

Foundation Models for Amodal Video Instance Segmentation in Automated Driving

Jasmin Breitenstein¹, Franz Jünger¹, Andreas Bär¹, and Tim Fingscheidt¹

Institute for Communications Technology
Technische Universität Braunschweig

{j.breitenstein, f.juenger, andreas.baer, t.fingscheidt}@tu-bs.de

Abstract. In this work, we study amodal video instance segmentation for automated driving. Previous works perform amodal video instance segmentation relying on methods trained on entirely labeled video data with techniques borrowed from standard video instance segmentation. Such amodally labeled video data is difficult and expensive to obtain and the resulting methods suffer from a trade-off between instance segmentation and tracking performance. To largely solve this issue, we propose to study the application of foundation models for this task. More precisely, we exploit the extensive knowledge of the Segment Anything Model (SAM), while fine-tuning it to the amodal instance segmentation task. Given an initial video instance segmentation, we sample points from the visible masks to prompt our amodal SAM. We use a point memory to store those points. If a previously observed instance is not predicted in a following frame, we retrieve its most recent points from the point memory and use a point tracking method to follow those points to the current frame, together with the corresponding last amodal instance mask. This way, while basing our method on an amodal instance segmentation, we nevertheless obtain video-level amodal instance segmentation results. Our resulting **S-AModal** method achieves state-of-the-art results in amodal video instance segmentation while resolving the need for amodal video-based labels. Code for **S-AModal** is available at <https://github.com/ifnspaml/S-AModal>.

1 Introduction

For safe automated driving, environment perception is required to perform at least on par with human drivers [15]. Recently, foundation models for environment perception have been introduced. Such models have been trained on a broad amount of data to perform their respective perception task on a wide range of data showcasing impressive zero-shot generalizability. A famous example for such foundation models is the Segment Anything Model (SAM) [25] which can be prompted via points, bounding boxes, or text, to segment instances in a zero-shot fashion in a wide range of images. In contrast to this type of perception, one important human ability is amodal perception allowing us to perceive the full shape of occluded objects. This ability is typically not found in machine learning perception methods, and, even worse, performance drastically drops

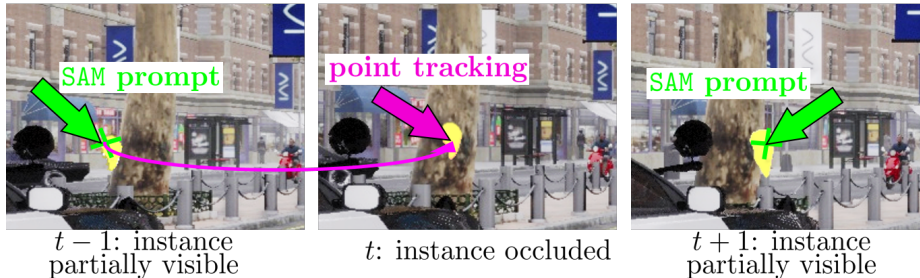


Fig. 1: In a video sequence, when an instance is (partially) visible (time step $t - 1$), we extract points (green arrow) from the predicted instance mask to prompt an amodal SAM method to generate an amodal mask (yellow). The corresponding points are stored, and if the instance is not visible (time step t), we track the previous points to the next frame (t), transferring the previous amodal mask to the next frame (yellow, purple arrow). Once the instance reappears (time step $t + 1$), we prompt the amodal SAM method again (green arrow).

when confronted with occlusions. However, handling occlusions is of high importance for safety in automated driving. Without a safe treatment of occlusions, one risks fatal accidents of the automated vehicle affecting both vehicle occupants and other traffic participants. Amodal segmentation methods offer a way to do this by segmenting not only what is visible, but the full shape of objects in a scene, thus, providing information about what is currently occluded. While there exist many methods working towards amodal segmentation on single images — and excelling at it [1–3, 16, 23, 29, 34], they are limited in the occlusion types they can resolve: Only partial occlusions can be identified and segmented on a single image basis. For occlusions that arise temporarily leading to almost no visible object parts or even to a totally occluded object, amodal segmentation methods working on a single image basis cannot rely on sufficient information. Here, recent new approaches have been introduced and investigated on the task of amodal video instance segmentation (VIS) [5, 50]. Amodal VIS builds upon the terminology of standard (visible) VIS and aims to track and segment the full shape on an instance throughout all frames of a video. Recent approaches have leveraged a full end-to-end training of amodal VIS [5], or investigated a self-supervised training approach based on the results of a VIS method on the considered data [50]. In contrast, our proposed **S-AModal** method relies during training on image-level amodal labels to fine-tune a foundation model to the task of (image-based) amodal segmentation. We build upon the highly generalizing pre-trained SAM [25] using an adaptive fine-tuning approach to facilitate the prediction of amodal masks. To extend this to videos, during inference a pre-trained VIS method provides point prompts for our amodal SAM method. Additionally, we exploit a point tracking method to provide point prompts for occluded instances by relying on a point memory to store points of previously observed instances. This high-level operation of our **S-AModal** method is illustrated in Figure 1. Whenever an instance is at least partially visible and predicted by the VIS method (yellow, left and right images in Figure 1, time instants $t - 1$, $t + 1$),

we extract points from the visible mask to prompt the SAM method fine-tuned on amodal data (highlighted by a green arrow). In Figure 1 one example point prompt is visualized by the green arrow and cross. If a previously predicted instance is not predicted for the current frame (purple, middle image of Figure 1), we apply a point tracking method to track the previous ($t - 1$) point prompt to the current frame t , and we move the amodal mask along the point trajectory. A reappearing instance is identified by the VIS method, so in this case the SAM method can again be prompted. Our amodal VIS method based on these foundation models is able to perform state-of-the-art amodal VIS *without* relying on amodal VIS labels in training.

Our contributions are as follows: First, we provide a fine-tuning strategy to leverage SAM for amodal segmentation. Second, we show that our proposed S-AModal method achieves state-of-the-art results in image-level amodal segmentation. Third, we are the first to apply such a foundation model to the task of amodal VIS, again achieving state-of-the-art results in this task. The remainder of this work is structured as follows: In Section 2, we review the related work w.r.t. SAM [25], point tracking, and amodal segmentation. Section 3 describes our proposed method in detail. In Section 4, we describe the experimental setup and report our results in Section 5. Finally, we conclude with Section 6.

2 Related Work

Here, we review works related to SAM, point tracking, and amodal segmentation. **Segment Anything Model (SAM)** [25] was first introduced as a foundation model that solves the task of promptable segmentation. It can be prompted using text, bounding boxes or points to produce a segmentation mask accordingly. SAM has been trained on an extensive amount of data to perform promptable segmentation in a zero-shot manner on many different types of images.

Adapters have been proposed to allow SAM to, e.g., work well on high-quality images [24, 47], or, in general, on underperformed scenes [7, 40, 45, 46]. The adapters follow fine-tuning strategies, so only the decoder and the prompt encoder [40] or added adapter layers of the encoder [7] have to be fine-tuned. Most related work about fine-tuning SAM is related to underperformed scenes, e.g., in the medical field [7, 40, 45]. In contrast, we investigate SAM not only on a different image domain, but also tailor it to the new task of amodal segmentation.

Additionally, strategies have been proposed to extend the SAM capabilities to video sequences [36]. Rajic et al. [36] perform video segmentation by applying a point tracking method to points of a query mask in the first frame to track it through an entire video sequence, Yang et al. [49] use SAM to improve the segmentation of a VIS method. Cheng et al. [8] extend SAM to track objects via text prompts. Recently, SAM 2 [37], the successor of SAM, has been introduced adding video segmentation capabilities to SAM. In this work, we build upon the benefits of the above by following a combination of SAM with VIS methods to perform the tracking and segmentation task. However, to resolve occlusions, we additionally add a point tracking method to identify points for occluded masks.

Point Tracking has first been introduced and defined by Doersch et al. [11], naming it the “tracking any point” (TAP) task. This task focuses on predicting the pixel positions of a given point in all subsequent frames of a video sequence, as well as identifying its occlusion state. Harley et al. [19] develop PIPs, one of the initial methods for the TAP task. Although PIPs can trace points through temporal occlusions lasting up to eight frames, it cannot account for the occlusion state of the points themselves [19]. Doersch et al. [11] address this limitation by introducing the TAP-Vid benchmark consisting of labelled real and synthetic data for the TAP task, enabling the end-to-end point tracking model TAP-Net. The TAPIR method [12] combines the advantages of both TAP-Net and PIPs. Another enhancement is the PIPs++ method by Zheng et al. [51]. It extends PIPs for improved long-term tracking. All the methods described above focus on tracking points independently, neglecting any potential correlations between them. Conversely, **OmniMotion** [44] aims to achieve globally consistent motion by optimizing volumetric representations for individual videos. Similarly, **CoTracker** [22] tracks points jointly without relying on specific per-video optimization. In this work, we employ a point tracking method to identify the location of points belonging to an occluded instance which is precisely what all aforementioned methods excel in. While any point tracking method can be used for our purpose, we choose to apply the **CoTracker** method [22] due to its performance and ease of use.

Amodal Segmentation describes the task of segmenting the full shape of an object even in the presence of occlusions. First methods investigate predicting an amodal mask given the visible mask and the input image [26, 52]. This has been extended to instance segmentation methods predicting the amodal instead of the visible mask directly from the input image [16, 21, 29, 32, 34, 38, 42]. Amodal semantic segmentation methods apply grouping and multi-task training to look behind occlusions [4, 6, 33]. Recently, amodal video instance segmentation (VIS) has been investigated more extensively, considering both end-to-end supervised training of VIS [5], and self-supervised approaches based on the visible masks [50]. While the end-to-end approach requires amodal video labels which are expensive and difficult to obtain, the self-supervised approach requires visible masks to be predicted. In this sense, our approach is closest to the self-supervised approach, termed **SAVOS** [50], as we strive towards amodal VIS without relying on video-based labels. **SAVOS** takes as input the images, the visible instance masks and the optical flow between two frames. In contrast, we build our method on top of an image-based amodal segmentation method, where datasets for training are available. We rely on point tracking methods to track points of instances through the sequence which diminishes the need for optical flow. Moreover, while **SAVOS** makes its prediction in an offline fashion, i.e., based on the entire video, our proposed method can operate in an online fashion.

One challenge in amodal segmentation is the availability of real datasets, as amodal labeling of real data is associated with high costs, especially for videos. However, on image level, many datasets are available: The KINS dataset [34] based on the KITTI dataset [17], the COCOA dataset [52] based on COCO [28], D2S [16], KITTI-360-APS [31] based on KITTI-360 [27], and Amodal Cityscapes

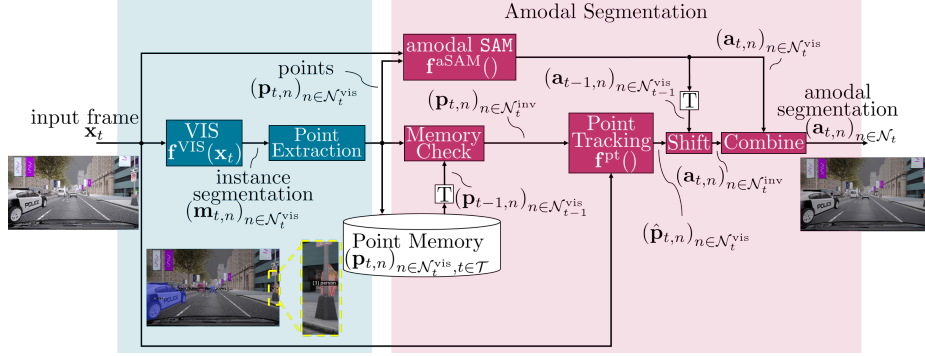


Fig. 2: Overview of our **S-Amodal** method: Given an input frame \mathbf{x}_t , it predicts amodal instance segmentation masks $(\mathbf{a}_{t,n})_{n \in \mathcal{N}_t}$. First, a VIS method provides visible instance masks $(\mathbf{m}_{t,n})_{n \in \mathcal{N}_t^{\text{vis}}}$, from which we extract points $(\mathbf{p}_{t,n})_{n \in \mathcal{N}_t^{\text{vis}}}$. These points prompt our amodal SAM method to produce amodal instance masks $(\mathbf{a}_{t,n})_{n \in \mathcal{N}_t^{\text{vis}}}$. Points are stored to track for occlusions, helping to update previous masks $(\mathbf{a}_{t-1,n})_{n \in \mathcal{N}_{t-1}^{\text{vis}}}$ to $(\mathbf{a}_{t,n})_{n \in \mathcal{N}_t^{\text{inv}}}$. Final amodal masks per frame $(\mathbf{a}_{t,n})_{n \in \mathcal{N}_t}$ combine $(\mathbf{a}_{t,n})_{n \in \mathcal{N}_t^{\text{inv}}}$ and $(\mathbf{a}_{t,n})_{n \in \mathcal{N}_t^{\text{vis}}}$. We denote delay units by “T”.

[4] based on Cityscapes [9]. For videos, two large-scale synthetic video datasets with amodal video-level labels exist: SAIL-VOS [21] is a synthetic dataset collected in the GTA V game. AmodalSynthDrive [39] is another synthetic dataset of automated driving sequences collected in the CARLA engine [14]. Additionally, on real data, Yao et al. [50] match amodal annotations of the KINS dataset with videos of KITTI to obtain amodal annotations on single frames of the videos, termed KINS-car. Note that in contrast to the synthetic datasets, for KINS-car no full video annotations are available. As we are interested in amodal video instance segmentation for automated driving, we investigate the performance of our method on AmodalSynthDrive and KINS-car. This way, we obtain evaluation results on videos. For training, only image annotations are needed.

3 Proposed S-Amodal Method

Our proposed **S-Amodal** method performs amodal VIS based on point-prompting a fine-tuned SAM network which we term amodal SAM \mathbf{f}^{aSAM} . We consider a video as sequence of frames $\mathbf{x}_t \in [0, 1]^{H \times W \times C}$ for $t \in \mathcal{T} = \{1, \dots, T\}$ with T being the video length. The overall method is depicted in Figure 2. We apply a standard online VIS method \mathbf{f}^{VIS} to the input frame \mathbf{x}_t to obtain the corresponding visible segmentation masks $\mathbf{f}^{\text{VIS}}(\mathbf{x}_t) = (\mathbf{m}_{t,n})_{n \in \mathcal{N}_t^{\text{vis}}} \in \{0, 1\}^{H \times W \times N_t}$ for all instances $n \in \mathcal{N}_t^{\text{vis}} = \{1, \dots, N_t^{\text{vis}}\}$ with N_t^{vis} being the number of instances visible in frame t . The total number of visible and occluded instances in frame t is denoted as $N_t = N_t^{\text{vis}} + N_t^{\text{inv}}$, and the total number of instances per video sequence is accordingly denoted as N . We define a point selected from the instance segmentation $\mathbf{m}_{t,n}$ as $p_{t,n,k} \in \{1, \dots, H \cdot W\}$, i.e., a pixel index where the visible

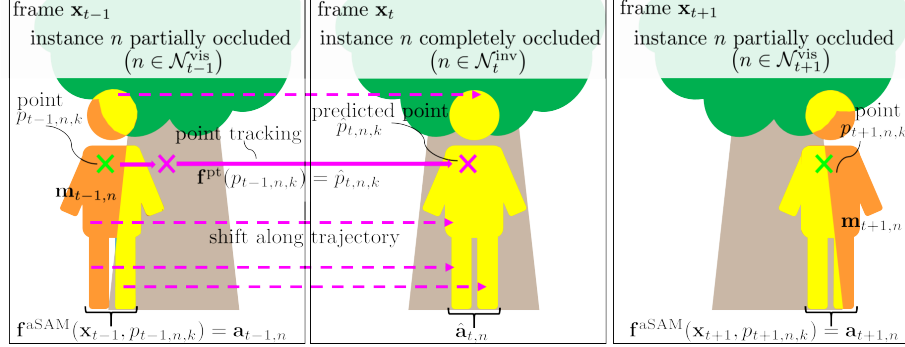


Fig. 3: Schematic view of a video sequence $\mathbf{x}_{t-1}, \mathbf{x}_t, \mathbf{x}_{t+1}$ and an instance n which is partially visible at $t-1$ at $t+1$, but fully occluded at t . For frames \mathbf{x}_{t-1} and \mathbf{x}_{t+1} , the points $p_{t-1,n,k}$ and $p_{t+1,n,k}$ of the predicted visible instance masks $\mathbf{m}_{t-1,n}$ and $\mathbf{m}_{t+1,n}$, are used to prompt the amodal SAM model to obtain $\mathbf{a}_{t-1,n}$ and $\mathbf{a}_{t+1,n}$, respectively. For frame \mathbf{x}_t , we apply point tracking to obtain the predicted point $\hat{p}_{t,n,k}$ and shift the amodal mask along this trajectory to $\hat{\mathbf{a}}_{t,n}$.

instance mask is predicted. From each instance segmentation $\mathbf{m}_{t,n}$, we extract a point K-tuple (henceforth referred to as points) $\mathbf{p}_{t,n} = (p_{t,n,k}) \in \{1, \dots, H \cdot W\}^K$ with K being the number of points used to prompt the amodal SAM method \mathbf{faSAM} . The amodal SAM method then provides the amodal instance segmentation $\mathbf{faSAM}(\mathbf{x}_t, \mathbf{p}_{t,n}) = \mathbf{a}_{t,n} \in \{0, 1\}^{H \times W}$, $n \in \mathcal{N}_t^{\text{vis}}$. Note that both, point extraction and prompting of amodal SAM, are done for all visible instances with index $n \in \mathcal{N}_t^{\text{vis}}$. Additionally, we store the extracted points $\mathbf{p}_{t,n}$ in a point memory, which we access for each frame to check for instances $n \in \mathcal{N}_t$ that are not visible (“memory check” in Figure 2). We term this subset of fully occluded (i.e., invisible) instances at frame t as $\mathcal{N}_t^{\text{inv}}$. The point memory contains all previously observed point K-tuples per previously observed instance, i.e., $(\mathbf{p}_{\tau,n})_{\tau \in \mathcal{T}_t, n \in \mathcal{N}_\tau}$ where $\mathcal{T}_t = \{1, \dots, t\}$, and t being the current frame index. At frame t , a memory retrieval checks whether any points in the K-tuple $\mathbf{p}_{t-1,n}$ are contained in the memory, for which the corresponding instance n has not been predicted by the VIS method \mathbf{f}^{VIS} for frame t , i.e., $n \in \mathcal{N}_t^{\text{inv}}$. A point tracking method \mathbf{f}^{Pt} then tracks the as occluded identified instance points $(\mathbf{p}_{t-1,n})_{n \in \mathcal{N}_t^{\text{inv}}}$, towards the current frame \mathbf{x}_t , to obtain predicted points $\mathbf{f}^{\text{Pt}}(\mathbf{x}_t, (\mathbf{p}_{t-1,n})_{n \in \mathcal{N}_t^{\text{inv}}}) = (\hat{\mathbf{p}}_{t,n})_{n \in \mathcal{N}_t^{\text{inv}}}$.

Let’s look into details: Given a point $p_{t-1,n,k} \in \{1, \dots, H \cdot W\}$ corresponding to the 2D pixel position $(h_{t-1,n}, w_{t-1,n})$, with $h_{t-1,n} \in \{1, \dots, H\}$, $w_{t-1,n} \in \{1, \dots, W\}$, the goal of the point tracking method is to determine the pixel position of that point at the next time step t . Moreover, a binary occlusion score $o_{t,n,k} \in \{0, 1\}$ predicts whether the point is actually visible or occluded by another object. Similar to the optical flow, these pixel positions can be represented as 2D $(\Delta h_{t-1,n}, \Delta w_{t-1,n})$ displacement vectors at time step $t-1$, where $\Delta h_{t-1,n} \in \mathbb{R}$ describes the vertical offset with respect to the source position and $\Delta w_{t-1,n} \in \mathbb{R}$ represents the horizontal offset. In summary, for each point $p_{t-1,n,k}$ that should be tracked, a point tracking method \mathbf{f}^{Pt} predicts the point $\hat{p}_{t,n,k}$ ac-

ording to $\mathbf{f}^{\text{pt}}(\mathbf{x}_t, (\mathbf{p}_{t-1,n})) = (\hat{\mathbf{p}}_{t,n},)$ and the occlusion score $o_{t,n,k}$. From the predicted point $\hat{p}_{t,n,k}$, we calculate vertical offset $\Delta h_{t-1,n}$ and the horizontal offset $\Delta w_{t-1,n}$, which we then use to shift the amodal mask $\mathbf{a}_{t-1,n}$ to time step t , i.e., $\mathbf{a}_{t,n}$. This way we obtain a trajectory from the point $p_{t-1,n,k}$, $n \in \mathcal{N}_t^{\text{inv}}$, to its position in the next frame, i.e., $\hat{p}_{t,n,k}$.

Figure 3 shows this for a schematic example of an occluded person. In image frame \mathbf{x}_{t-1} , the person is partially visible and amodal SAM \mathbf{f}^{aSAM} is prompted using a point $p_{t-1,n,k}$ (green) from the visible mask $\mathbf{m}_{t-1,n}$ (orange) to obtain the full amodal mask for instance n , i.e., $\mathbf{f}^{\text{aSAM}}(\mathbf{x}_{t-1}, p_{t-1,n,k}) = \mathbf{a}_{t-1,n}$. Using a VIS method for tracking and segmentation allows identifying a fully occluded instance as a missing segmentation mask in the output. As this instance n is no longer visible in frame \mathbf{x}_t , we use a point tracking method \mathbf{f}^{pt} to predict the location of $p_{t-1,n,k}$ in \mathbf{x}_t , i.e., obtaining $\mathbf{f}^{\text{pt}}(\mathbf{x}_t, p_{t-1,n,k}) = \hat{p}_{t,n,k}$ (illustrated in pink in Figure 3). This gives a trajectory as visualized by pink arrows in Figure 3. The amodal mask is moved along this trajectory to obtain the amodal mask $\hat{\mathbf{a}}_{t,n}$. If the number of points in the K-tuple is $K > 1$, we obtain a trajectory per point from the point tracking method.

We use the predicted points $(\hat{\mathbf{p}}_{t,n})_{n \in \mathcal{N}_t^{\text{inv}}}$ to shift the corresponding previously observed amodal instance masks $(\hat{\mathbf{a}}_{t-1,n})_{n \in \mathcal{N}_t^{\text{inv}}}$ to the current frame \mathbf{x}_t along the trajectory, and obtain the amodal masks for instances that were not detected by the VIS method in frame \mathbf{x}_t , i.e., $(\hat{\mathbf{a}}_{t,n})_{n \in \mathcal{N}_t^{\text{inv}}}$. The simple shifting operation translates the previous amodal mask $\mathbf{a}_{t-1,n}$ to the current frame by adding the displacement from the previous point $p_{t-1,n,k}$ to the predicted point $\hat{p}_{t,n,k}$ to all coordinates of the amodal mask $\mathbf{a}_{t-1,n}$. Note that if $K > 1$, we calculate the displacement as the average of the predicted points. Combining the amodal masks from the point tracking branch $(\hat{\mathbf{a}}_{t,n})_{n \in \mathcal{N}_t^{\text{inv}}}$ and the amodal SAM branch $(\hat{\mathbf{a}}_{t,n})_{n \in \mathcal{N}_t^{\text{vis}}}$ gives the full set of amodal instance masks for frame \mathbf{x}_t , $(\hat{\mathbf{a}}_{t,n})_{n \in \mathcal{N}_t}$.

Our approach is independent of the choice of the VIS method \mathbf{f}^{VIS} and of the point tracking method \mathbf{f}^{pt} , so any method in these fields can be chosen. Our amodal SAM is fine-tuned on the amodal ground truth on images. For fine-tuning SAM on amodal data, we follow the approach of Chen et al. [7], keeping the image encoder fixed, while we add additional adapter layers to the image encoder and fine-tune the decoder. In addition, we keep the fixed prompt encoder to insert point prompts into the network, see more details in the Supplementary Sec. A.

4 Experimental Setup

We consider two datasets: AmodalSynthDrive \mathcal{D}_{ASD} [39] and KINS-car $\mathcal{D}_{\text{Kcar}}$ [50]. As \mathcal{D}_{ASD} only provides training ($\mathcal{D}_{\text{ASD}}^{\text{train}}$) and validation ($\mathcal{D}_{\text{ASD}}^{\text{val}}$) data with ground-truth annotations, we report results on the validation set. For $\mathcal{D}_{\text{Kcar}}$, in addition to training ($\mathcal{D}_{\text{Kcar}}^{\text{train}}$) and validation ($\mathcal{D}_{\text{Kcar}}^{\text{val}}$) data, labeled test data $\mathcal{D}_{\text{Kcar}}^{\text{test}}$ exists on which our results are reported. Both are video-based datasets, while $\mathcal{D}_{\text{Kcar}}$ only provides an image-based ground truth and no tracking information.

On *image level*, we report mean intersection over union (mIoU), average precision (AP), and AP₅₀ according to the COCO evaluation [28]. Additionally,

we report derivatives of these metrics for small, medium, and large objects as well as partial and heavy occlusions following standards in literature [5, 21, 28].

On *video level*, we report video average precision (vAP) and its derivatives metrics following the SAIL-VOS dataset benchmark [21], and Breitenstein et al. [5]. Definitions are recapitulated in the Supplementary Sec. B. Note that for our proposed method, the instance class prediction is simply derived from the underlying VIS method. Hence, all our metrics are reported in a class-agnostic setting. The quality of class prediction is not influenced by the investigated methods and gives no indication about their performance. In this setting, class predictions do not influence whether a prediction is defined as true positive, false positive or false negative, instead, this decision is only based on the IoU between predicted and ground-truth mask.

We compare our results with SAVOS [50], which is closest to our method as it also does not require amodal video-based labels during training. SAVOS performs self-supervised amodal VIS on top of a visible instance segmentation. It is trained without amodal labels using optical flow and the visible instance masks. We train SAVOS [50] following the original setting on both datasets.

On $\mathcal{D}_{\text{Kcar}}$, we use PointTrack (PT) [48] as VIS method with the same checkpoint as SAVOS [50]. For \mathcal{D}_{ASD} , we choose GenVIS [20], one of the current top VIS methods. We train GenVIS on $\mathcal{D}_{\text{ASD}}^{\text{train}}$ using the full image resolution. It achieves an AP of 30.38%, an AP₅₀ of 43.83%, a vAP of 16.36% and a vAP₅₀ of 23.13% on $\mathcal{D}_{\text{ASD}}^{\text{val}}$. Detailed results are reported in Supplementary Sec. C. We also replace the VIS method in Figure 2 by the ground truth (GT) to cancel out the VIS performance, giving the methods access to the ground-truth visible masks.

To obtain our amodal SAM \mathbf{f}^{aSAM} , SAM with the added adapters is fine-tuned on the training data of both datasets using a batchsize of 1 on one NVidia A100 GPU for 20 epochs. Due to memory constraints, we use the vision transformer ViT-B backbone [13] with pre-trained weights as provided by the original SAM codebase [25]. We use the AdamW optimizer and a start learning rate of 0.00001. The learning rate is multiplied by 0.1 every 10 epochs. As loss function, we use a combination of focal loss and dice loss as is common for fine-tuning SAM [7, 40, 45]. We report results as mean across three inference runs. If not stated otherwise, we prompt amodal SAM \mathbf{f}^{aSAM} with $K = 1$ point. As point tracking method, we use CoTracker [22] with the given checkpoints. No further fine-tuning is necessary.

5 Experiments and Discussion

Quantitative Results: First, we regard the amodal image-level results on both datasets. Table 1 reports the results on the validation data of AmodalSynthDrive ($\mathcal{D}_{\text{ASD}}^{\text{val}}$) and on the test data of KINS-car ($\mathcal{D}_{\text{Kcar}}^{\text{test}}$). We compare our results against the SAVOS method [50]. We observe that our **S-AModal** method outperforms SAVOS on both datasets in almost all metrics. Especially, on $\mathcal{D}_{\text{Kcar}}^{\text{test}}$ with the PointTrack (PT) method [48], the average precision (AP) of **S-AModal** on heavily occluded objects (AP₅₀^H), **S-AModal** (35.08%) excels the SAVOS result (30.72%) by 4.36% absolute. We observe similar performance improvements on $\mathcal{D}_{\text{ASD}}^{\text{val}}$ using the VIS method GenVIS [20]: **S-AModal** leads to better performance

Data	Method	VIS	AP	AP ₅₀	AP ₅₀ ^P	AP ₅₀ ^H	AP ₅₀ ^L	AP ₅₀ ^M	AP ₅₀ ^S
$\mathcal{D}_{\text{ASD}}^{\text{val}}$	SAVOS [50]	GV	7.41	13.19	16.13	2.34	12.52	16.24	6.91
	S-AModal (ours)	GV	21.59	35.00	38.50	10.43	66.71	50.23	14.75
	SAVOS [50]	GT	50.89	76.26	81.89	41.92	91.49	84.21	41.62
	S-AModal (ours)	GT	46.91	73.86	80.85	43.79	97.15	82.41	60.13
$\mathcal{D}_{\text{Kcar}}^{\text{test}}$	SAVOS [50]	PT	40.50	61.80	77.89	30.72	96.38	89.83	39.38
	S-AModal (ours)	PT	41.08	74.28	78.23	35.08	97.40	85.25	57.13

Table 1: Amodal **image-level** instance segmentation metrics on $\mathcal{D}_{\text{ASD}}^{\text{val}}$ and on $\mathcal{D}_{\text{Kcar}}^{\text{test}}$ using GenVIS (GV) [20], the ground-truth visible masks (GT), and PointTrack (PT) [48] as VIS methods, respectively. Best results in **bold**.

Data	Method	VIS	vAP	vAP ₅₀	vAP ₅₀ ^P	vAP ₅₀ ^H	vAP ₅₀ ^L	vAP ₅₀ ^M	vAP ₅₀ ^S
$\mathcal{D}_{\text{ASD}}^{\text{val}}$	SAVOS [50]	GV	1.27	3.95	12.46	0.16	7.56	0.16	0.43
	S-AModal (ours)	GV	2.96	6.03	13.44	0.63	27.91	4.12	1.64
	SAVOS [50]	GT	20.53	38.39	45.54	31.39	79.60	73.46	14.77
	S-AModal (ours)	GT	30.88	56.60	67.03	45.57	98.91	80.46	29.10

Table 2: Amodal **video-level** instance segmentation metrics on $\mathcal{D}_{\text{ASD}}^{\text{val}}$ using GenVIS (GV) [20] and ground-truth visible masks (GT) as VIS methods. Best results in **bold**.

in comparison to SAVOS, even increasing AP by 14.18% absolute to 21.59% and AP₅₀^H by 21.81% absolute to 35.00%. We perform an additional experiment on $\mathcal{D}_{\text{ASD}}^{\text{val}}$, using the visible ground truth as VIS method. Table 1 shows the results: Especially for heavy occlusions S-AModal increases AP₅₀^H by 1.87% absolute to 43.79%. However, SAVOS achieves slightly better results on 4 out of the 7 metrics, e.g., AP and AP₅₀. This can be attributed to the differences in both methods which gives SAVOS an advantage in this setting: SAVOS takes as input the image, the visible mask and the optical flow to predict the full amodal mask. This means that in the ground truth setting it can simply learn to just add amodal areas beside the visible mask. In contrast, S-AModal only takes points of the visible mask as input to predict the amodal mask, and hence, cannot leverage the full mask-specific information, such as shape. However, the improvement in AP₅₀^H shows that once amodal and visible mask become more different due to occlusion, our proposed approach is better suited for this task. The higher robustness of our approach for practical cases, where no ground truth (GT) is available, can be seen in the first and the third row segment of Table 1.

Table 2 reports the video-level metrics of the proposed S-AModal method and SAVOS [50] on $\mathcal{D}_{\text{ASD}}^{\text{val}}$. Note that we cannot report video-level results for $\mathcal{D}_{\text{Kcar}}^{\text{test}}$ since video-level ground truth is not available. Using GenVIS, we see that results on video level show the same tendency as on image level: S-AModal outperforms SAVOS in all metrics, e.g., leading to an 1.69% absolute performance improvement towards a vAP₅₀ of 2.96%. The amodal VIS results of Table 2 are in line with the VIS results of GenVIS and can be attributed to relatively low tracking performance of the underlying VIS method. When using the ground truth as input (lower segment in Table 2), S-AModal interestingly outperforms SAVOS in all metrics, leading to an 14.18% absolute improvement in vAP₅₀^H, showing that S-AModal is better suited to handle heavy occlusions.



Fig. 4: Qualitative results of the proposed **S-AModal** method for three sequences \mathbf{x}_{t-1}^{t+1} with overlaid colored amodal predictions \mathbf{a}_{t-1}^{t+1} on AmodalSynthDrive $\mathcal{D}_{\text{ASD}}^{\text{val}}$.

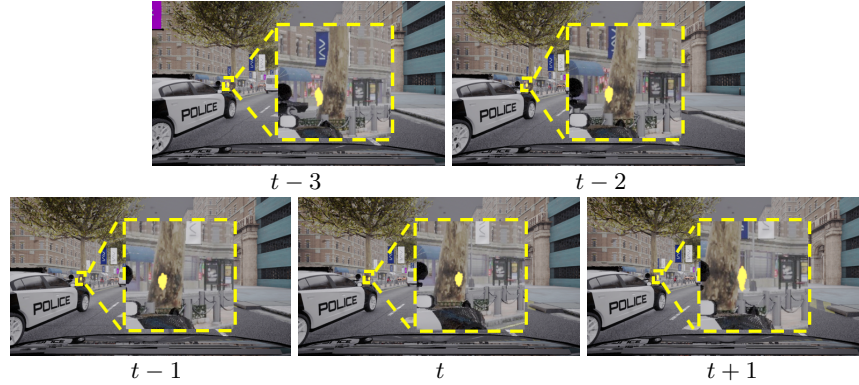


Fig. 5: Qualitative results of the proposed **S-AModal** method for a sequence \mathbf{x}_{t-3}^{t+1} with overlaid colored amodal predictions \mathbf{a}_{t-3}^{t+1} on AmodalSynthDrive $\mathcal{D}_{\text{ASD}}^{\text{val}}$, illustrating a fully occluded person for a time span of 3 frames ($t-2, t-1, t$).

Qualitative Results: Figure 4 shows qualitative results of **S-AModal** for three sequences \mathbf{x}_{t-1}^{t+1} of the validation data of AmodalSynthDrive $\mathcal{D}_{\text{ASD}}^{\text{val}}$. The amodal predictions \mathbf{a}_{t-1}^{t+1} are overlaid over the image for visualization purposes. The same color indicates the same identified instance. In all three videos, occluded pedestrians are tracked through occlusions with plausible amodal masks showing that our **S-AModal** method provides high-quality results in amodal VIS.

Figure 5 shows an example of **S-AModal** with a longer occlusion of a person (yellow mask) behind a large tree spanning 3 frames ($t-2, t-1, t$). The amodal mask of the last appearance at $t-3$ is shifted along the predicted point trajectory to frames $t-2, t-1, t$. **S-AModal** is clearly able to follow the person throughout this full occlusion until its reappearance in frame $t+1$.

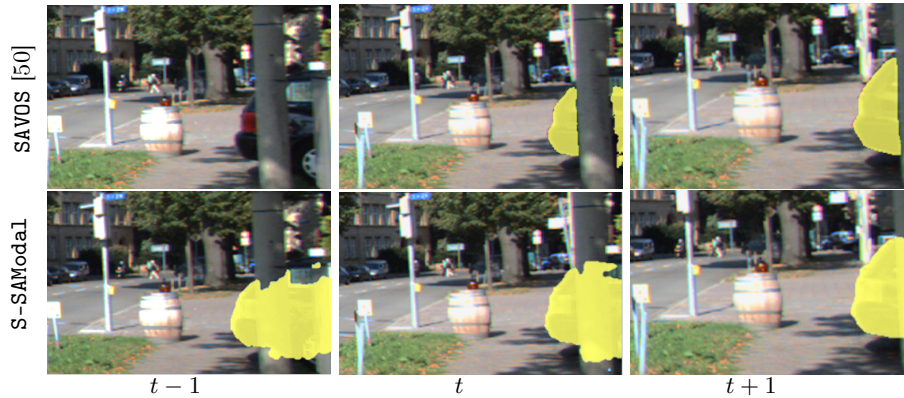


Fig. 6: Qualitative results of the SAVOS [50] (top) and the proposed S-AModal (bottom) methods for a sequence \mathbf{x}_{t-1}^{t+1} with yellow amodal predictions \mathbf{a}_{t-1}^{t+1} on $\mathcal{D}_{\text{Kcar}}^{\text{test}}$.

	Number of Point Prompts			
	$K = 1$	$K = 2$	$K = 3$	$K = 4$
AP	46.91 \pm 0.03	41.75 \pm 0.03	35.47 \pm 0.01	30.01 \pm 0.13
AP ₅₀	73.86 \pm 0.06	68.57 \pm 0.21	61.20 \pm 0.09	53.54 \pm 0.27
AP ₅₀ ^P	80.85 \pm 0.08	76.45 \pm 0.23	69.11 \pm 0.14	62.34 \pm 0.07
AP ₅₀ ^H	43.79 \pm 0.38	37.58 \pm 0.61	28.36 \pm 0.08	20.53 \pm 0.11
AP ₅₀ ^L	97.15 \pm 0.53	97.71 \pm 0.42	96.84 \pm 0.15	95.94 \pm 0.16
AP ₅₀ ^M	82.41 \pm 0.08	80.70 \pm 0.62	72.56 \pm 0.11	63.23 \pm 0.51
AP ₅₀ ^S	60.13 \pm 0.42	51.39 \pm 0.15	42.88 \pm 0.11	34.95 \pm 0.01

Table 3: Ablation: Amodal image-level instance segmentation metrics in AP and its metric derivatives on $\mathcal{D}_{\text{ASD}}^{\text{val}}$ using the ground-truth visible masks (GT) as VIS method and using different numbers of points K for prompting the amodal SAM network of S-AModal. Best results in bold.

In Figure 6 we compare our proposed S-AModal method (bottom) with the SAVOS method (top) on a sequence \mathbf{x}_{t-1}^{t+1} of $\mathcal{D}_{\text{Kcar}}^{\text{test}}$. The quantitative performance gain of S-AModal is reflected in these qualitative results as well. The visualized amodal segmentation masks of the car (yellow) are seemingly more accurate compared to the SAVOS results in handling the occlusion by the tree. Note that the tracking quality of both methods is inherited from PointTrack [48]. When the VIS method fails as in frame $t - 1$, SAVOS is not able to make a prediction.

Ablation: Amodal SAM is prompted using points as input. For all above experiments the number of point prompts per instance was chosen as one. Table 3 shows the results of the proposed S-AModal on image-level using different numbers K of points to prompt the amodal SAM network. Surprisingly, using more points does not lead to better performance. Using only $K = 1$ point leads to the best performance overall, e.g., an AP of 46.91%. Only for large objects, AP₅₀^L is slightly higher (97.71%) when using $K = 2$ point prompts compared to $K = 1$. However, when regarding the standard deviations, the slightly higher mean value is not significant. Since our point prompts are randomly selected from the visible mask, additional points may not provide additional information to amodal SAM.

	Point Selection Method			
	Random	Saliency	Erosion (default)	Erosion (best)
AP	46.91 \pm 0.03	36.84 \pm 0.15	48.03 \pm 0.18	49.21 \pm 0.04
AP ₅₀	73.86 \pm 0.06	58.16 \pm 0.19	75.25 \pm 0.10	76.26 \pm 0.38
AP ₅₀ ^P	80.85 \pm 0.08	65.83 \pm 0.28	82.30 \pm 0.10	82.85 \pm 0.07
AP ₅₀ ^H	43.79 \pm 0.38	22.87 \pm 0.31	45.90 \pm 0.25	47.84 \pm 0.24
AP ₅₀ ^L	97.15 \pm 0.53	88.76 \pm 0.53	97.65 \pm 0.45	98.05 \pm 0.11
AP ₅₀ ^M	82.41 \pm 0.08	69.15 \pm 0.72	83.58 \pm 0.50	85.61 \pm 0.16
AP ₅₀ ^S	60.13 \pm 0.42	42.17 \pm 0.08	62.17 \pm 0.09	62.66 \pm 0.43

Table 4: Ablation: Amodal **image-level** instance segmentation metrics in AP and its metric derivatives on $\mathcal{D}_{\text{ASD}}^{\text{val}}$ using the ground-truth visible masks (GT) as VIS method and using different point selection methods to prompt the amodal SAM network of S-AModal. Best results in **bold**.

We also investigate different point selection methods. For our main results in Tables 1, 2, we select points randomly from the visible mask. In Table 4, we show results on image level when instead of selecting a random point, we select the point with the highest saliency of the visible mask [10]. Moreover, we show results when applying the erosion algorithm to the visible mask to ensure, we do not sample our point prompt from the boarder regions of an instance. We report results for erosion using two different kernel sizes: the default size of 3×3 (default [43]) and the size 7×7 , which led to the best performance (best). As can be seen by the results in Table 4, sampling the point with the highest saliency does not lead to better results since it does not provide more information to amodal SAM. However, ensuring through the erosion algorithm [43] that we do not sample point prompts from boarder regions does lead to impressive performance gains in all metrics, e.g., an AP performance improvement of 1.12% absolute from random sampling to the sampling after default erosion (48.03%), and even 2.30% absolute from random sampling to sampling after erosion using the kernel size 7×7 (49.21%). Note that this simple post-processing of the visible segmentation masks leads to significant performance improvements while only adding a small computational overhead, i.e., the application of erosion per visible mask. The results in Table 4 support our hypothesis that point prompts sampled from ambiguous boarder regions of an instance confuse the amodal SAM method, resulting in failure cases as shown in Figure 8. This highlights the potential of prompt engineering [10, 18, 30, 35, 41] for this task and opens up a new research direction to design powerful prompts for amodal segmentation.

Additional qualitative results: Figure 7 shows additional qualitative results of our proposed S-AModal on three videos of $\mathcal{D}_{\text{ASD}}^{\text{val}}$. In Video 1, a person is heavily occluded by the turning taxi. S-AModal recovers the shape reliably throughout this occlusion. Video 2 shows in this case an occluded car. It vanishes behind a white truck in frame $t-1$. In the middle frame t the car is completely occluded but its position and shape are reasonably recovered by S-AModal. The reappearance of the car in frame $t+1$ is also challenging to segment amodally in this case due to multi-layer occlusion of the car by the white van, another car and the motorcyclist in front. In this complex scenario, S-AModal predicts a slightly too large amodal mask for the car in frame $t+1$. In Video 3, a pedestrian vanishes

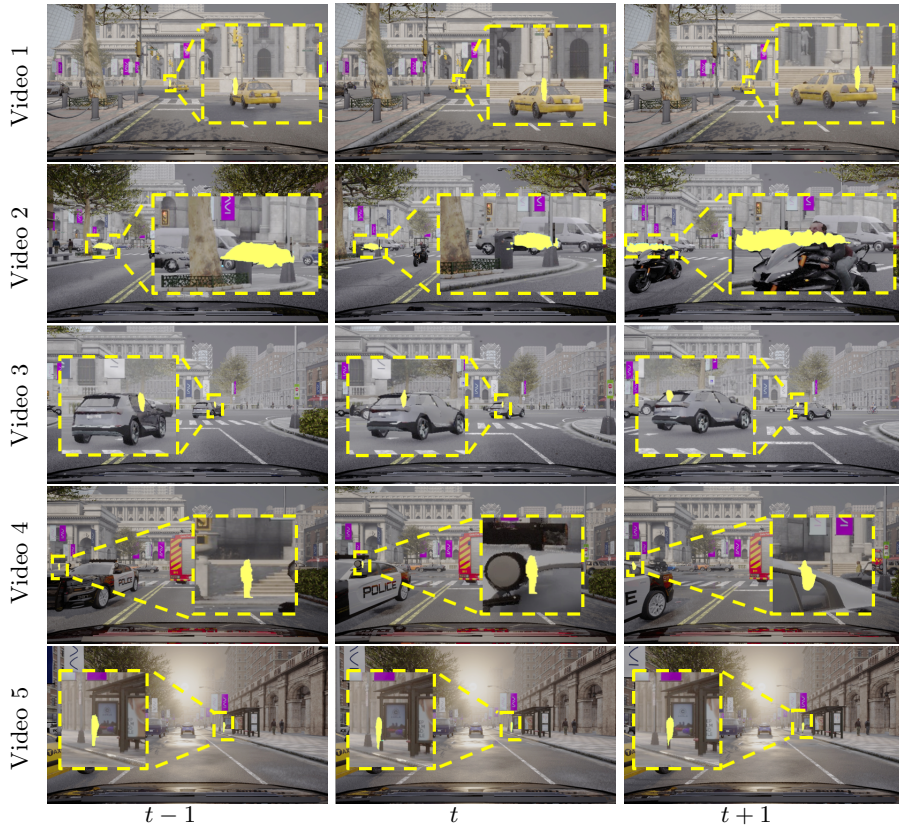


Fig. 7: Qualitative results of the proposed **S-AModal** method for five sequences \mathbf{x}_{t-1}^{t+1} with overlaid colored amodal predictions \mathbf{a}_{t-1}^{t+1} on AmodalSynthDrive $\mathcal{D}_{ASD}^{\text{val}}$.

behind a turning car. In frame $t - 1$, just part of the head is still visible. The person is completely occluded in frame t and reappears in frame $t + 1$. We see that in both partial occlusions ($t - 1, t + 1$) only a small part of the person is visible, and still **S-AModal** recovers the full shape of the person. In frame t , by relying on point tracking, the full shape of the person is predicted behind the car. Video 4 shows again a pedestrian fully occluded by a bypassing car. Also in this case, **S-AModal** is able to recover the full occlusion in frame t . In Video 5, a pedestrian is heavily occluded by a bus stop. In frames $t - 1, t + 1$, the feet are partly visible and hence, we are able to prompt the amodal **SAM** method to recover the full shape. In frame t , the pedestrian is not visible. Hence in this case, point tracking allows us to predict the amodal mask in frame t .

Limitations: Figure 8 illustrates limitations of our method. In Video 1, a person walks behind a tree and reappears. However, in frame $t - 1$, where the person is partially visible, amodal **SAM** predicts a wrong mask due to similar textures of person and car, which is then propagated to predict the amodal instance mask in the total occlusion in frame t . In Video 2, a person is occluded by a pillar. While the shape of the person is recovered in frames $t - 1$ and $t + 1$, the more

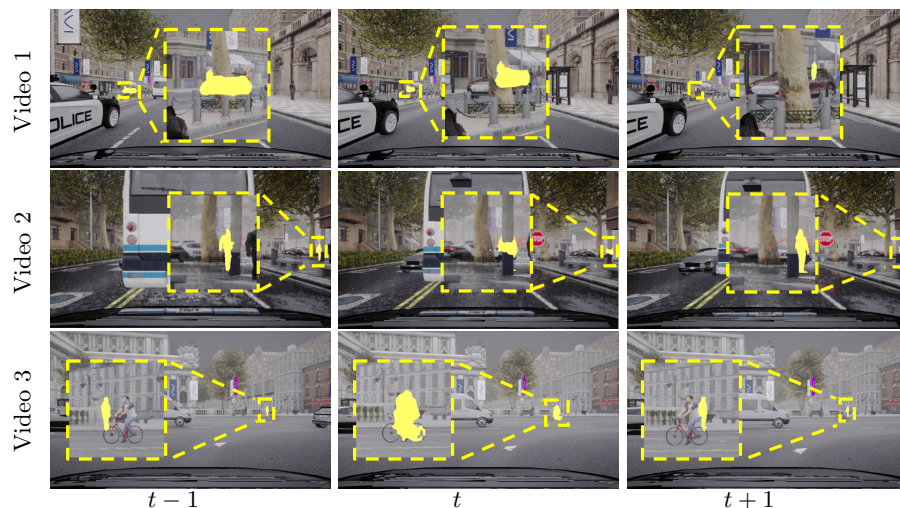


Fig. 8: Failure Cases of the proposed **S-AModal** method for three sequences \mathbf{x}_{t-1}^{t+1} with overlaid colored amodal predictions \mathbf{a}_{t-1}^{t+1} on AmodalSynthDrive $\mathcal{D}_{\text{ASD}}^{\text{val}}$.

challenging occlusion in frame t cannot be fully resolved: Only the upper body is predicted. We attribute this mainly to two reasons: First, the occlusion in frame t , where object parts are visible to the left and right of an occluder, is more often seen for the typically wider vehicle classes whose full shape is similar to the one recovered in frame t . Second, Video 2 is affected by challenging conditions like heavy rain and low light. In Video 3, a pedestrian in frame t is heavily but not fully occluded by a bypassing cyclist. In frames $t-1$ and $t+1$, the pedestrian is correctly segmented, however in frame t , only a very small part of the pedestrian is visible behind the cyclist and **S-AModal** is prompted using a point close to the cyclist. This leads to a confusion of the instances and the faulty segmentation of the cyclist in frame t . Future work could address these limitations by developing stronger prompts for amodal segmentation and adapting the newly published **SAM 2** [37] to this task to enhance prediction consistency.

6 Conclusions

In this work, we propose **S-AModal** for amodal video instance segmentation (VIS) with a focus on automated driving. To our knowledge, this is the first work to incorporate foundation models into the amodal segmentation task. We show that it is possible to adapt the original **SAM** model to the prompted amodal instance segmentation task using point prompts. Moreover, we show that incorporating this model into a VIS pipeline leads to an amodal VIS method with state-of-the-art (SOTA) performance while not relying on video-wise amodal VIS labels. Additionally, by incorporating the recent advances in point tracking into our pipeline, we are able to surpass limitations of previous amodal segmentation methods on both image- and video-level metrics and are able to provide instance masks also for temporal full occlusions of instances in the video sequence.

References

1. Ao, J., Ke, Q., Ehinger, K.A.: Image Amodal Completion: A Survey. *Computer Vision and Image Understanding* **229**, pp. 1–18 (Mar 2023)
2. Ao, J., Ke, Q., Ehinger, K.A.: Amodal Intra-class Instance Segmentation: Synthetic Datasets and Benchmark. In: *Proc. of WACV*. pp. 281–290. Waikoloa, HI, USA (Jan 2024)
3. Back, S., Lee, J., Kim, T., Noh, S., Kang, R., Bak, S., Lee, K.: Unseen Object Amodal Instance Segmentation via Hierarchical Occlusion Modeling. In: *Proc. of ICRA*. pp. 5085–5092. Philadelphia, PA, USA (May 2022)
4. Breitenstein, J., Fingscheidt, T.: Amodal Cityscapes: A New Dataset, its Generation, and an Amodal Semantic Segmentation Challenge Baseline. In: *Proc. of IV*. pp. 1018–1025. Aachen, Germany (Jun 2022)
5. Breitenstein, J., Jin, K., Hakiri, A., Klingner, M., Fingscheidt, T.: End-to-end Amodal Video Instance Segmentation. In: *Proc. of BMVC - Workshops*. pp. 1–15. Aberdeen, UK (Nov 2023)
6. Breitenstein, J., Löhdefink, J., Fingscheidt, T.: Joint Prediction of Amodal and Visible Semantic Segmentation for Automated Driving. In: *Proc. of ECCV - Workshops*. pp. 633–645. Tel Aviv, Israel (Oct 2022)
7. Chen, T., Zhu, L., Ding, C., Cao, R., Zhang, S., Wang, Y., Li, Z., Sun, L., Mao, P., Zang, Y.: SAM-Adapter: Adapting Segment Anything in Underperformed Scenes. In: *Proc. of ICCV - Workshops*. pp. 3367–3375. Paris, France (Oct 2023)
8. Cheng, Y., Li, L., Xu, Y., Li, X., Yang, Z., Wang, W., Yang, Y.: Segment and Track Anything. *arXiv* **2305.06558**, pp. 1–8 (May 2023)
9. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The Cityscapes Dataset for Semantic Urban Scene Understanding. In: *Proc. of CVPR*. pp. 3213–3223. Las Vegas, NV, USA (Jun 2016)
10. Dai, H., Ma, C., Yan, Z., Liu, Z., Shi, E., Li, Y., Shu, P., Wei, X., Zhao, L., Wu, Z., Zeng, F., Zhu, D., Liu, W., Li, Q., Sun, L., Liu, S.Z.T., Li, X.: SAMAug: Point Prompt Augmentation for Segment Anything Model. *arXiv* **2307.01187**, pp. 1–16 (Jul 2023)
11. Doersch, C., Gupta, A., Markeeva, L., Recasens, A., Smaira, L., Aytar, Y., Carreira, J., Zisserman, A., Yang, Y.: TAP-Vid: A Benchmark for Tracking Any Point in a Video. In: *Proc. of NeurIPS*. pp. 13610–13626. New Orleans, LA, USA (Dec 2022)
12. Doersch, C., Yang, Y., Vecerik, M., Gokay, D., Gupta, A., Aytar, Y., Carreira, J., Zisserman, A.: TAPIR: Tracking Any Point with per-Frame Initialization and Temporal Refinement. In: *Proc. of ICCV*. pp. 10061–10072. Paris, France (Oct 2023)
13. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In: *Proc. of ICLR*. pp. 1–21. virtual (May 2021)
14. Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V.: CARLA: An Open Urban Driving Simulator. In: *Proc. of CoRL*. pp. 1–16. Mountain View, CA, USA (Nov 2017)
15. Fingscheidt, T., Gottschalk, H., Houben, S. (eds.): *Deep Neural Networks and Data for Automated Driving: Robustness, Uncertainty Quantification, and Insights Towards Safety*. Springer Nature, Cham (2022). <https://doi.org/10.1007/978-3-031-01233-4>, <https://library.oapen.org/handle/20.500.12657/57375>

16. Follmann, P., König, R., Härtinger, P., Klostermann, M.: Learning to See the Invisible: End-to-End Trainable Amodal Instance Segmentation. In: Proc. of WACV. pp. 1328–1336. Waikoloa Village, HI, USA (Jan 2019)
17. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision Meets Robotics: The KITTI Dataset. *International Journal of Robotics Research (IJRR)* **32**(11), pp. 1231–1237 (Aug 2013)
18. Gu, J., Han, Z., Chen, S., Beirami, A., He, B., Zhang, G., Liao, R., Qin, Y., Tresp, V., Torr, P.: A Systematic Survey of Prompt Engineering on Vision-Language Foundation Models. arXiv **2307.12980**, pp. 1–21 (Jul 2023)
19. Harley, A.W., Fang, Z., Fragkiadaki, K.: Particle Video Revisited: Tracking Through Occlusions Using Point Trajectories. In: Proc. of ECCV. pp. 59–75. Tel Aviv, Israel (Oct 2022)
20. Heo, M., Hwang, S., Hyun, J., Kim, H., Oh, S.W., Lee, J.Y., Kim, S.J.: A Generalized Framework for Video Instance Segmentation. In: Proc. of CVPR. pp. 14623–14632. Vancouver, BC, Canada (Jun 2023)
21. Hu, Y.T., Chen, H.S., Hui, K., Huang, J.B., Schwing, A.G.: SAIL-VOS: Semantic Amodal Instance Level Video Object Segmentation – A Synthetic Dataset and Baselines. In: Proc. of CVPR. pp. 3105–3115. Long Beach, CA, USA (Jun 2019)
22. Karaev, N., Rocco, I., Graham, B., Neverova, N., Vedaldi, A., Rupperecht, C.: CoTracker: It is Better to Track Together. arXiv **2307.07635**, pp. 1–13 (Jul 2023)
23. Ke, L., Tai, Y.W., Tang, C.K.: Deep Occlusion-Aware Instance Segmentation with Overlapping BiLayers. In: Proc. of CVPR. pp. 4019–4028. Nashville, TN, USA (Jun 2021)
24. Ke, L., Ye, M., Danelljan, M., Liu, Y., Tai, Y.W., Tang, C.K., Yu, F.: Segment Anything in High Quality. In: Proc. of NeurIPS. pp. 29914–29934. Vancouver, BC, Canada (Dec 2023)
25. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollár, P., Girshick, R.: Segment Anything. In: Proc. of ICCV. pp. 4015–4026. Paris, France (Oct 2023)
26. Li, K., Malik, J.: Amodal Instance Segmentation. In: Proc. of ECCV. pp. 677–693. Amsterdam, The Netherlands (Oct 2016)
27. Liao, Y., Xie, J., Geiger, A.: KITTI-360: A Novel Dataset and Benchmarks for Urban Scene Understanding in 2D and 3D. arXiv **2109.13410**, pp. 1–32 (Sep 2021)
28. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common Objects in Context. In: Proc. of ECCV. pp. 740–755. Zurich, Switzerland (Sep 2014)
29. Ling, H., Acuna, D., Kreis, K., Kim, S.W., Fidler, S.: Variational Amodal Object Completion. In: Proc. of NeurIPS. pp. 16246–16257. Vancouver, BC, Canada (Dec 2020)
30. Lüddecke, T., Ecker, A.: Image Segmentation Using Text and Image Prompts. In: Proc. of CVPR. pp. 7086–7096. New Orleans, LA, USA (Jun 2022)
31. Mohan, R., Valada, A.: Amodal Panoptic Segmentation. In: Proc. of CVPR. pp. 21023–21032. New Orleans, LA, USA (Jun 2022)
32. Nguyen, K., Todorovic, S.: A Weakly Supervised Amodal Segmenter with Boundary Uncertainty Estimation. In: Proc. of ICCV. pp. 2995–3003. Virtual (Oct 2021)
33. Purkait, P., Zach, C., Reid, I.D.: Seeing Behind Things: Extending Semantic Segmentation to Occluded Regions. In: Proc. of IROS. pp. 1998–2005. Macau, SAR, China (Nov 2019)

34. Qi, L., Jiang, L., Liu, S., Shen, X., Jia, J.: Amodal Instance Segmentation With KINS Dataset. In: Proc. of CVPR. pp. 3014–3023. Long Beach, CA, USA (Jun 2019)
35. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning Transferable Visual Models From Natural Language Supervision. In: Proc. of ICML. pp. 8748–8763. virtual (Jul 2021)
36. Rajič, F., Ke, L., Tai, Y.W., Tang, C.K., Danelljan, M., Yu, F.: Segment Anything Meets Point Tracking. arXiv **2307.01197**, pp. 1–15 (Dec 2023)
37. Ravi, N., Gabeur, V., Hu, Y.T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., Mintun, E., Pan, J., Alwala, K.V., Carion, N., Wu, C.Y., Girshick, R., Dollár, P., Feichtenhofer, C.: SAM 2: Segment Anything in Images and Videos. arXiv **2408.00714**, pp. 1–41 (Aug 2024)
38. Reddy, N.D., Tamburo, R., Narasimhan, S.: WALT: Watch And Learn 2D Amodal Representation using Time-lapse Imagery. In: Proc. of CVPR. pp. 9356–9366. New Orleans, LA, USA (Jun 2022)
39. Sekkat, A.R., Mohan, R., Sawade, O., Matthes, E., Valada, A.: AmodalSynthDrive: A Synthetic Amodal Perception Dataset for Autonomous Driving. arXiv **2309.06547**, pp. 1–12 (Sep 2023)
40. Shaharabany, T., Dahan, A., Giryas, R., Wolf, L.: AutoSAM: Adapting SAM to Medical Images by Overloading the Prompt Encoder. In: Proc. of BMVC. pp. 1–15. Aberdeen, UK (Nov 2023)
41. Shtedritski, A., Rupprecht, C., Vedaldi, A.: What Does CLIP Know About a Red Circle? Visual Prompt Engineering for VLMs. In: Proc. of ICCV. pp. 11987–11997. Paris, France (Oct 2023)
42. Sun, Y., Kortylewski, A., Yuille, A.: Amodal Segmentation through Out-of-Task and Out-of-Distribution Generalization with a Bayesian Model. In: Proc. of CVPR. pp. 1215–1224. New Orleans, LA, USA (Jun 2022)
43. Van der Walt, S., Schönberger, J.L., Nunez-Iglesias, J., Boulogne, F., Warner, J.D., Yager, N., Gouillart, E., Yu, T.: scikit-image: Image Processing in Python. *PeerJ* **2**, p. e453 (May 2014)
44. Wang, Q., Chang, Y.Y., Cai, R., Li, Z., Hariharan, B., Holynski, A., Snavely, N.: Tracking Everything Everywhere All at Once. In: Proc. of ICCV. pp. 19795–19806. Paris, France (Oct 2023)
45. Wu, J., Ji, W., Liu, Y., Fu, H., Xu, M., Xu, Y., Jin, Y.: Medical SAM Adapter: Adapting Segment Anything Model for Medical Image Segmentation. arXiv **2304.12620**, pp. 1–10 (Apr 2023)
46. Xiao, A., Xuan, W., Qi, H., Xing, Y., Ren, R., Zhang, X., Ling, S., Lu, S.: CAT-SAM: Conditional Tuning Network for Few-Shot Adaptation of Segmentation Anything Model. arXiv **2402.03631**, pp. 1–25 (Feb 2024)
47. Xie, Z., Guan, B., Jiang, W., Yi, M., Ding, Y., Lu, H., Zhang, L.: PA-SAM: Prompt Adapter SAM for High-quality Image Segmentation. In: Proc. of ICME. pp. 1–10. Niagra Falls, Canada (Jul 2024)
48. Xu, Z., Zhang, W., Tan, X., Yang, W., Huang, H., Wen, S., Ding, E., Huang, L.: Segment as Points for Efficient Online Multi-Object Tracking and Segmentation. In: Proc. of ECCV. pp. 1–17. Glasgow, UK (Aug 2020)
49. Yang, J., Gao, M., Li, Z., Gao, S., Wang, F., Zheng, F.: Track Anything: Segment Anything Meets Videos. arXiv **2304.11968**, pp. 1–7 (Apr 2023)
50. Yao, J., Hong, Y., Wang, C., Xiao, T., He, T., Locatello, F., Wipf, D., Fu, Y., Zhang, Z.: Self-supervised Amodal Video Object Segmentation. In: Proc. of NeurIPS. pp. 6278–6291. New Orleans, LA, USA (Dec 2022)

51. Zheng, Y., Harley, A.W., Shen, B., Wetzstein, G., Guibas, L.J.: PointOdyssey: A Large-Scale Synthetic Dataset for Long-Term Point Tracking. In: Proc. of ICCV. pp. 19855–19865. Paris, France (Oct 2023)
52. Zhu, Y., Tian, Y., Metaxas, D., Dollár, P.: Semantic Amodal Segmentation. In: Proc. of CVPR. pp. 1464–1472. Honolulu, HI, USA (Jul 2017)