all visible metrics as well (e.g., AP=17.3% vs. 16.1%).* Finally, we report not only the image-wise metrics on the SAIL-VOS dataset, but also video-based metrics, which can be used as a novel reference to further advance research in amodal VIS.

Acknowledgements: This work results from the project KI Data Tooling (19A20001M) funded by the German Federal Ministry for Economic Affairs and Climate Action (BMWK).

References

[1] Seunghyeok Back, Joosoon Lee, Taewon Kim, Sangjun Noh, Raeyoung Kang, Seongho Bak, and Kyoobin Lee. Unseen Object Amodal Instance Segmentation via Hierarchical Occlusion Modeling. In *Proc. of ICRA*, pages 5085–5092, Philadelphia, PA, USA, May 2022.

[2] Andreas Bär, Jonas Löhdefink, Nikhil Kapoor, Serin John Varghese, Fabian Hüger, Peter Schlicht, and Tim Fingscheidt. The Vulnerability of Semantic Segmentation Networks to Adversarial Attacks in Autonomous Driving: Enhancing Extensive Environment Sensing. *IEEE Signal Processing Magazine*, 38(1):42–52, January 2021.

[3] Daniel Bogdoll, Stefani Guneshka, and J. Marius Zöllner. One Ontology to Rule Them All: Corner Case Scenarios for Autonomous Driving. In *Proc. of ECCV - Workshops*, pages 409–425, Tel Aviv, Israel, October 2022.

[4] Jan-Aike Bolte, Andreas Bär, Daniel Lipinski, and Tim Fingscheidt. Towards Corner Case Detection for Autonomous Driving. In *Proc. of IV*, pages 366–373, Paris, France, June 2019.

[5] Jan-Aike Bolte, Markus Kamp, Antonia Breuer, Silviu Homoceanu, Peter Schlicht, Fabian Hüger, Daniel Lipinski, and Tim Fingscheidt. Unsupervised Domain Adaptation to Improve Image Segmentation Quality Both in the Source and Target Domain. In *Proc. of CVPR - Workshops*, pages 1404–1413, Long Beach, CA, USA, June 2019.

[6] Jasmin Breitenstein and Tim Fingscheidt. Amodal Cityscapes: A New Dataset, its Generation, and an Amodal Semantic Segmentation Challenge Baseline. In *Proc. of IV*, pages 1018–1025, Aachen, Germany, June 2022.

[7] Jasmin Breitenstein, Jan-Aike Termöhlen, Daniel Lipinski, and Tim Fingscheidt. Systematization of Corner Cases for Visual Perception in Automated Driving. In *Proc. of IV*, pages 986–993, Las Vegas, NV, USA, October 2020.

[8] Jasmin Breitenstein, Jonas Löhdefink, and Tim Fingscheidt. Joint Prediction of Amodal and Visible Semantic Segmentation for Automated Driving. In *Proc. of ECCV - Workshops*, pages 633–645, Tel Aviv, Israel, October 2022.

[9] Antonia Breuer, Sven Elflein, Tim Joseph, Jan-Aike Bolte, Silviu Homoceanu, and Tim Fingscheidt. Analysis of the Effect of Various Input Representations for LSTM-Based Trajectory Prediction. In *Proc. of ITSC*, pages 2728–2735, Auckland, NZ, October 2019.

[10] Antonia Breuer, Jana Kirschner, Silviu Homoceanu, and Tim Fingscheidt. Towards Tactical Maneuver Detection for Autonomous Driving Based on Vision Only. In *Proc. of IV*, pages 941–948, Paris, France, June 2019.

[11] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. MMDetection: Open MMLab Detection Toolbox and Benchmark. *arXiv preprint arXiv:1906.07155*, pages 1–13, June 2019.

[12] Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Quai. Vision Transformer Adapter for Dense Predictions. In *Proc. of ICLR*, pages 1–20, Kigali, Rwanda, May 2023.

[13] Bowen Cheng, Maxwell D. Collins, Yukun Zhu, Ting Liu, Thomas S. Huang, Hartwig Adam, and Liang-Chieh Chen. Panoptic-DeepLab: A Simple, Strong, and Fast Baseline for Bottom-Up Panoptic Segmentation . In *Proc. of CVPR*, pages 12475–12485, Seattle, WA, USA, June 2020.

[14] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention Mask Transformer for Universal Image Segmentation. In *Proc. of CVPR*, pages 1290–1299, New Orleans, LO, USA, June 2022.

[15] MMTracking Contributors. MMTracking: OpenMMLab Video Perception Toolbox and Benchmark. https://github.com/open-mmlab/mmtracking, 2020.

[16] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The Cityscapes Dataset for Semantic Urban Scene Understanding. In *Proc. of CVPR*, pages 3213–3223, Las Vegas, NV, USA, June 2016.

[17] Sudeep Fadadu, Shreyash Pandey, Darshan Hegde, Yi Shi, Fang-Chieh Chou, Nemanja Djuric, and Carlos Vallespi-Gonzalez. Multi-View Fusion of Sensor Data for Improved Perception and Prediction in Autonomous Driving. In *Proc. of WACV*, pages 2349–2357, Waikoloa, HI, USA, January 2022.

[18] Tim Fingscheidt, Hanno Gottschalk, and Sebastian Houben, editors. *Deep Neural Networks and Data for Automated Driving: Robustness, Uncertainty Quantification, and Insights Towards Safety*. Springer Nature, Cham, 2022. doi: 10.1007/978-3-031-01233-4. URL https://library.oapen.org/handle/20.500.12657/57375.

[19] Patrick Follmann, Rebecca König, Philipp Härtinger, and Michael Klostermann. Learning to See the Invisible: End-to-End Trainable Amodal Instance Segmentation. In *Proc. of WACV*, pages 1328–1336, Waikoloa Village, HI, USA, January 2019.

[20] Mingqi Gao, Feng Zheng, James J. Q. Yu, Caifeng Shan, Guiguang Ding, and Jungong Han. Deep Learning for Video Object Segmentation: A Review. *Artificial Intelligence Review*, 56(1):457–531, April 2022.

[21] 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 *Proc. of CVPR*, pages 2918–2928, Nashville, TN, USA, June 2021.

[22] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *Proc. of ICCV*, pages 2980–2988, Venice, Italy, October 2017.

[23] Florian Heidecker, Jasmin Breitenstein, Kevin Rösch, Jonas Löhdefink, Maarten Bieshaar, Christoph Stiller, Tim Fingscheidt, and Bernhard Sick. An Application-Driven Conceptualization of Corner Cases for Perception in Highly Automated Driving. In *Proc. of IV*, pages 644–651, Nagoya, Japan, July 2021.

[24] Yuan-Ting Hu, Hong-Shuo Chen, Kexin Hui, Jia-Bin Huang, and Alexander G. Schwing. SAIL-VOS: Semantic Amodal Instance Level Video Object Segmentation – A Synthetic Dataset and Baselines. In *Proc. of CVPR*, pages 3105–3115, Long Beach, CA, USA, June 2019.

[25] Jitesh Jain, Jiachen Li, MangTik Chiu, Ali Hassani, Nikita Orlov, and Humphrey Shi. OneFormer: One Transformer to Rule Universal Image Segmentation. *arXiv preprint arXiv:2211.06220*, pages 1–19, November 2022.

[26] Jitesh Jain, Jiachen Li, MangTik Chiu, Ali Hassani, Nikita Orlov, and Humphrey Shi. OneFormer: One Transformer to Rule Universal Image Segmentation. In *Proc. of CVPR*, pages 1–19, Vancouver, BC, Canada, June 2023.

[27] Lei Ke, Xia Li, Martin Danelljan, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Prototypical Cross-Attention Networks for Multiple Object Tracking and Segmentation. In *Proc. of NeurIPS*, pages 1192–1203, Virtual, December 2021.

[28] Lei Ke, Yu-Wing Tai, and Chi-Keung Tang. Deep Occlusion-Aware Instance Segmentation with Overlapping BiLayers. In *Proc. of CVPR*, pages 4019–4028, Nashville, TN, USA, June 2021.

[29] Ke Li and Jitendra Malik. Amodal Instance Segmentation. In *Proc. of ECCV*, pages 677–693, Amsterdam, The Netherlands, October 2016.

[30] Lihuan Li, Maurice Pagnucco, and Yang Song. Graph-based Spatial Transformer with Memory Replay for Multi-future Pedestrian Trajectory Prediction. In *Proc. of CVPR*, pages 2231–2241, New Orleans, LO, USA, June 2022.

[31] Huan Ling, David Acuna, Karsten Kreis, Seung Wook Kim, and Sanja Fidler. Variational Amodal Object Completion. In *Proc. of NeurIPS*, pages 16246–16257, Vancouver, BC, Canada, December 2020.

[32] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei, and Baining Guo. Swin Transformer V2: Scaling Up Capacity and Resolution. In *Proc. of CVPR*, pages 12009–12019, New Orleans, LO, USA, June 2022.

[33] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. In *Proc. of CVPR*, pages 3431–3440, Boston, MA, USA, June 2015.

[34] Albert Michotte, Georges Thinès, and Geneviève Crabbé. Amodal Completion of Perceptual Structures. *Michotte's experimental phenomenology of perception*, 1:140–167, January 1991.

[35] Rohit Mohan and Abhinav Valada. EfficientPS: Efficient Panoptic Segmentation. *International Journal of Computer Vision*, 129:1551 – 1579, February 2020.

[36] Bence Nanay. The Importance of Amodal Completion in Everyday Perception. *i-Perception*, 9(4):1–16, July 2018.

[37] Khoi Nguyen and Sinisa Todorovic. A Weakly Supervised Amodal Segmenter with Boundary Uncertainty Estimation. In *Proc. of ICCV*, pages 2995–3003, Virtual, October 2021.

[38] Jiangmiao Pang, Linlu Qiu, Xia Li, Haofeng Chen, Qi Li, Trevor Darrell, and Fisher Yu. Quasi-Dense Similarity Learning for Multiple Object Tracking. In *Proc. of CVPR*, pages 164–173, June 2021.

[39] Pulak Purkait, Christopher Zach, and Ian D. Reid. Seeing Behind Things: Extending Semantic Segmentation to Occluded Regions. In *Proc. of IROS*, pages 1998–2005, Macau, SAR, China, November 2019.

[40] Jiyang Qi, Yan Gao, Yao Hu, Xinggang Wang, Xiaoyu Liu, Xiang Bai, Serge Belongie, Alan Yuille, Philip HS Torr, and Song Bai. Occluded Video Instance Segmentation: A Benchmark. *International Journal of Computer Vision*, 130(8):2022–2039, June 2022.

[41] Lu Qi, Li Jiang, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Amodal Instance Segmentation With KINS Dataset. In *Proc. of CVPR*, pages 3014–3023, Long Beach, CA, USA, June 2019.

[42] N Dinesh Reddy, Robert Tamburo, and Srinivasa Narasimhan. WALT: Watch And Learn 2D Amodal Representation using Time-lapse Imagery. In *Proc. of CVPR*, pages 9356–9366, New Orleans, LA, USA, June 2022.

[43] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You Only Look Once: Unified, Real-Time Object Detection. In *Proc. of CVPR*, pages 779–788, Las Vegas, NV, USA, June 2016.

[44] Hwanjun Song, Deqing Sun, Sanghyuk Chun, Varun Jampani, Dongyoon Han, Byeongho Heo, Wonjae Kim, and Ming-Hsuan Yang. ViDT: An Efficient and Effective Fully Transformer-based Object Detector. In *Proc. of ICLR*, pages 1–18, virtual, April 2022.

[45] Yihong Sun, Adam Kortylewski, and Alan Yuille. Amodal Segmentation through Out-of-Task and Out-of-Distribution Generalization with a Bayesian Model. In *Proc. of CVPR*, pages 1215–1224, New Orleans, LA, USA, June 2022.

[46] Leon Amadeus Varga, Benjamin Kiefer, Martin Messmer, and Andreas Zell. SeaDronesSee: A Maritime Benchmark for Detecting Humans in Open Water. In *Proc. of WACV*, pages 2260–2270, Waikoloa, HI, USA, January 2022.

[47] Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, Baoshan Cheng, Hao Shen, and Huaxia Xia. End-to-End Video Instance Segmentation with Transformers. In *Proc. of CVPR*, pages 8741–8750, Nashville, TN, USA, June 2021.

[48] Sanghyun Woo, Shoubhik Debnath, Ronghang Hu, Xinlei Chen, Zhuang Liu, In So Kweon, and Saining Xie. ConvNeXt V2: Co-designing and Scaling ConvNets with Masked Autoencoders. *arXiv preprint arXiv:2301.00808*, pages 1–15, January 2023.

[49] Junfeng Wu, Qihao Liu, Yi Jiang, Song Bai, Alan Yuille, and Xiang Bai. In Defense of Online Models for Video Instance Segmentation. In *Proc. of ECCV*, pages 588–605, Tel Aviv, Israel, October 2022.

[50] Linjie Yang, Yuchen Fan, and Ning Xu. Video Instance Segmentation. In *Proc. of ICCV*, pages 5188–5197, Seoul, Korea, October 2019.

[51] Jian Yao, Yuxin Hong, Chiyu Wang, Tianjun Xiao, Tong He, Francesco Locatello, David Wipf, Yanwei Fu, and Zheng Zhang. Self-supervised Amodal Video Object Segmentation. In *Proc. of NeurIPS*, pages 1–13, New Orleans, LA, USA, November 2022.

[52] Yuhui Yuan, Xilin Chen, and Jingdong Wang. Object-Contextual Representations for Semantic Segmentation. In *Proc. of ECCV*, pages 173–190, Glasgow, United Kingdom, August 2020.

[53] Xunli Zeng, Xiaoli Liu, and Jianqin Yin. Amodal Segmentation Just Like Doing a Jigsaw. *Applied Sciences*, 12(8):1–10, April 2022.

[54] Zitong Zhan, Daniel McKee, and Svetlana Lezebnik. Robust Online Video Instance Segmentation with Track Queries. *arXiv preprint arXiv:2211.09108*, pages 1–12, November 2022.

[55] Ziheng Zhang, Anpei Chen, Ling Xie, Yu Jingyi, and Shenghua Gao. Learning Semantics-aware Distance Map with Semantics Layering Network for Amodal Instance Segmentation. In *Proc. of MM*, pages 2124–2132, Nice, France, October 2019.

[56] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene Parsing through ADE20K Dataset. In *Proc. of CVPR*, pages 633–641, Honulu, HI, USA, July 2017.

[57] Tianfei Zhou, Fatih Porikli, David J Crandall, Luc Van Gool, and Wenguan Wang. A Survey on Deep Learning Technique for Video Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–20, November 2022.

[58] Hongru Zhu, Peng Tang, Alan Yuille, Soojin Park, and Jeongho Park. Robustness of Object Recognition under Extreme Occlusion in Humans and Computational Models. In *Proc. of CogSci*, pages 3213–3219, Montreal, QC, Canada, July 2019.

[59] Yan Zhu, Yuandong Tian, Dimitris Metaxas, and Piotr Dollár. Semantic Amodal Segmentation. In *Proc. of CVPR*, pages 1464–1472, Honolulu, HI, USA, July 2017.