Panoptic Out-of-Distribution Segmentation

Rohit Mohan, Kiran Kumaraswamy, Juana Valeria Hurtado, Kürsat Petek, and Abhinav Valada

Abstract—Deep learning has led to remarkable strides in scene understanding with panoptic segmentation emerging as a key holistic scene interpretation task. However, the performance of panoptic segmentation is severely impacted in the presence of outof-distribution (OOD) objects i.e. categories of objects that deviate from the training distribution. To overcome this limitation, we propose panoptic out-of-distribution segmentation for joint pixellevel semantic in-distribution and out-of-distribution classification with instance prediction. We extend two established panoptic segmentation benchmarks, Cityscapes and BDD100K, with outof-distribution instance segmentation annotations, propose suitable evaluation metrics, and present multiple strong baselines. Importantly, we propose the novel PoDS architecture with a shared backbone, an OOD contextual module for learning global and local OOD object cues, and dual symmetrical decoders with task-specific heads that employ our alignment-mismatch strategy for better OOD generalization. Combined with our data augmentation strategy, this approach facilitates progressive learning of out-of-distribution objects while maintaining indistribution performance. We perform extensive evaluations that demonstrate that our proposed PoDS network effectively addresses the main challenges and substantially outperforms the baselines. We make the dataset, code, and trained models publicly available at http://pods.cs.uni-freiburg.de.

I. INTRODUCTION

Recent advances in deep learning have substantially improved the capabilities of autonomous systems to interpret their surroundings [1], [2]. Central to these advancements is panoptic segmentation [3], which integrates semantic segmentation with instance segmentation, providing a holistic understanding of the environment. However, a significant challenge is that these models yield overconfident predictions of object categories out of the distribution they were trained on, known as out-ofdistribution (OOD) objects. Segmenting these OOD objects poses a major challenge as they can vary significantly in appearance and semantics, include fine-grained details, and share visual characteristics with in-distribution objects, leading to ambiguity. Moreover, learning to jointly segment both OOD objects and in-distribution categories is extremely challenging as detailed in Sec. III-3. Given the potential consequences of autonomous systems malfunctioning due to unexpected inputs [4], it is crucial to ensure the safe and robust deployment.

To directly address these challenges at the task level, we introduce panoptic out-of-distribution segmentation that focuses on holistic scene understanding while effectively segmenting OOD objects. Fig. 1 illustrates our proposed task that aims to predict both the semantic segmentation of *stuff* classes and instance segmentation of *thing* classes as well as an OOD class. An object is considered OOD if it is not present in the training distribution but appears in the testing/deployment

Department of Computer Science, University of Freiburg, Germany.



Fig. 1: The *panoptic segmentation* network (*) presents erroneous predictions when the input contains objects that are not representative of the distribution it was trained on. *Panoptic out-of-distribution segmentation* aims to address this by predicting both semantic and instance segmentation of *stuff* and *thing* classes, while also predicting instances of unseen out-of-distribution classes.

stages. This distinguishes panoptic OOD segmentation from the closely related open-set panoptic segmentation [5]. Further, panoptic OOD segmentation does not reason about the semantic differences between OOD objects since in most robotics settings, especially navigation, it is sufficient to identify OOD objects and further semantically categorizing them does not provide significant utility.

In this work, we establish two challenging benchmarks, Cityscapes-OOD and BDD100K-OOD, by extending the standard autonomous driving datasets with OOD instance segmentation annotations. We present several strong baselines by combining semantic out-of-distribution segmentation methods with a class-agnostic instance segmentation decoder or adapting open-set segmentation approaches. We also introduce a tailored Panoptic Out-of-Distribution Quality (POD-Q) metric to quantify the performance. More importantly, as a first novel approach, we propose the PoDS architecture that incorporates out-of-distribution perception ability into a panoptic segmentation network conditioned on prior knowledge of in-distribution classes. By doing so, the network avoids the pitfalls of ambiguously modeling both OOD and in-distribution classes, thereby improving generalization and adept handling of unseen OOD objects. We perform extensive experimental evaluations that first demonstrate the feasibility of the task and further that our proposed PoDS architecture significantly outperforms the baselines, ensuring a balanced performance on both in-distribution and out-of-distribution classes.

In summary, the contributions of this work are as follows:

- 1) We introduce the novel panoptic OOD segmentation task, identifying its main challenges, along with multiple baselines, and a suitable POD-Q metric.
- We present the Cityscapes-OOD and BDD100K-OOD benchmarks, which extend the established datasets with

This work was funded by the German Research Foundation (DFG) Emmy Noether Program grant No 468878300.

OOD instance segmentation annotations.

- 3) We propose the novel PoDS architecture that incorporates the proposed modules to embed OOD segmentation capabilities into a panoptic segmentation network leveraging conditional in-distribution priors.
- 4) We present comprehensive quantitative and qualitative evaluations to demonstrate the feasibility of the task and the efficacy of our proposed PoDS architecture.
- 5) We make the code, datasets, and models publicly available at http://pods.cs.uni-freiburg.de.

II. RELATED WORK

In this section, we present an overview of panoptic segmentation methods, followed by out-of-distribution segmentation approaches and open-set panoptic segmentation methods.

Panoptic Segmentation methods can be categorized as top-down and bottom-up approaches. Top-down methods [6] employ taskspecific heads, where the instance segmentation head predicts bounding boxes and corresponding masks for objects, while the semantic segmentation head generates dense semantic predictions for each class. The outputs from these heads are then combined using heuristic-based fusion modules [7], [6]. Conversely, bottom-up methods [8] begin with semantic segmentation and then employ various techniques [9] to group thing pixels together to obtain instance segmentation. Recently, Mohan et al. [10] introduced the PAPS architecture with a shared backbone, an asymmetrical dual-decoder, and several modules for amodal panoptic segmentation [11], which predicts both visible and occluded object segments. We base our approach on PAPS's modal variant, which perceives only visible segments as it outperforms other bottom-up methods.

Semantic Out-of-Distribution Segmentation is often addressed through the use of uncertainty estimation techniques. A popular method is the maximum softmax probability (MSP) [12] that uses probabilities from the softmax distribution. Following, maximum logit (MaxLogit) [13] uses the negative of the maximum unnormalized logit to deliver improved performance in semantic out-of-distribution segmentation over MSP. On the other hand, Bayesian networks generate uncertainty estimates by modeling their weights and outputs as probability distributions rather than fixed values [14]. However, Bayesian inferences can be computationally expensive, hence in practice methods such as Dropout [15] or ensembles [16] which capture model uncertainty by averaging predictions over multiple models are often used as approximations. Various frameworks also use density estimation [17] via estimating the likelihood of samples with respect to the training distribution for addressing semantic out-of-distribution segmentation. Furthermore, [18] proposes a loss function to yield high entropy for out-ofdistribution sample predictions. The use of autoencoders on in-distribution data has also been explored to identify erroneous and less reliable reconstructions of out-of-distribution samples due to unseen patterns during training. Generative models [19] generate OOD data as boundary samples. This is however very challenging to scale to complex and high-dimensional data such as high-resolution images of urban scenes. Other approaches include using adversarial perturbations on the input during training and test-time to predict in- and out-of-distribution

samples. ODIN [20] uses temperature scaling with small adversarial perturbations on the input at test-time, while [21] use adversarial attacks during training as a proxy for out-of-distribution training samples. In this work, we adapt a number of aforementioned methods to serve as baselines for panoptic out-of-distribution segmentation as described in Sec. III-5.

Open-Set Panoptic Segmentation: There are only two approaches that have been proposed thus far. EOPSN [5] groups similar unlabeled objects across multiple inputs during training and assigns labels to unlabeled objects that are surrounded by known segments. Following, Xu *et al.* [22] use a known classification head to reject segments while employing a classagnostic classifier to identify the segments as unknown objects. We use these methods as baselines with some adaptations as described in Sec. III-5.

III. PANOPTIC OUT-OF-DISTRIBUTION SEGMENTATION

1) Task Definition: Panoptic out-of-distribution segmentation aims to assign each pixel *i* of an input image to an output pair $(c_i, \kappa_i) \in (C \cup O) \times N$. Here, *C* denotes known semantic classes, while *O* represents the out-of-distribution class, such that $C \cap O = \emptyset$, and *N* is the total number of instances. *C* is further divided into *stuff* labels C^S (e.g., sidewalks) and *thing* labels C^T (e.g., pedestrians). In this task, the variable c_i can be a semantic or OOD class, and κ_i indicates the corresponding instance ID. For *stuff* classes, κ_i is not applicable.

2) Evaluation Metric: To quantify the performance, it is essential to evaluate both in-distribution and out-of-distribution performance equally. To this end, we introduce the Panoptic Out-of-Distribution Quality (POD-Q), which builds upon the panoptic quality (PQ) metric [3]. We first determine PQ_{out} representing the PQ for the OOD class and PQ_{in} accounting for all the in-distribution semantic classes. Finally, POD-Q is computed as the geometric mean of PQ_{out} and PQ_{in} :

$$POD-Q = (PQ_{out} \times PQ_{in})^{\frac{1}{2}}.$$
 (1)

We use the geometric mean to incentivize balanced performance in both out-of-distribution and in-distribution segmentation while strictly penalizing methods that only excel in one aspect of the task. For further details on PQ_{out} and PQ_{in} , please refer to Sec. S.1 of the supplementary material.

3) Challenges: Classifying and segmenting objects that do not belong to the known training distribution is challenging due to the absence of explicit knowledge about diverse OOD object characteristics. This becomes more challenging with the simultaneous identification and segmentation of OOD objects with panoptic segmentation of in-distribution classes. The increased complexity makes naive adaption of methods from the less complex tasks such as semantic out-of-distribution segmentation vulnerable to trade-offs prioritizing one aspect over the other. Unsupervised methods that condition segmentation outputs based on threshold scores to predict OOD objects become sensitive, as any fragmentation in predictions can result in false instance predictions. Conversely, in learning with supervised OOD data, where training data is limited and does not encompass all OOD object variations, models can



Fig. 2: Incorporation of OOD objects into a scene from the Cityscapes dataset is shown, with (c) random scaling and (d) depth-based OOD object scaling.



Fig. 3: Sample images from the Cityscapes-OOD and BDD100K datasets.

overfit to specific OOD objects encountered during training. Consequently, this results in difficulties recognizing new OOD objects during inference. Considering the panoptic out-ofdistribution task, which demands a balanced performance for both in-distribution and out-of-distribution scene elements, it becomes evident that a comprehensive approach is required.

4) Datasets: Given the high cost and complexity of annotating panoptic segmentation data, it is impractical to manually label a new dataset that encompasses a diverse set of real-world out-of-distribution instances for panoptic out-of-distribution segmentation. As an alternative, we extend established urban scene understanding datasets for panoptic segmentation by incorporating real-world OOD object instances, creating two new datasets: Cityscapes-OOD and BDD100K-OOD.

Dataset Creation Protocol: We extract atypical objects from the LVIS [23] instance segmentation dataset using their segmentation masks. Objects such as cats and desks that are not present in the original panoptic segmentation dataset are added to the images. Their positions and the number of instances are randomized, with the object likelihood based on their typical locations (e.g., couches at the bottom, airplanes at the top). We further employ depth-dependent scaling to resize the OOD objects, ensuring that objects near the ego-car are relatively larger than the ones positioned far away. To do so, we begin by determining the sizes of objects in the original panoptic segmentation dataset and then match these sizes to bins established based on depth. Based on their size, the extracted OOD objects are paired with known semantic classes (e.g., surfboard with person, couch with car). We then overlay these objects, selecting a size from the depth bin randomly based on their positioning (Fig. 2 (a) and (b)). We use blending techniques [17] such as color shifts, depth blur, color curve, and gamma transformations, and we remove low-quality samples to enhance the dataset's quality and remove low-quality samples to improve the dataset's quality. Lastly, we ensure OOD objects in the training set are distinct from those in the test set, guaranteeing novelty during testing, and consistent with the requirements of the panoptic out-of-distribution segmentation task.

Cityscapes-OOD: We create the Cityscapes-OOD dataset for panoptic out-of-distribution segmentation, with 11 *stuff* classes,



Fig. 4: Dataset statistics for (a) Cityscapes-OOD and (b) BDD100K-OOD. Note that each *stuff* class has a single occurrence per image.

8 *thing* classes, and an *OOD* class by extending Cityscapes [24]. It consists of 2975 training and 500 test images at a resolution of 2048×1024 pixels. Test set annotations which are generated from the validation set of Cityscapes are not publicly released, and evaluation is only possible through an online server. Fig. 3 (a) and Fig. 4 (a) shows an example and dataset statistics.

BDD100K-OOD dataset consists of 7000 training and 1000 validation images with a resolution of 1280×720 pixels and is an extension of BDD100K [25], augmented with out-of-distribution objects. It features one *OOD* class, 11 *stuff* classes including roads and buildings, and eight *thing* classes such as cars and bicycles. Fig. 3 (b) and Fig. 4 (b) present an example and dataset statistics, respectively.

5) Baselines: We present seven baselines for the panoptic out-of-distribution segmentation task. We adapt four effective semantic out-of-distribution segmentation methods (MSP [12], MaxLogit [13], ODIN [21], Meta-OOD [18]) with the PAPS [10] modal panoptic segmentation architecture (PAPS* as described in Sec. IV-1). We compute the OOD semantic class from the semantic segmentation output from PAPS* and then use the post-processing approach from [8] for *thing*+OOD foreground segmentation to obtain the final panoptic out-of-distribution segmentation prediction. For the two remaining baselines, EPSON [5] and DD-OPS [22], we restrict the segmentation of unknown classes to a single OOD class and enhance their base network with EfficientPS [6], a state-of-the-art top-down panoptic segmentation network.

IV. PODS NETWORK ARCHITECTURE

In this section, we detail our proposed PoDS architecture depicted in Fig. 5. We first present an overview of the PoDS network, followed by a detailed description of each constituting component. PoDS builds on top of a base panoptic segmentation network that has a shared backbone and task-specific decoders (purple) by incorporating modules specially designed to embed out-of-distribution capabilities based on prior knowledge of in-distribution classes. We incorporate an OOD contextual module (blue) that complements the robust in-distribution semantic features of the shared backbone with both global discriminatory and fine local OOD object representations. Subsequently, we introduce an additional task-specific decoder (green), equipped with dynamic modules, alongside the existing ones. This design allows for adaptive integration of OOD features while preserving the in-distribution features of the high-performing base panoptic network. The unique dual task-



Fig. 5: Illustration of our proposed PoDS architecture that consists of a shared backbone with an OOD contextual module and symmetrical task-specific decoder arranged in a dual configuration setup to facilitate an alignment-mismatch learning strategy. The shared backbone learns robust feature representations for in-distribution semantic categories while the OOD contextual module supports both global and local features for OOD objects. The network comprises symmetrical semantic and instance decoders that include dynamic modules to adaptively balance the features between in- and out-distribution representations.

specific decoder configuration benefits further from our novel alignment-mismatch loss. This loss encourages learning finer details within in-distribution semantic classes and what lies outside by balancing consensus and divergence between the two heads. Furthermore, we incorporate a data augmentation strategy to facilitate the training of our novel modules. Please refer to Sec. S.2 of the supplementary material for further implementation details.

1) Base Network: Building on the modal variant of PAPS [10], which excels in panoptic segmentation, we develop an architecture for panoptic out-of-distribution segmentation. The modal PAPS architecture has a shared backbone, decoders, prediction heads, a context extractor, and a cross-task module. For our PoDS network, we streamline this by excluding the instance segmentation decoder and cross-task module. Instead, we adopt the semantic segmentation decoder with the dense prediction cell module, along with two upsampling stages and skip connections for the instance segmentation decoder. The instance segmentation head remains intact, handling instance center prediction and regression. This streamlined PAPS architecture, termed PAPS*, achieves a PQ score of 63.7 on the Cityscapes validation set, close to PAPS's 64.3. As shown in Fig. 5 purple boxes with red locks, we pretrain PAPS* on in-distribution panoptic segmentation datasets, and keep its weights fixed throughout the out-of-distribution segmentation training.

2) OOD Contextual Module: We introduce the OOD Contextual Module for the PoDS architecture, designed to capture both global and local features of out-of-distribution (OOD) objects in images. As depicted in Fig. 5 (blue box), this module incorporates two residual bottleneck blocks, similar to the fourth and fifth stages of Regnet [26]. The module takes the output from the last layer of the backbone's stage 2, processes it through the first block, combines it with the output from the last layer of stage 3, and routes it through the second block. Subsequently, the output from the second block, named O_{ocm} , proceeds to a global average pooling layer and then to a classification head. In parallel, Oocm undergoes upsampling followed by two convolution layers. This processed output splits into two branches: one path goes to an in/outdistribution segmentation head, while the other undergoes further upsampling and convolutions before reaching a second segmentation head. Both heads distinguish between pixelwise in-distribution and out-of-distribution regions. During training, we apply random data augmentation, as detailed in Sec. IV-6, generating samples with OOD objects from the panoptic segmentation dataset to train both the classification (targeting OOD global features) and pixel-wise segmentation heads (focusing on OOD local features). We employ binary cross-entropy loss for the classification head and weighted pixelwise cross-entropy loss for the in/out-distribution segmentation heads, with weights of 0.7 and 0.8 respectively. Thus, L_{ocm} is the sum of the aforementioned losses. Notably, backpropagation for the pixel-wise loss is only triggered when an OOD object is present in the input.

3) Dynamic Module: The dynamic module is defined by the following inputs: an input feature map F, O_{ocm} (G) from the OOD contextual module, and feature map and convolution functions (K and $g_1(\cdot, w_1)$ and $g_2(\cdot, w_2)$, respectively) from the base network. The inputs from the base network are represented by the red, yellow, and pink arrows in Fig. 5, respectively.

$$F_R = g_1^* (F, w_1 + \Delta w_1^*), \tag{2}$$

$$F_R = g_2^* (F_R, w_2 + \Delta w_2^*), \tag{3}$$

$$F_O = h_1(G) \cdot F_R + (1 - h_1(G)) \cdot K, \tag{4}$$

where $g_1^*(\cdot, w_1 + \Delta w_1^*)$ and $g_2^*(\cdot, w_2 + \Delta w_2^*)$ denote convolution functions with learned weights w^* as offsets from the weights w of $g_1(\cdot, w_1)$ and $g_2(\cdot, w_2)$, respectively. The computation of F_R is performed in a sequential manner with an initial input F using the functions $g_1^*(\cdot)$ and $g_2^*(\cdot)$. The weighted gating function h_1 is composed of a consecutive global pooling and a 1×1 convolution layer. By using offset weights and OOD contextual features O_{ocm} , the proposed module establishes a pre-training and training link between the convolutional weights of the base network and its own weights, which can be dynamically adjusted. This allows the module to maintain the pre-learned knowledge of the base network while incorporating out-of-distribution capabilities, making minor adjustments if the OOD object is similar to known semantic classes and significant adjustments if it is considerably different.

4) PoDS Decoders and Heads: In the PoDS framework, we utilize additional decoders akin to the base network described in Sec. IV-1, as visualized with green boxes in Fig. 5. Each decoder starts by merging the upsampled DPC features from their base network's task-specific decoders (purple boxes) with the A_8 features from the OOD contextual module (blue boxes). The resulting features (F) are then processed by a dynamic module, which also takes in the output of the existing convolution layers (K_1) of the ×8 stage in the base network. The output of the module is then upsampled and concatenated with C_4 (Sec. IV-1) and both are fed to another dynamic module along with the output of the existing convolution layers (K_2) of ×4 stage in the base network. The final output of this module is then fed to the corresponding task-specific heads.

The PoDS base network targets only in-distribution classes. To expand the network's capabilities, we incorporate additional task-specific heads that can learn about both in-distribution and out-of-distribution classes. These heads consist of two sequential layers of 3×3 depthwise-separable convolutions, followed by a task-specific 1×1 predictor. The OOD semantic segmentation head uses a predictor with $N_{stuff} + N_{thing} + 1$ for segmentation labels. The OOD instance segmentation heads have two predictors: instance center prediction and instance center regression, which learn on *thing* + *void* regions. To train the semantic head, we use the weighted bootstrapped crossentropy loss (L_{sem}) . For the instance center prediction, we use the Mean Squared Error (MSE) loss (L_{cp}) to minimize the distance between the predicted heatmaps and the 2D Gaussian encoded groundtruth heatmaps. For instance center regression, we use the L1 loss.

5) Learning from alignment-mismatch: We train the semantic segmentation head SH_{in} from the PoDS base network only for known in-distribution classes and we train the PoDS head SH_{out} for an added OOD class. The SH_{in} consistently labels pixels with known semantic classes, irrespective of inor out-of-distribution object. During the training of SH_{out} , we aim to amplify the prediction discrepancies between $(SH_{in} \text{ and } SH_{out})$ for out-of-distribution class pixels while promoting consensus for in-distribution object predictions. To implement the alignment-mismatch strategy, we ensure the output dimensions of SH_{in} and SH_{out} match. Given SH_{in} has $(N_{stuff} + N_{thing}) \times H \times W$ channels and SH_{out} has $(N_{stuff} + N_{thing} + 1) \times H \times W$, we derive an extra channel for SH_{in} by taking the maximum across the semantic class dimension. We employ the following loss to foster alignmentmismatch between the two heads, depending on whether the

pixel belongs to an in-distribution or out-of-distribution object:

$$e_i = |s(SH_{in}(x_i)) - s(SH_{out}(x_i))|^2,$$
(5)

$$s_i = (1 - y_i)e_i,\tag{6}$$

$$d_i = y_i \max(0, m - e_i),\tag{7}$$

$$L_{s-am} = \frac{1}{2N} \sum_{i=1}^{N} s_i + d_i,$$
(8)

where N is the number of pixels, x_i is the input image pixel, y_i is the label that indicates out- or in-distribution class, m is the hyperameter and $s = \ln (1 + e^x)$ is the softplus acitvation function. We use the softplus to encourage that SH_{out} predicts void class for out-of-distribution pixels with higher logits compared to SH_{in} 's maximum logit. We use softplus to ensure SH_{out} predicts the ood class for out-of-distribution pixels with logits higher than the maximum logit from SH_{in} . Since softplus always yields positive output and weights of SH_{in} are frozen, the only way for SH_{out} to reduce the loss is by predicting out-of-distribution classes with high logits, especially when the margin hyperparameter m in the loss is sufficiently large.

For instance segmentation, we strive to foster alignmentmismatch between the instance center prediction and instance center regression heads. We achieve this by employing a similar loss as L_{s-am} but applied in the feature space to the features X_{in} and X_{out} prior to the predictor for respective heads. This separates in-distribution from out-of-distribution features, making it easier to perform the center prediction and center regression. We define instance segmentation losses as

$$e_i = |X_{j-in}^i - X_{j-out}^i|,$$
 (9)

$$L_{j-am} = \frac{1}{2N} \sum_{i=1}^{N} s_i + d_i, \qquad (10)$$

where X_{j-in}^i and X_{j-out}^i are the features computed at location i for the instance segmentation heads. s_i and d_i are the same as (6) and (7), respectively. $j \in [c, r]$ represents the instance center prediction or instance center regression heads, from which we obtain the losses L_{c-am} and L_{r-am} , respectively.

6) Data Augmentation: For training the PoDS architecture, we mix samples containing solely in-distribution classes with those that include both in- and out-of-distribution objects. To curate out-of-distribution samples, we source web images via specific keywords, ensuring they exclude known in- or out-ofdistribution objects from the Cityscapes-OOD and BDD100K-OOD test set. Using an unsupervised instance segmentation network [27], we generate pseudo instance masks for these images, facilitating the extraction and compilation of a diverse out-of-distribution (OOD) object repository. During training, images are either augmented with randomly positioned and scaled OOD objects or left as in-distribution. Additionally, the compiled OOD objects are split into two types based on similarity to known semantic classes. Initially, training focuses on vastly dissimilar objects such as hair dryers, with a progressive shift towards more similar objects such as monkeys, ensuring gradual learning from distinct to closely related OOD objects.

TABLE I: Panoptic out-of-distribution benchmarking results on the Cityscapes-OOD and BDD100K-OOD test set. Subscripts *out* and *in* refer to out-ofdistribution class and in-distribution classes, respectively. All scores in [%].

Model	Cityscapes-OOD			BDD100K-OOD		
	POD-Q	$PQ_{\textit{out}}$	PQ_{in}	POD-Q	$PQ_{\it out}$	PQ_{in}
MSP [12]	12.8	3.4	47.6	9.1	2.6	32.1
MaxLogit [13]	15.9	5.2	48.6	12.7	4.7	34.5
ODIN [20]	20.8	8.7	49.8	16.9	7.9	36.1
EPSON [5]	29.4	15.9	54.4	23.7	14.3	39.4
Meta-OOD [18]	41.7	31.3	55.6	34.5	28.6	41.6
DD-OPS [22]	46.1	36.1	58.7	38.0	33.2	43.5
PoDS(Ours)	53.4	45.9	62.2	42.3	38.7	46.3

V. EXPERIMENTAL EVALUATION

A. Training and Inference Protocol

We adopt a two-stage training approach for our network. Initially, we train the base layers of the PoDS network for 160,000 iterations on Cityscapes and 240,000 on BDD100K to instill strong in-distribution priors. Subsequently, these base layers are frozen and the other components of the PoDS network are trained using the data augmentation techniques outlined in Sec. IV-6 for 90K and 150K iterations on Cityscapes and BDD100K, respectively. For each training phase, we employ the Adam optimizer with a poly learning rate schedule, setting the initial learning rates at 0.001 for Cityscapes and 0.005 for BDD100K. We optimize the following loss functions for training the network:

$$L = L_{ocm} + L_{sem} + \alpha L_{cp} + \beta_1 (L_{cr} + L_{s-ad}) + \beta_2 (L_{c-ad} + L_{r-ad}),$$
(11)

where $\alpha = 200$, $\beta_1 = 0.01$ and $\beta_2 = 0.001$. All of the individual losses are defined in Sec. IV. We set the margin hyperparameter m to 50. During inference, we use the same post-processing as [8] with the semantic and instance segmentation head predictors that learn with the inclusion of OOD class.

B. Benchmarking Results

In Tab. I, we compare the performance of our PoDS architecture with the baselines on the Cityscapes-OOD and BDD100K-OOD test sets. The first three baselines, MSP [12], MaxLogit [13], and ODIN [20], adapt any panoptic segmentation network for out-of-distribution segmentation without modifications. We observe that they perform poorly compared to other reported methods as the task also requires identifying instances of out-of-distribution objects and not only obtaining dense predictions for them. While thresholding confidence scores from these baselines enhances OOD object sensitivity, it often results in fragmented detections and misclassifications of in-distribution objects as OOD. Consequently, these baselines are not ideal for directly employing them for panoptic out-ofdistribution segmentation. EPSON [5] mines labels from void regions to learn clusters for OOD objects. While it improves OOD detection and reduces in-distribution misclassification, its low POD-Q scores indicate limited generalization to unseen OOD objects during testing. Meta-OOD [18] emphasizes higher entropy for OOD predictions, whereas DD-OPS [22] refines

TABLE II: Evaluation of various architectural components in PoDS. Results are presented on the Cityscapes-OOD test set. Subscripts *out* and *in* refer to out-of-distribution class and in-distribution classes. All scores are in [%].

Model	POD-Q	PQ_{out}	PQ_{in}
M1	28.2	16.2	49.3
M2	26.9	15.3	47.5
M3	47.1	38.4	58.0
M4 (PoDS)	53.4	45.9	62.2

the utilization of void regions, rejecting objects using known classes and employing a class-agnostic classifier for OOD determination. PoDS employs a dual-head predictive setting to delineate the boundaries between in-distribution and out-of-distribution classes and to increase confidence in OOD object segmentation during training. By leveraging the alignment and mismatch between the heads, PoDS achieves the highest POD-Q scores of 53.4 on Cityscapes-OOD and 42.3 on BDD100K-OOD.

C. Ablation Study

1) Detailed Study on the PoDS Architecture: While developing the PoDS architecture, we incorporated various components to address specific challenges. Tab. II presents four model configurations, labeled M_i , to determine the impact of each component. The M1 configuration uses the base network PAPS* with OOD classes as an extra class, trained from scratch with data augmentation. We observe that M1 achieves a low POD-Q score of 28.2 as it tries to cover both indistribution and out-of-distribution classes, leading to poor generalization on the test set for unseen OOD classes. In M2, we incorporate the dual predictive head architecture of PoDS into M1 and train it with our alignment-mismatch loss. However, this leads to a drop of 1.3 in the POD-Q score compared to M1, indicating that the pretrained backbone does not encode OOD objects effectively enough for the new decoders and heads to learn. As a result, the alignmentmismatch loss hinders the performance of M2. In M3, we incorporate the OOD contextual module into M2. The notable performance improvement compared to M2 indicates that by learning highly discriminative features through the simplified task of OOD classification and segmentation, the dual predictive head, combined with the alignment-mismatch loss, prioritizes understanding what lies outside the in-distribution rather than trying to model the distribution of out-of-distribution objects. Finally, in M4, we incorporate the dynamic module into the non-pretrained decoders of M3. The results of M4, with a POD-Q score of 50.5, underscore the benefits of learning features not previously encompassed in the in-distribution feature space. As elaborated in Sec. IV-3, this is achieved by leveraging pretrained weight offsets and dynamically transitioning between the robust knowledge base of the pretrained decoder for indistribution classes and the decoder trained to recognize features in the presence of both in-distribution and out-of-distribution objects. Moreover, the improvements from M_{i-1} to M_i result not only from the new additions but also from their synergy with the existing modules. M4 embodies our proposed PoDS architecture.



Fig. 6: Qualitative panoptic out-of-distribution segmentation results of our proposed PoDS network in comparison to the state-of-the-art baseline DD-OPS [22] on Cityscapes-OOD (a, d) and BDD100K-OOD (b, c) datasets.

TABLE III: Evaluation of the top two baselines with PoDS' data augmentation. Results are presented on the Cityscapes-OOD test set. Subscript *out* and *in* refers to out-of-distribution and in-distribution classes. All scores are in [%].

Model	Data Augmentation	POD-Q	PQ_{out}	PQ_{in}
Meta-OOD [18] DD-OPS [22]		$ \begin{array}{c} 41.7 \\ 46.1 \end{array} $	$\begin{array}{c} 31.3\\ 36.1 \end{array}$	$55.6 \\ 58.7$
Meta-OOD [18] DD-OPS [22]		43.0 48.5	$32.7 \\ 39.4$	$56.5 \\ 59.8$

2) Impact of Data Augmentation Strategy: We evaluate our proposed augmentation learning strategy, which introduces diverse out-of-distribution objects not sourced from a fixed distribution. We apply this strategy to the top-performing baselines, Meta-OOD and DD-OPS, on the Cityscapes-OOD dataset. The results in Tab. III show that their POD-Q scores increase by 1.3 for Meta-ODD and 2.4 for DD-OPS postaugmentation without affecting the PQin performance. This increase stems from the progressive exposure of augmented OOD data that allows the network to initially discern objects distinctly different from the in-distribution objects. However, comparing these improvements with M1 and M2's performance from Sec. V-C1, we infer that while augmentation does contribute to the improved performance of PoDS, the other network components of PoDS are equally crucial for its significant improvement.

3) Evaluation in Real-World OOD Scenarios: In this experiment, we evaluate the utility of our models and baselines in real-world settings using the Cityscapes dataset. We include two *thing* classes, bicycle, and motorcycle, from the eight Cityscapes *thing* classes as part of the OOD class. We exclude any image from the training set containing at least one instance of this OOD class, reducing the training set from 2975 to 2620 images. This exclusion ensures that the bicycle and motorcycle TABLE IV: Evaluation of best panoptic out-of-distribution segmentation methods on the Cityscapes val set. Subscript *out* and *in* refers to out-of-distribution and in-distribution classes. Subscript *base* refers to the base panoptic segmentation network. All scores are in [%].

(a) Panoptic out-of-distribution performance (b) Influence of OOD seg. on using bicycle and motorcycle classes as OOD. in-distribution performance.

Model	POD-Q	PQout	PQ_{in}	Model	PQ	PQ_{base}
Meta-OOD [18] DD-OPS [22] PoDS	$\begin{array}{c c} 39.1 \\ 44.7 \\ 50.3 \end{array}$	$27.1 \\ 33.4 \\ 39.8$	$56.3 \\ 59.8 \\ 63.6$	Meta-OOD [18] DD-OPS [22] PoDS	$ \begin{array}{c} 60.7 \\ 62.5 \\ 63.1 \end{array} $	$63.9 \\ 63.9 \\ 63.7$

TABLE V: Performance of PoDS models trained on Cityscapes but evaluated on BDD100K val set and BDD100K-OOD test set. All scores are in [%].

Training	Method	Evaluation Dataset				
Dataset		BDD100K BDD100K-OOD			OD	
		PQ	POD-Q	PQ_{in}	PQ_{out}	
Cityscapes	PAPS* PoDS	$39.6 \\ 38.9$	28.1	38.2	20.6	

classes are treated as unseen OOD objects during evaluation. The results, presented in Tab. IVa demonstrate that PoDS consistently outperforms the top two baselines by a substantial margin, reinforcing the findings from Tab. I and underscoring its applicability to real-world OOD scenarios. In Sec. S.3, we further qualitatively demonstrate the generalization ability of PoDS in real-world driving scenarios using our in-house data collected in Freiburg.

4) Influence of OOD Segmentation on In-Distribution Performance: We first study the impact of learning panoptic out-ofdistribution segmentation on network performance when only in-distribution classes are present in the input. We compare with the top three methods: Meta-OOD, DD-OPS, and PoDS, and also report the performance of their base panoptic segmentation networks. From the results shown in Tab. IVb, we observe that the PQ score of Meta-OOD substantially decreases, while DD-OPS and PoDS show a smaller drop in performance. However, PoDS shows the least drop of 0.6, demonstrating the ability to segment out-of-distribution objects while preserving the indistribution class knowledge. Subsequently, we evaluate the generalization ability of PoDS by training it on the Cityscapes dataset and evaluating it on the BDD100K dataset. Results from this experiment presented in Tab. V show that PoDS performs nearly as well as its base network PAPS*, achieving a POD-Q score of 28.1 on BDD100K-OOD and a PQ of 38.9 on BDD100K. This highlights the ability of PoDS to infer known semantic class boundaries from Cityscapes, constrained only by its base network's performance. We anticipate further advancements in this field by the robotics community will surpass these limitations in the future.

D. Qualitative Evaluations

We qualitatively compare the performance of PoDS with the best-performing baseline DD-OPS [22] as illustrated in Fig. 6. We observe that DD-OPS misclassifies OOD objects with known semantic classes, while PoDS excels at distinguishing them. PoDS exploits its dynamic module and the alignment-mismatch strategy to identify OOD features based on known semantic characteristics, enabling it to accurately distinguish between OOD objects, and bicycles and cars. However, we observe that PoDS struggles with cluttered OOD objects and is limited by its base network, as shown in Fig. 6 (d), misclassifying a bus as *stuff* due to its size and sample constraints. We hope that this work encourages solutions in the future to address these limitations.

VI. CONCLUSION

In this work, we introduced the panoptic out-of-distribution segmentation task, proposed two suitable datasets, established an interpretable evaluation metric, and adapted several openset and semantic out-of-distribution segmentation methods for baselines. We also proposed the novel PoDS architecture, which sets a new benchmark in performance. It also demonstrates the feasibility of incorporating OOD segmentation without a significant drop in in-distribution performance. We presented an extended evaluation of each module that we used in our network with quantitative and qualitative evaluations that demonstrate their utility. Our novel framework shows the feasibility of this crucial and holistic scene parsing task and we aim that our publicly released datasets and benchmark facilitate further research in this direction.

REFERENCES

- N. Vödisch, D. Cattaneo, W. Burgard, and A. Valada, "Continual slam: Beyond lifelong simultaneous localization and mapping through continual learning," in *The Int. Symposium of Robotics Research*, 2022, pp. 19–35.
- [2] N. Gosala, K. Petek, P. L. Drews-Jr, W. Burgard, and A. Valada, "Skyeye: Self-supervised bird's-eye-view semantic mapping using monocular frontal view images," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2023, pp. 14901–14910.
- [3] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár, "Panoptic segmentation," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2019, pp. 9404–9413.

- [4] D. Bozhinoski, D. Di Ruscio, I. Malavolta, P. Pelliccione, and I. Crnkovic, "Safety for mobile robotic systems: A systematic mapping study from a software engineering perspective," *Journal of Systems and Software*, vol. 151, pp. 150–179, 2019.
- [5] J. Hwang, S. W. Oh, J.-Y. Lee, and B. Han, "Exemplar-based open-set panoptic segmentation network," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2021, pp. 1175–1184.
- [6] R. Mohan and A. Valada, "Efficientps: Efficient panoptic segmentation," *Int. Journal of Computer Vision*, vol. 129, no. 5, pp. 1551–1579, 2021.
- [7] A. Kirillov, R. Girshick, K. He, and P. Dollár, "Panoptic feature pyramid networks," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2019, pp. 6399–6408.
- [8] B. Cheng, M. D. Collins, Y. Zhu, T. Liu, T. S. Huang, H. Adam, and L.-C. Chen, "Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation," in *Proc. of the IEEE Conf. on Computer Vision* and Pattern Recognition, 2020.
- [9] J. Uhrig, E. Rehder, B. Fröhlich, U. Franke, and T. Brox, "Box2pix: Single-shot instance segmentation by assigning pixels to object boxes," in *IEEE Intelligent Vehicles Symposium*, 2018, pp. 292–299.
- [10] R. Mohan and A. Valada, "Perceiving the invisible: Proposal-free amodal panoptic segmentation," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9302–9309, 2022.
- [11] —, "Amodal panoptic segmentation," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2022, pp. 21 023–21 032.
- [12] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," *arXiv preprint* arXiv:1610.02136, 2016.
- [13] D. Hendrycks, S. Basart, M. Mazeika, M. Mostajabi, J. Steinhardt, and D. Song, "Scaling out-of-distribution detection for real-world settings," *arXiv preprint arXiv:1911.11132*, 2019.
- [14] A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?" Advances in Neural Information Processing Systems, vol. 30, 2017.
- [15] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *Int. Conf. on Machine Learning*, 2016, pp. 1050–1059.
- [16] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," Advances in Neural Information Processing Systems, vol. 30, 2017.
- [17] H. Blum, P.-E. Sarlin, J. Nieto, R. Siegwart, and C. Cadena, "Fishyscapes: A benchmark for safe semantic segmentation in autonomous driving," in *Proc. of the IEEE/CVF Int. Conf. on Computer Vision Workshops*, 2019.
- [18] R. Chan, M. Rottmann, and H. Gottschalk, "Entropy maximization and meta classification for out-of-distribution detection in semantic segmentation," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2021, pp. 5128–5137.
- [19] Y. Xia, Y. Zhang, F. Liu, W. Shen, and A. L. Yuille, "Synthesize then compare: Detecting failures and anomalies for semantic segmentation," in *Europ. Conf. on Computer Vision*, 2020, pp. 145–161.
- [20] S. Liang, Y. Li, and R. Srikant, "Enhancing the reliability of outof-distribution image detection in neural networks," *arXiv preprint* arXiv:1706.02690, 2017.
- [21] V. Besnier, A. Bursuc, D. Picard, and A. Briot, "Triggering failures: Out-of-distribution detection by learning from local adversarial attacks in semantic segmentation," in *Int. Conf. on Computer Vision*, 2021, pp. 15701–15710.
- [22] H.-M. Xu, H. Chen, L. Liu, and Y. Yin, "Dual decision improves open-set panoptic segmentation," in *British Mac. Vision Conf.*, 2022.
- [23] A. Gupta, P. Dollar, and R. Girshick, "Lvis: A dataset for large vocabulary instance segmentation," in *Proc. of the IEEE Conf. on Computer Vision* and Pattern Recognition, 2019, pp. 5356–5364.
- [24] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2016, pp. 3213–3223.
- [25] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving dataset for heterogeneous multitask learning," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2020, pp. 2636–2645.
- [26] J. Xu, Y. Pan, X. Pan, S. Hoi, Z. Yi, and Z. Xu, "Regnet: self-regulated network for image classification," *IEEE Transactions on Neural Networks* and Learning Systems, 2022.
- [27] X. Wang, Z. Yu, S. De Mello, J. Kautz, A. Anandkumar, C. Shen, and J. M. Alvarez, "Freesolo: Learning to segment objects without annotations," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2022, pp. 14 176–14 186.

Panoptic Out-of-Distribution Segmentation

- Supplementary Material -

Rohit Mohan, Kiran Kumaraswamy, Juana Valeria Hurtado, Kürsat Petek, and Abhinav Valada

In this supplementary material, we begin with a detailed description of the POD-Q metric in Sec. S.1, followed by additional details on our PoDS architecture in Sec. S.2. Finally, in Sec. S.3, we provide further evidence supporting the pertinence of our proposed panoptic out-of-distribution (PoDs) task in real-world scenarios.

S.1. PANOPTIC OUT-OF-DISTRIBUTION QUALITY

The objective of panoptic out-of-distribution segmentation is to accurately classify and segment both in-distribution and out-of-distribution objects in a scene. To effectively evaluate the performance of this task, we need to equally assess the performance of both object categories while having the ability to distinguish the category-specific results. Furthermore, an ideal evaluation metric should be interpretable and easy to implement, promoting transparency and simplicity. In this direction, we propose the Panoptic Out-of-Distribution Quality (POD-Q), a metric based on the popular panoptic quality (PQ) metric [3]. To compute the POD-Q metric, we first compute PQ_{OOD} as the PQ for the OOD classes $o \in O$ given the predicted object segments P and their ground truth object segments G as follows:

$$PQ_{OOD} = \frac{\sum_{(p,g)\in TP_o, IoU(p,g)} IoU(p,g)}{|TP_o| + \frac{1}{2}|FP_o| + \frac{1}{2}|FN_o|},$$
(1)

Subsequently, we compute the PQ score for all the indistribution semantic classes PQ_{in} as

$$PQ_{in} = \frac{1}{|C|} \sum_{c \in C} \frac{\sum_{(p,g) \in TP_c, IoU(p,g)} IoU(p,g)}{|TP_c| + \frac{1}{2}|FP_c| + \frac{1}{2}|FN_c|}, \qquad (2)$$

where C is the set of in-distribution semantic classes. For both PQ_{OOD} and PQ_{in} , the true positives (TP_i) , false positives (FP_i) , and false negatives (FN_i) are defined as

$$TP_i = \{ p_i \in \{P\} \mid IoU(p_i, g_i) > 0.5, \forall g_c \in \{G\} \}, \quad (3)$$

$$FP_i = \{ p_i \in \{P\} \mid IoU(p_i, g) <= 0.5, \forall g \in \{G\} \}, \quad (4)$$

$$FN_i = \{g_i \in \{G\} \mid IoU(g_i, p) \le 0.5, \forall p \in \{P\}\}.$$
 (5)

where *i* takes the value of *o* for OOD class and *c* for indistribution semantic classes with $c \in C$. Finally, we compute POD-Q as the geometric mean between PQ_{OOD} and PQ_{in}, as

$$POD-Q = (PQ_{OOD} \times PQ_{in})^{\frac{1}{2}}.$$
 (6)

Department of Computer Science, University of Freiburg, Germany. Project page: http://pods.cs.uni-freiburg.de

S.2. PODS ARCHITECTURE

This section describes the base network in more detail and the OOD contextual module of the PoDS architecture proposed in Fig. S.1.

A. Base Network

The PAPS* architecture which is the base network of PoDS, employs HRNet as its backbone which is designed to retain high-resolution information throughout the network. The backbone generates four parallel feature map outputs at scales $\times 4$, $\times 8$, $\times 16$, and $\times 32$ with respect to the input, named B₄, B₈, B₁₆, and B₃₂. These feature maps are further upsampled to a resolution of $\times 4$ and combined to create the C₄ feature maps. The outputs from the backbone are fed to both the semantic and instance decoders, which have a similar architecture. The decoders take B₃₂, B₁₆, and C₄ as inputs. The B_{32} feature maps are first upsampled to $\times 16$ resolution and concatenated with B_{16} . This result is then fed into the dense prediction cell (DPC). The output from the DPC is then further upsampled to $\times 8$ resolution and processed through two consecutive 3×3 depthwise-separable convolutions. In the next step, we upsample the output $(\times 4)$ and concatenate it with C4. This is then followed by two sequential 3×3 depth-wise separable convolutions before being fed to the task-specific heads. Following, each task-specific head has a similar design and consists of sequential layers of two 3×3 depthwise-separable convolutions, followed by a task-specific 1×1 predictor. In PoDS, the PAPS^{*} part of the network is pretrained on the in-distribution panoptic segmentation dataset and its weights remain frozen during the entire panoptic outof-distribution segmentation training.

B. OOD Contextual Module

The architecture of the OOD Contextual Module (OCM) is shown in Fig. S.1. It consists of two residual bottleneck blocks comprising repeating units that employ group convolutions similar to the fourth and fifth stages in Regnet and a decoder. The output from the last layer of stage 2 of the backbone is processed by the first block of OCM, then concatenated with the output from the last layer of stage 3 and passed to the second block. Subsequently, the output of the second block (O_{ocm}) is fed to a global average pooling layer which is followed by a 1×1 convolution layer that acts as a classification head. Following, the decoder takes O_{ocm} as input and upsamples it by $\times 2$ scale followed by two sequential 3×3 depthwiseseparable convolutions and an additional $\times 2$ scale upsampling. Subsequently, we process the resulting features A_8 in two



Fig. S.1: Illustration of our proposed PoDS architecture that consists of a shared backbone with an OOD contextual module and symmetrical task-specific decoder arranged in a dual configuration setup to facilitate an alignment-mismatch learning strategy. The shared backbone learns robust feature representations for in-distribution semantic categories while the OOD contextual module supports both global and local features for OOD objects. The network comprises symmetrical semantic and instance decoders that include dynamic modules to adaptively balance the features between in- and out-distribution representations.

branches. The first branch uses a 1×1 convolution layer as an auxiliary semantic segmentation head. The second branch processes the output with two sequential 3×3 depthwiseseparable convolutions followed by $\times 2$ scale upsampling and 1×1 convolution layer as a second auxiliary semantic segmentation head for the upsampled features.

S.3. GENERALIZATION IN REAL-WORLD SCENARIOS

In this section, we aim to demonstrate the challenges associated with panoptic out-of-distribution segmentation and demonstrate the ability of our proposed network to segment OOD objects, as well as panoptic segmentation of in-distribution classes. To do so, we collect sequences of RGB images comprising real-world scenes captured in driving scenarios. We use the recorded data to qualitatively compare the performance of a conventional panoptic segmentation network and the proposed PoDs architecture. This comparison provides valuable insights into the differences between the two networks in scenarios that involve the segmentation of out-of-distribution objects. The results of this comparison show the ability of our proposed network to reason about objects that are very different from those presented during training. As a result, when having real-world images as an input, our network can provide a segmentation mask for unknown objects where the panoptic segmentation network incorrectly classifies them as background.

A. Evaluation in Real-World Scenarios

Panoptic segmentation is crucial for robot perception, as it enables robots to comprehend visual scenes they encounter by semantically segmenting and distinguishing instances from each other. Despite the holistic scene understanding provided by panoptic segmentation, it still presents limitations when identifying and segmenting out-of-distribution objects and maintaining the panoptic segmentation quality simultaneously. Panoptic out-of-distribution segmentation aims to address the gap between the current models trained for in-distribution tasks and the demands of robot perception in real-world scenarios. This task enables robots to recognize objects not seen during training, facilitating their operation in dynamic and unstructured environments with greater flexibility. Furthermore, this task also allows robots to preserve the panoptic segmentation quality. Panoptic out-of-distribution segmentation represents a significant step forward in making robots more suitable for realworld scenarios, thus improving their capability to comprehend visual scenes after deployment.

B. Data Description

In order to demonstrate the significance of panoptic out-ofdistribution segmentation in practical settings, we conducted an additional collection of RGB data using a vehicle equipped with a sensor array using a FLIR Blackfly 23S3C camera with a resolution of 1920×800 pixels. The collected data consists of video sequences, as opposed to isolated images found in the Cityscapes-OOD and BDD100K-OOD datasets. Similar to these datasets, the collected data depicts driving scenarios relevant to autonomous driving applications. The driving scenes are composed of in-distribution and out-out-distribution objects. For in-distribution, the scene contains common objects such as cars, buildings, and vegetation. Additionally, for out-ofdistribution, we placed uncommon objects, such as a fan, teddy bear, chair, kettle, suitcase, bottle, and trash bin on the road and sidewalk.

C. Experimental Setup

We use (Proposal-free Amodal Panoptic Segmentation) PAPS and PoDS trained on Cityscapes to represent the results while using a panoptic segmentation network and a panoptic outof-distribution segmentation network, respectively. We train both models using the Cityscapes dataset with images with a resolution of 2048×1024 . We evaluate on our collected data that has images with a resolution of 1920×800 . Given the



Fig. S.2: Qualitative comparison of the PAPS panoptic segmentation network with our PoDS panoptic out-of-distribution segmentation network on real-world scenes featuring out-of-distribution objects. The results highlight the feasibility and importance of panoptic out-of-distribution segmentation for safety-critical applications.

camera resolution difference, as well as the domain difference between the dataset and our collected data obtaining an accurate output on these images is challenging. The low illumination conditions add further complexity to the panoptic out-ofdistribution prediction. As a result, our collected data with realworld scenes is suitable to test the quality of the segmentation after deployment.

D. Qualitative Results

We present qualitative comparisons of PAPS and our proposed PoDS architecture in the supplementary video. Additionally, we present results in Fig. S.2. We observe that both models face challenges inherent to panoptic segmentation as well as challenges due to domain, dataset, and camera setup changes. The PAPS model is also unable to identify and segment out-ofdistribution objects effectively. The absence of a mechanism for categorizing novel objects as unknown, results in PAPS misclassifying a suitcase as a car and traffic sign in Fig. S.2 (a), and all out-of-distribution (OOD) objects, either part of road or sidewalk in Fig. S.2 (b, c, d). In contrast, our PoDS network generalizes from learning specific objects in the training data to unseen objects in the real world. Our network is able to identify more uncommon objects and correctly classify and segment the pixels corresponding to classes such as suitcase, trash bin, and chair. The qualitative results are promising and demonstrate the feasibility of our task as well as the benefits of learning to segment out-of-distribution objects in the scene for safety-critical applications.