Waymo Open Dataset: Panoramic Video Panoptic Segmentation

Jieru Mei[1], Alex Zihao Zhu[2(✉)], Xinchen Yan[2], Hang Yan[2], Siyuan Qiao[3], Liang-Chieh Chen[3], and Henrik Kretzschmar[2]

[1] Johns Hopkins University, Baltimore, USA
[2] Waymo LLC, Mountain View, USA
alexzzhu@waymo.com
[3] Google Research, Mountain View, USA

Abstract. Panoptic image segmentation is the computer vision task of finding groups of pixels in an image and assigning semantic classes and object instance identifiers to them. Research in image segmentation has become increasingly popular due to its critical applications in robotics and autonomous driving. The research community thereby relies on publicly available benchmark dataset to advance the state-of-the-art in computer vision. Due to the high costs of densely labeling the images, however, there is a shortage of publicly available ground truth labels that are suitable for panoptic segmentation. The high labeling costs also make it challenging to extend existing datasets to the video domain and to multi-camera setups. We therefore present the Waymo Open Dataset: Panoramic Video Panoptic Segmentation, a large-scale dataset that offers high-quality panoptic segmentation labels for autonomous driving. We generate our dataset using the publicly available Waymo Open Dataset (WOD), leveraging the diverse set of camera images. Our labels are consistent over time for video processing and consistent across multiple cameras mounted on the vehicles for full panoramic scene understanding. Specifically, we offer labels for 28 semantic categories and 2,860 temporal sequences that were captured by five cameras mounted on autonomous vehicles driving in three different geographical locations, leading to a total of 100k labeled camera images. To the best of our knowledge, this makes our dataset an order of magnitude larger than existing datasets that offer video panoptic segmentation labels. We further propose a new benchmark for Panoramic Video Panoptic Segmentation and establish a number of strong baselines based on the DeepLab family of models. We have made the benchmark and the code publicly available, which we hope will facilitate future research on holistic scene understanding. Our dataset can be found at: waymo.com/open.

Keyword: Panoramic video panoptic segmentation

J. Mei—Work done as an intern at Waymo.

1   Introduction

Semantic visual scene understanding has been studied extensively for decades
in the field of computer vision [17,37,61,68,77,84]. Researchers have tackled
tasks of varying difficulty, ranging from segmenting distinct objects in individ-
ual camera images [9,24,26,44] to tracking and segmenting multiple objects
in videos [14,69,75]. Robotic applications, such as autonomous driving, have
led to new challenges and opportunities for semantic visual scene understand-
ing [12,21].

Modern autonomous vehicles tend to be equipped with multiple cameras and
LiDAR scanners. The cameras provide rich semantic information about the scene,
whereas the LiDAR scanners capture sparse, but geometrically highly accurate
information. Autonomous vehicles need to be able to fuse and interpret the data
stream from multiple sensors to build and maintain over time an accurate and
consistent estimate of the world. One challenge when tracking and segmenting
multiple objects is that objects of interest may leave the field of view of a camera
to enter the field of view of another camera across consecutive video frames.

In this paper, we study the new task of video panoptic segmentation [32,35]
for autonomous vehicles equipped with multiple cameras. See Fig. 1 for an illus-
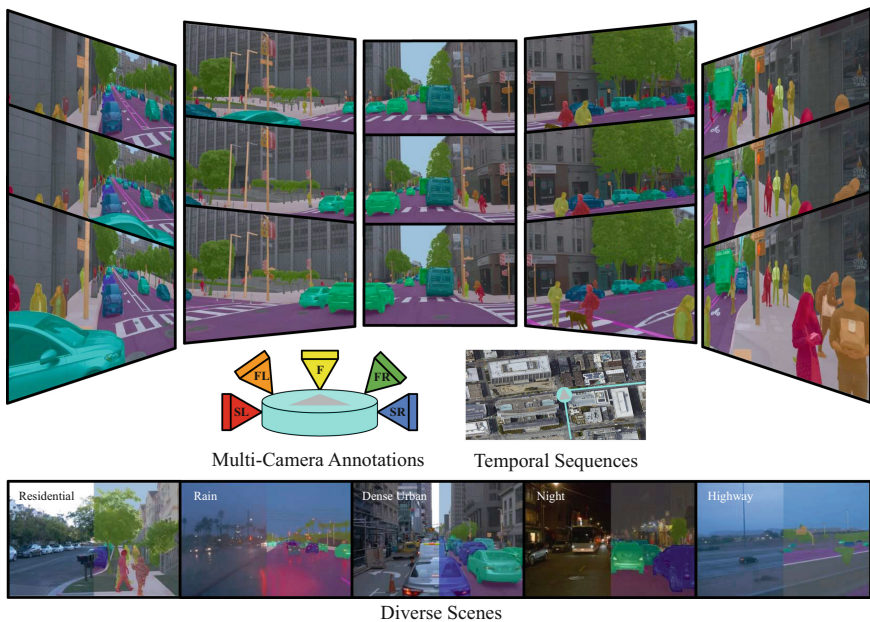
Fig. 1. We provide panoptic segmentation labels for 100k camera images of the Waymo
Open Dataset. Our dataset is grouped into 2,860 temporal sequences captured by
five cameras, mounted on autonomous vehicles driving in three geographical locations.
Instance segmentation labels are consistent both across cameras and over time. Our
dataset offers diversity in terms of object classes, locations, weather, and time of day.

tration. Panoptic segmentation enables autonomous vehicles to reason about their surroundings in terms of semantic and geometry properties, such as fine-grained object contours. There are also important offboard applications, including auto-labeling [53,79,87] and camera sensor simulation [10,42,46]. On the one hand, most existing panoptic segmentation datasets [12,49] provide labels for individual camera images. This makes it difficult to train models that fuse information from multiple camera images, either temporally or by leveraging a multi-camera setup. On the other hand, datasets that provide panoptic segmentation labels for video data [32,74] tend to be scarce and much smaller than datasets for object detection and tracking for autonomous driving [21,64].

To bridge this gap, we present a new benchmark dataset for panoptic segmentation based on the popular Waymo Open Dataset (WOD). Specifically, we provide panoptic segmentation labels for video data that are consistent across five cameras mounted on the vehicles. We further present a benchmark that captures the task of multi-camera panoptic segmentation in video data for autonomous driving. Overall, we provide panoptic segmentation labels for 100k camera images, which we group into training (70%), validation (10%) and test (20%) sets. The training set consists of 2,800 sequences, each of which comprises labels for five cameras spanning 1.2 s and five temporal frames. In contrast, our validation and test sets consist of 60 longer sequences, in order to facilitate the evaluation of long-term tracking. Each validation and test sequence consists of 100 temporal frames, spanning the full 20 s of a scene, while also providing labels across all five cameras.

We extend the Segmentation and Tracking Quality (STQ) metric [74] to support our multi-camera setup by computing a weight for pixels depending on the cameras they correspond to. We also extend a state-of-the-art video panoptic segmentation method, ViP-DeepLab [54], to our multi-camera setup by training separate models on each camera view and by training a model on a panorama generated from all views. We present an extensive experimental evaluation on the proposed dataset and metric.

We have open-sourced our full dataset to enhance video panoptic segmentation research, while also opening up the field of panoramic video panoptic segmentation.

2   Related Work

Panoptic Segmentation. The task of panoptic segmentation [35] aims to unify semantic segmentation [26] and instance segmentation [24], requiring assigning a class label and instance ID to all pixels in an image. Modern panoptic segmentation systems could be roughly categorized into top-down (or proposal-based) [34,38,43,52,72,76] and bottom-up (or proposal-free) [11,20,70,71,83] approaches. Our adopted baseline methods belong to the bottom-up category.

Video Panoptic Segmentation. Extending panoptic segmentation to the video domain, Video Panoptic Segmentation (VPS) [32] requires generating the

instance tracking IDs (*i.e.*, temporally consistent instance IDs) along with panoptic segmentation results across video frames. Current VPS datasets are small scale in terms of semantic classes and sizes. Specifically, Cityscapes-VPS [32] sparsely annotates (every five frame) Cityscapes [12] video sequences, resulting in only 3,000 frames with 19 semantic classes for training and testing. Recently, STEP [74] extends KITTI-MOTS [21,69] and MOTS-Challenge [14,69] for VPS. However, their annotated datasets are still small-scale (18K annotated frames with 19 semantic classes for KITTI-STEP, and 2K frames with 8 classes for MOTChallenge-STEP), and the video sequences are only captured by a single front-view camera. On the other hand, our annotated dataset presents the first large-scale VPS annotations and extends to the multi-camera scenario.

Segmentation Benchmarks. There are other popular video segmentation benchmarks existing in the literature, *e.g.*, VSPW [47] for video semantic segmentation, while MOTS [69] and Youtube-VIS [82] for video instance segmentation. Our benchmark is also related to urban scene understanding, where typical benchmarks include [2,4–6,12,21,22,28,39–41,49,64,81,85,86,88]. Our work is most related to WildPASS [81], WildPPS [30] and UP-Drive [50], which also aim to endow machines with large field-of-view perception. However, building on top of the large-scale Waymo Open Dataset [64], our benchmark provides much more high-quality annotated video sequences.

Multi-Camera Multi-Object Tracking. Consistently tracking objects across multiple cameras, multi-camera multi-object tracking [1,3,13,15,19,27,56,58] has been a popular research topic in the computer vision community. Typical benchmarks [7,18,23,36,55,65,78] only track a single class (*e.g.*, people or vehicles) with bounding boxes, while our proposed benchmark demands for pixel-level tracking and segmentation for multiple classes.

Panoramic Semantic Segmentation. Panoramic semantic segmentation provides surround-view perception [48,62,66,80,81,89], but limited to semantic segmentation without temporal and instance-level understanding. Our work is similar, but additionally tackles video panoptic segmentation. Recently, [51,57] predict bird's-eye view semantic segmentation using multi-camera inputs.

Table 1. Dataset comparison. Our WOD:PVPS is a new large-scale panoramic video panoptic segmentation dataset. † indicates panoramas.

| Dataset statistics	WOD:PVPS (ours)	WildPASS [81]	WildPPS [30]	Cityscapes-VPS [32]	KITTI-STEP [74]	MOT-STEP [74]
# sequences	2860	-	-	500	50	4
# images	100,000	500†	40†	3,000	19,103	2,075
# tracking classes	8	-	-	8	2	1
# semantic classes	28	8	4	19	19	7
panoramic	✓	✓	✓	✗	✗	✗
panoptic	✓	✗	✓	✓	✓	✓
video panoptic	✓	✗	✗	✓	✓	✓

3   WOD:PVPS Dataset

In this section, we first recap the existing Waymo Open Dataset (WOD) [64], one of the largest and most diverse multi-sensor datasets in the autonomous driving domain. We leverage the existing data that comes with coarse-level annotations (e.g., 2D and 3D bounding boxes) as the foundation, and subsample images for our dataset. We then provide an overview of our WOD:PVPS dataset, including panorama generation, statistics of the semantic classes, and temporal frame sampling. Finally, we explain in details our hybrid scheme to address the challenges in multi-camera and video labeling. We obtain consistent instance IDs across temporal frames and cameras by associating the panoptic labels from each individual image with the existing box-level annotations.

3.1   Dataset Overview

The Waymo Open Dataset contains 1,150 scenes, each consisting of 20 s of data captured 10 Hz (*i.e.*, 10 frames per second, and thus 200 frames per scene). Each data frame in the dataset includes 3D point clouds from the LiDAR devices, images from five cameras (positioned at Front, Front-Left, Front-Right, Side-Left, and Side-Right), and ground truth 3D and 2D bounding boxes annotated by humans in the LiDAR point clouds and camera images, respectively. Each bounding box contains an ID that is unique to that object across the entirety of each scene. For the LiDAR data, this allows for tracking in the whole scene. For the camera data, these IDs are consistent within each camera's images only.

Built on top of the WOD, Our WOD:PVPS dataset consists of 100,000 images with panoptic segmentation labels using a prescribed train, validation, and test set split, subsampled from the existing 1.15 million images. In Table 1, we compare our proposed WOD:PVPS dataset with the public datasets for video panoptic segmentation. Our dataset is the only one that provides panoptic segmentation annotations that are consistent both across multiple cameras and across time. Furthermore, our dataset is much larger both in terms of number of frames and number of semantic classes than existing datasets [32,74,81].

Equirectangular Panorama. We reconstruct the equirectangular panorama (220° coverage from five cameras) by stitching each individual camera images as an alternative input format to our dataset. Specifically, we first use the extrinsics and intrinsics from the five cameras provided by WOD to unproject each pixel coordinates to the 3D space. We then set a virtual camera [63] located at the geometric mean of all five camera centers and compute the pixel colors by equirectangular projection from the 3D space with bilinear sampling. For pixels correspond to multiple camera views, we compute the weights based on the distance of each pixel in the panorama to each of the camera views' boundaries. For panoptic labels, we compute labels in each camera view given the camera parameters of five cameras and the virtual camera using the nearest sampling. Then we use the method in Qiao *et al.* [54] to stitch the panorama labels to maintain the view consistency. Finally, we fused the five panorama labels together based on the correspondences and the distances to the camera view's boundaries.
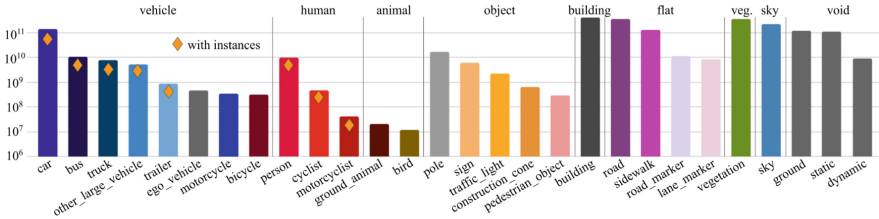
Fig. 2. Histogram of the 28 semantic categories in our dataset in terms of their pixel distributions. The vertical axis denotes the number of pixels for each class in log scale. We provide instance IDs for classes marked with diamonds.
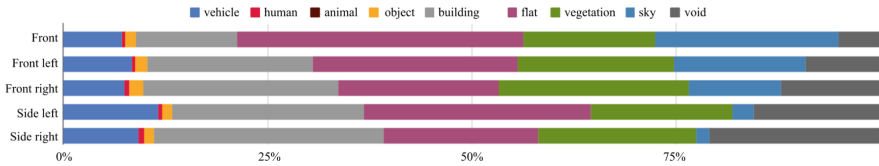
Fig. 3. Super-class distributions for each camera. Each camera sees a different distribution of classes, due to their fixed positions and different field-of-views.

There are more sophisticated methods [60,67] that leverage cross-frame information and the geometry captured from LiDAR sensors to potentially improve panorama generation. We leave this as an open research topic in the future.

Semantic Class Distribution. In total, our dataset contains 28 semantic categories, outlined with their frequency in pixels in Fig. 2. In addition, we provide instance IDs for most of the classes under the `vehicle` and `human` super-classes, as they are major dynamic categories in the autonomous driving space. We also outline the pixel distribution for each camera view in Fig. 3, where we see notable differences in the distributions in each camera. For example, the front camera covers more of `flat` (e.g., road surfaces) and `sky` pixels than the rest of the cameras, while the side left camera covers more `vehicle` pixels due to the ego-vehicle driving on the right hand side of the road. This analysis is important as machine learning models trained on the images captured by a single camera from the existing datasets may not necessarily generalize to the other cameras due to large domain gaps across different cameras. In contrast, our proposed task has an emphasis on the holistic scene understanding, which grants our WOD:PVPS dataset unique value to the research community.

Temporal Frame Sampling for Human Annotations. To maximize the diversity of the images on the training set, we subsample *sparsely* from each scene, labeling chunks of five-frame sequences from all the cameras. We start by randomly selecting 700 out of the 798 scenes. For each scene, which typically has 200 frames, we annotate *four* sets of five-frame sequences, starting at frame indices {25, 50, 125, 150} (*i.e.*, we pick 25th, 50th, 125th, and 150th frames as the first frame of each five-frame sequence for annotation). For each set, we further select frames with offsets {0, 4, 6, 8, 12} w.r.t.the first frame for annotations. For example, the first set of five-frame sequences will contain frames with indices

$\{25, 29, 31, 33, 37\}$. Our *sparse* sampling strategy facilitates a variety of different sequence lengths, allowing users to train on frame pairs with time difference as small as two frames (0.2 s) and as large as 12 frames (1.2 s). As a result, our training set contains groups of five temporal frames across all five cameras, yielding 2,800 sequences of 25 images (5 temporal frames × 5 cameras), or 70,000 images in total. Finally, we provide the associations between each instance ID and the corresponding 3D LiDAR bounding box, allowing us to compute very long associations (up to 13.7 s between all four sequences), if an object persists across multiple sequences in the same scene.

For the validation and test sets, we aim to enable the testing of long-term consistency across cameras and frames. We therefore *densely* sample frames 5 Hz from chunks of 100 frames across all cameras (*i.e.*, every other two frames are sampled in the 200 frame sequence). We select 20 and 40 scenes for validation and test sets by maintaining diversity in the location, density of object, and time of day distributions of WOD. In contrast to the training set, for each scene selected from the validation and test tests, we *densely* subsample the scenes for these splits by labeling every other frame, resulting in sequences with 100 temporal frames across all five cameras. In the end, our validation set contains annotations for 20 sequences of 500 images (100 temporal frames × 5 cameras), and our test set consists of 40 sequences of 500 images (or totally 10,000 and 20,000 annotations for validation and test sets, respectively). The test set annotations will not be made publicly available, but instead we will prepare a test server to evaluate the held-out test set, once the dataset is released.

3.2   Associating Instance IDs Across Cameras and Frames

In constructing the panoramic video panoptic segmentation dataset, ensuring the annotations have consistent instance IDs across cameras and temporal frames is one of the major challenges. Manual labeling is a straight-forward option, but is time-consuming and expensive at large scales. In addition, it is difficult to develop an effective labeling interface that allows human annotators to iteratively refine instance labels across cameras and temporary frames.

We instead assigned human annotators to label each camera image for panoptic segmentation separately and employed a hybrid scheme that leverages the existing coarse-level annotations in WOD. The coarse-level annotations include (i) 3D bounding boxes with corresponding IDs that are consistent across all frames and cameras; and (ii) 2D bounding boxes with IDs that are consistent across temporal frames, but annotated independently for each camera. Associations were then computed between each instance and its corresponding 3D LiDAR boxes and 2D camera boxes. Instances determined to correspond to the same object are then mapped to the same ID in all frames across cameras. A sample sequence from this process can be found in Fig. 4.

For a given frame with instance labels, 3D point clouds, and 3D bounding boxes, we associate instances with boxes by filtering the LiDAR points within
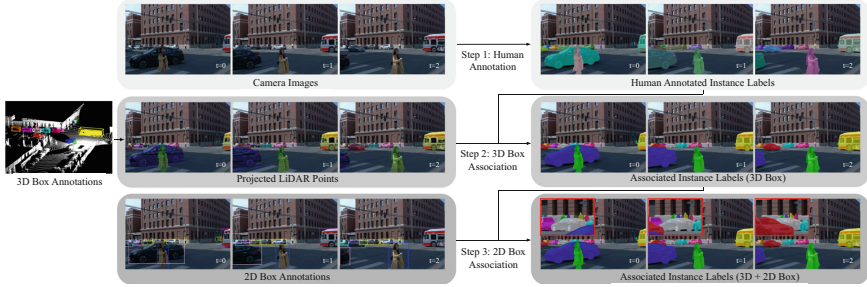
Fig. 4. Labeling and Association Overview. Human annotators first label each camera image for panoptic segmentation separately (step 1). LiDAR points within each ground truth 3D bounding box are then projected to each image, and associate with the single frame instance labels (step 2). For far-range instances without corresponding 3D bounding boxes, we associate the single frame instance labels over time using the ground truth 2D bounding boxes within each camera (step 3). New associations are highlighted in the zoomed-in views at the bottom.

each box, and projecting them onto the image. Association scores are then computed using IoU between the convex hull of the projected LiDAR points and each instance label. Bipartite matching is then applied to match each projected box with an instance label. For 3D driving scenes, points inside the bounding box almost entirely correspond to the instances inside of them, and so these projected LiDAR points have a high overlap with their corresponding instance masks in the image. Our label association step is related to the prior work [29,40], but, our association leverages the ground-truth labeled 3D boxes and only transfers instance IDs rather than fine-grained per-pixel labels.

There are, however, a small number of instances without corresponding LiDAR ground truth boxes due to occlusions, rolling shutter artifacts, and the limited range of the provided LiDAR scans (75 m). We apply an additional matching step by associating the 2D bounding boxes with our instance labels. First, we score matches between 2D boxes and instances by computing the IoU between each 2D box and the tightly-fitting bounding boxes around each instance mask, and then compute associations with bipartite matching.

For boxes with existing 3D associations, we extend these tracks by propagating the existing ID to all other instances that match with the same 2D box. This resolves cases where only a object track misses 3D associations in a few frames. Then, we assign the remaining boxes without any matches to the ID of their corresponding 2D box, if any. Finally, to capture any additional cross-camera associations, we project all of the camera views onto the panorama, and associate instances which overlap in this joint representation.

In order to identify any instances that are still not associated with any ground truth boxes after these steps, we provide an additional mask for these instance pixels indicating that they are not tracked, similar to the *crowd* mask used in single frame instance segmentation labels [12].

4   Benchmark and Evaluation Metrics

In this section, we first describe the task of Panoramic Video Panoptic Segmentation (PVPS). Then we review the evaluation metrics used in the literature, and propose a new metric designed for PVPS with an emphasis on consistent multi-object tracking and segmentation across multiple cameras.

4.1   Problem Definition

We represent a multi-camera video sequence with T frames and M independent camera views as $\{\mathbf{I}_i^{1:T}\}_{i=1}^M$, where $\mathbf{I}_i^t$ is the i-th camera view captured at the t-th time step in the video sequence. Along with the multi-view representation of the full scene, we define the panorama at t-th time step as $\mathbf{I}_{\mathrm{pano}}^t$. In the task of Panoramic Video Panoptic Segmentation (PVPS), we require a mapping f of every pixel (x, y, t, i) in the multi-camera video sequence to a semantic category $c \in \mathbf{C}$ and an instance ID z consistent across camera views and temporal frames. Here, (x, y, t, i) indicates the spatial coordinate (x, y) of the i-th camera view captured at the t-th time step, and $\mathbf{C}$ is the set of semantic categories. Accordingly, we define the mappings f_{id} and f_{sem} for a particular instance ID z and semantic category c in Eq. (1) and Eq. (2), respectively. The mapping functions are the building blocks of our proposed metric introduced in Sect. 4.2.

$$f_{\mathrm{id}}(z) = \{(x, y, i, t) | f(x, y, i, t) = (c, z), c \in \mathbf{C}\}, \tag{1}$$
$$f_{\mathrm{sem}}(c) = \{(x, y, i, t) | f(x, y, i, t) = (c, *), c \in \mathbf{C}\}. \tag{2}$$

Compared to the existing tasks including Video Panoptic Segmentation (VPS) and Panoramic Semantic Segmentation, the proposed task is more challenging in the following aspects. First, each individual camera has its own unique viewpoint and field-of-view such that the semantic class statistics are different across cameras (*e.g.*, see Fig. 3). This leads to a large domain gap between videos captured with different cameras. Second, the instance ID prediction, with the long-term consistency across *both* time and cameras, requires holistic scene understanding.

4.2   Evaluation Metrics

In this subsection, we overview the existing Video Panoptic Segmentation (VPS) metric: Segmentation and Tracking Quality (STQ) [74], which we extend to evaluate the Panoramic Video Panoptic Segmentation (PVPS) task.

VPS Metric. We use f and g to indicate the prediction and ground-truth mapping, respectively. We define the true positive associations (TPA) [45] of a specific instance as $\mathrm{TPA}(z_f, z_g) = |f_{\mathrm{id}}(z_f) \cap g_{\mathrm{id}}(z_g)|$, where z_f is the predicted instance, $z_g \in \mathbf{G}$ is the ground-truth instance, and $\mathbf{G}$ is the set containing all unique ground-truth instances across cameras and temporal frames. Similarly, false negative associations (FNA) and false positive associations (FPA) can be

Fig. 5. Visualization of the weights tensor for all cameras. Pixels in the blue region have weights 0.5 during evaluation, as they are covered by two cameras. (Color figure online)

defined to compute the Intersection over Union ($\mathrm{IoU_{id}}$) for evaluating tracking quality. Formally, STQ is defined as follows.

$$STQ = (AQ \times SQ)^{\frac{1}{2}}, \tag{3}$$
$$AQ = \frac{1}{|\mathbf{G}|} \sum_{z_g \in \mathbf{G}} \frac{1}{|g_{\mathrm{id}}(z_g)|} \sum_{z_f, |z_f \cap z_g| \neq \emptyset} \mathrm{TPA}(z_f, z_g) \times \mathrm{IoU_{id}}(z_f, z_g),$$
$$SQ = \frac{1}{|\mathbf{C}|} \sum_{c \in \mathbf{C}} \frac{f_{\mathrm{sem}}(c) \cap g_{\mathrm{sem}}(c)}{f_{\mathrm{sem}}(c) \cup g_{\mathrm{sem}}(c)}.$$

As defined in Eq (3), STQ fairly balances segmentation and tracking performance, and is suitable for evaluating video sequences of arbitrary length. The Association Quality (AQ) measures the association quality for tracking classes, while the Segmentation Quality (SQ) measures the segmentation quality for semantic classes. Specifically, AQ involves the $\mathrm{IoU_{id}}$ computation for predicted instance IDs (and further weighted by true positive associations to encourage long-term tracking [74]), while SQ is the typical semantic segmentation metric [16] (*i.e.*, mean $\mathrm{IoU_{sem}}$ for predicted semantic classes).

PVPS Metric. We propose to extend the metric STQ [74] for Panoramic Video Panoptic Segmentation (PVPS). However, naïvely adopting STQ for the multi-camera scenario results in a potential issue, where pixels in the overlapping regions covered by multiple cameras will be counted multiple times. Instead, we employ a simple and effective solution by exploiting the pixel-centric property of STQ. In particular, we weight each pixel prediction w.r.t.its coverage by the number of cameras, as determined by the mapping between the camera images and the panorama image. For example, if a pixel is covered by N cameras (in our dataset, $N = 2$), its prediction will contribute $1/N$ when computing AQ, and SQ. We name the resulting metrics as *weighted* STQ (wSTQ), since each pixel prediction takes a different weight depending on its coverage by the number of cameras. In Fig. 5, we visualize the weights for an example of five-camera images.

PS Metric. We also briefly review the metric PQ (panoptic quality) [35] for evaluating image Panoptic Segmentation (PS), since we will build image-level baselines purely trained with image panoptic annotations.

For a particular semantic class c, the sets of true positives ($\mathrm{TP_c}$), false positives ($\mathrm{FP_c}$), and false negatives ($\mathrm{FN_c}$) are formed by matching predictions z_f to the ground-truth masks z_g based on the IoU scores. A minimal threshold of

greater than 0.5 IoU is chosen to guarantee unique matching. Formally,

$$PQ_c = \frac{\sum_{(z_f, z_g) \in \text{TP}_c} \text{IoU}(z_f, z_g)}{|\text{TP}_c| + \frac{1}{2}|\text{FP}_c| + \frac{1}{2}|\text{FN}_c|}, \tag{4}$$

where the final PQ is then obtained by averaging PQ_c over semantic classes.

5   Experimental Results

In this section, we introduce our PVPS baselines, which exploit the property of multi-camera images by taking as input either individual camera views or panorama images (generated from all camera views). We then provide extensive experiments on the proposed dataset and metric.

5.1   ViP-DeepLab Extensions as PVPS Baselines

To tackle the new challenging PVPS task, we extend the state-of-art video panoptic segmentation method, ViP-Deeplab [54], to panoramic views.

Baseline Overview. For completeness, we first briefly review ViP-DeepLab [54]. ViP-DeepLab extends the state-of-art image panoptic segmentation model, Panoptic-DeepLab [11], to the video domain. Panoptic-DeepLab employs two separate prediction branches for semantic segmentation [9] and instance segmentation [31], respectively. Both segmentation results are then merged [83] to form the final panoptic segmentation result. To perform video panoptic segmentation, ViP-DeepLab adopts a two-frame image panoptic segmentation framework. Specifically, during training, ViP-DeepLab takes a pair of image frames as input and their panoptic segmentation ground-truths as training target. During inference, ViP-DeepLab performs two-frame image panoptic predictions at each time step, and continues the inference process for every two consecutive frames (*i.e.*, with one overlapping frame at the next time step) in a video sequence. The predictions in the overlapping frames are "stitched" together by propagating instance IDs based on mask IoU between region pairs (*i.e.*, if two masks have high IoU overlap, they will be re-assigned with the same instance ID), and thus temporally consistent IDs are obtained (see Fig. 4 of Qiao *et al.* [54] for an illustration). We refer this postprocessing as "panoptic stitching over time".

Baseline Extension for PVPS. We explore several ViP-DeepLab extensions for PVPS, which takes as input individual camera views or panorama images (generated from all camera views). The input types could be different during training and evaluation. Specifically, we define three training schemes: *View*, *Pano*, and *Ensemble-View*. The View scheme refers to the case where ViP-DeepLab is trained with images from all camera views, while Pano means the model is trained with full panorama images. The Ensemble-View scheme refers to the case where we have five camera-specific ViP-DeepLab models, each of which
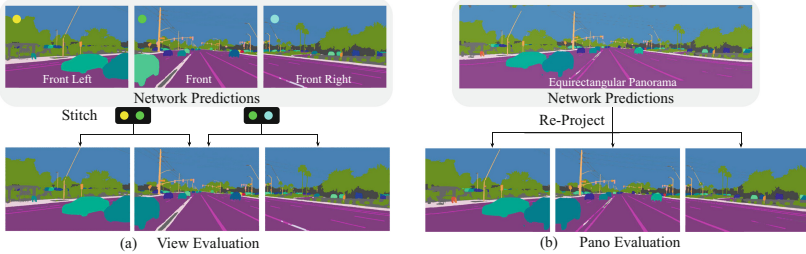
Fig. 6. We experiment with two evaluation schemes: (a) View and (b) Pano. The View evaluation scheme takes individual camera views as input and generates their panoptic predictions, which are then "stitched over cameras" to obtain consistent instance IDs between cameras. The Pano evaluation scheme takes as panorama images as input and generates panoramic panoptic predictions, which are then reprojected back to each camera for evaluation.

is trained and evaluated on their own camera images. We also have two evaluation schemes: *View* and *Pano*. The View scheme refers to the case where the trained model is fed with images from individual camera views and generates the corresponding panoptic predictions for each view. However, the predicted instance IDs are not consistent between cameras, since the predictions are made independently for each view. To generate consistent instance IDs between cameras, we propose a similar method to "panoptic stitching over time": if two masks have high IoU overlap in the overlapping regions between two cameras' field-of-view, we re-assign the same instance ID for them, resulting in the "panoptic stitching over cameras" post-processing method. For the Pano evaluation scheme, the model is fed with panorama images and generates panoramic panoptic predictions. We then re-project panoramic panoptic predictions onto each camera for evaluation. Note that, for the Pano evaluation scheme, the instance IDs are consistent between cameras by nature. We visualize the evaluation schemes in Fig. 6.

Implementation Details. We build our image-based and video-based baselines on top of Panoptic-DeepLab [11] and ViP-DeepLab [54], respectively, using the official code-base [73]. The training strategy follows Panoptic-DeepLab and ViP-DeepLab. Specifically, the models are trained with 32 TPU cores for 60k steps, batch size 32, Adam [33] optimizer and a poly schedule learning rate of 2.5×10^{-4}. We use an ImageNet-1K-pretrained [59] ResNet-50 [25] with stride 16 as the backbone (using atrous convolution [8]). For image-based methods, we use the crop size 1281×1921 during training, while, during inference, we use the whole image (or panorama). We use a similar strategy for the video-based methods, but we use a ResNet-50 backbone with stride 32 and crop size 641×961 due to memory constraints.
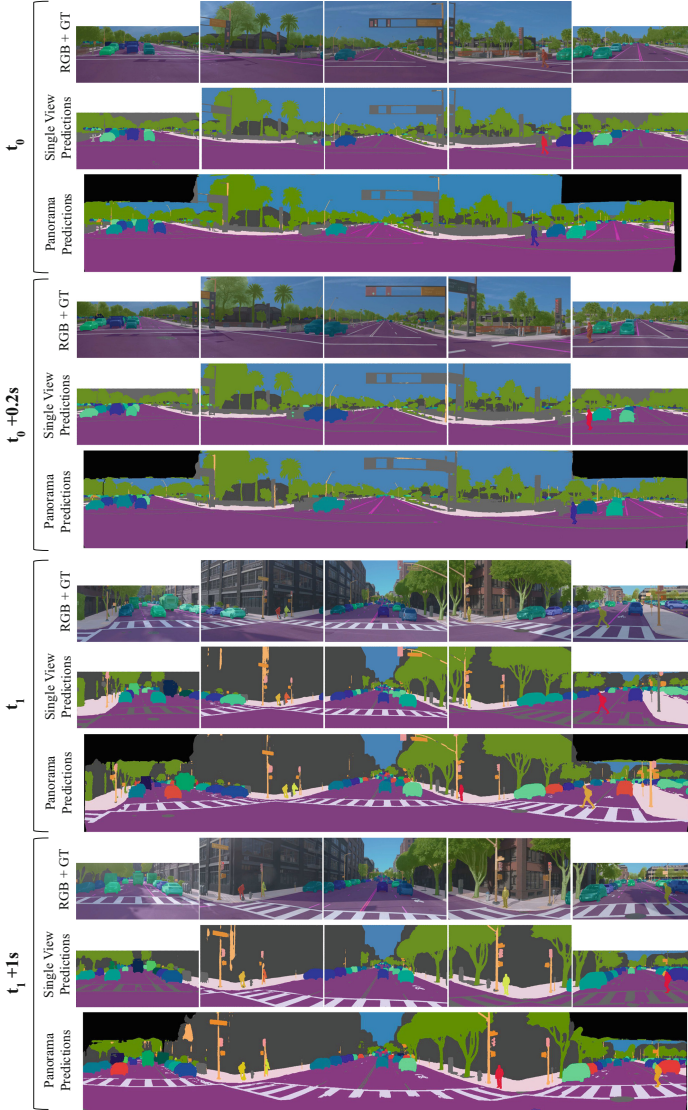
Fig. 7. Comparison of qualitative results from our baseline ViP-DeepLab [54] models over different time intervals. Results show models trained on single images with panoptic stitching over cameras, and trained directly on panorama images. Our baseline models show strong performance for the majority of the scene, although tracking small/distant objects and crowded scenes remains challenging.

5.2   Qualitative Evaluation

In Fig. 7, we provide results from our two ViP-DeepLab baselines, both trained and evaluated on single images and on panorama images (*i.e.*, two models using

View and Pano schemes, respectively, for both training and evaluation), over two (non-adjacent) temporal frames. As shown in the figure, the baseline models accurately track objects in very dense scenes. In addition, there are some qualitative benefits provided by the panorama model in these examples. In particular, the single view model has an inconsistent prediction on the crosswalk in the left and right images for the single view model, but the panorama model attains the full context of the scene and avoids this mistake. Further, the single view models fail to track the car crossing the front right and side right cameras at t_0, but the panorama is again able to track this object correctly.

5.3   Baseline Comparisons

Video-Based Baselines. In Table 2(a), we provide video-based baseline comparisons using ViP-DeepLab [54], evaluated by the proposed weighted STQ (wSTQ). We compare different training and evaluation schemes. When both evaluated with View scheme, training with View scheme performs better than training with Ensemble-View by 0.86% wSTQ. That is, training a single model with all the camera views performs better than training five camera-specific models with its own camera views. Also, when training with View scheme, using the Pano evaluation scheme degrades the performance by 2.91% wSTQ. When training with Pano scheme, using View scheme is better than Pano scheme for evaluation. Due to memory limits, we use a smaller crop size 641×961 of panorama images udring training (from the original resolution 1000×5875), which makes the training and evaluation settings inconsistent. However, the Pano scheme is still valuable, as we observed that the Pano scheme provides more consistent results for large objects that span across multiple cameras. The current best setting is trained and evaluated with the View scheme, reaching 17.78% wSTQ. We observe that our dataset is very challenging in terms of both tracking and segmentation, since our best wAQ is only 8.21% and best wSQ is 39.78%. In summary, two major factors contributed to the low wAQ on our datasets, namely, the scene complexity (e.g., number of objects) and the failure modes of our baselines (see the supplementary doc).

Image-Based Baselines. In Table 2(b), we provide image-based baseline comparisons using Panoptic-DeepLab [11], evaluated by image panoptic segmentation metric PQ [35] and semantic segmentation metric mIoU [16]. Basically, we observe the same trend of image-based baselines and video-based baselines.

5.4   Ablation Studies

Transferability of Models Between Viewpoints. We measure the ability to transfer models between viewpoints. As shown in Table 3, we have the following observations: First, all models, even trained on left side views, perform better on right side views. This phenomenon is due to the ego-vehicle driving on the

Table 2. Quantitative evaluation: the baselines can take different types of inputs: View: individual camera views; Pano: panoramas; Ensemble-View: camera-specific views. Results include (a) video-baseline using ViP-DeepLab, measured by wSTQ; and (b) image-baseline using Panoptic-DeepLab, measured by PQ and mIoU.

<table>
<tr><td colspan="5">(a) Video-baseline Comparison</td><td colspan="4">(b) Image-baseline Comparison</td></tr>
<tr><td>Training Scheme</td><td>Eval Scheme</td><td>wSTQ</td><td>wAQ</td><td>wSQ</td><td>Training Scheme</td><td>Eval Scheme</td><td>PQ</td><td>mIoU</td></tr>
<tr><td>Ensemble-View</td><td>View</td><td>16.92</td><td>7.61</td><td>37.33</td><td>Ensemble-View</td><td>View</td><td>35.70</td><td>48.15</td></tr>
<tr><td>View</td><td>View</td><td>**17.78**</td><td>**8.21**</td><td>38.46</td><td>View</td><td>View</td><td>**40.00**</td><td>**53.64**</td></tr>
<tr><td>View</td><td>Pano</td><td>14.87</td><td>6.13</td><td>36.04</td><td>View</td><td>Pano</td><td>33.65</td><td>50.61</td></tr>
<tr><td>Pano</td><td>View</td><td>17.56</td><td>8.11</td><td>38.04</td><td>Pano</td><td>View</td><td>38.93</td><td>51.65</td></tr>
<tr><td>Pano</td><td>Pano</td><td>15.72</td><td>6.22</td><td>**39.78**</td><td>Pano</td><td>Pano</td><td>36.32</td><td>52.19</td></tr>
</table>

Table 3. View transferability on our video-based baselines, measured by wSTQ. We evaluate models (1st column) trained on a specific view w.r.t.other camera views. The last row, MultiCamera, refers to the model trained with all camera views (*i.e.*, training scheme View), and the last column, All, denotes the evaluation set using all camera views (*i.e.*, evaluation scheme View).

Model \ Eval	Side left	Front left	Front	Front right	Side right	All
Side left	**18.79**	17.41	14.56	19.06	19.40	**16.31**
Front left	16.88	**18.39**	12.84	19.22	18.49	15.36
Front	16.58	18.02	14.54	18.55	18.96	15.98
Front right	16.56	17.36	**14.99**	**19.40**	19.16	16.18
Side right	17.91	16.50	13.16	18.23	**20.47**	15.65
MultiCamera	20.11	19.54	15.63	20.67	21.53	17.78

right side of the road, and providing wider scope, more instances, and smaller objects on the left side (*i.e.*, the left side views are more challenging). Second, the front camera performance is inferior compared to the other cameras. We hypothesize that the front camera captures more diverse and challenging views, *e.g.*, vehicles driving in multiple directions, more dynamic and smaller objects, making tracking more challenging.

6   Conclusion

In this work, we presented a new benchmark called WOD:PVPS. Our benchmark extends video panoptic segmentation to a more challenging multi-camera setting that requires consistent instance IDs both across cameras and over time. Our dataset is an order of magnitude larger than all the existing video panoptic segmentation datasets. We establish several strong baselines evaluated with a new metric, wSTQ, that takes multi-camera, multi-object tracking and segmentation into consideration. We will make our benchmark publicly available, and we hope that it will facilitate future research on panoramic video panoptic segmentation.

References

1. Baqué, P., Fleuret, F., Fua, P.: Deep occlusion reasoning for multi-camera multi-target detection. In: ICCV (2017)
2. Behley, J., et al.: Semantickitti: a dataset for semantic scene understanding of lidar sequences. In: ICCV (2019)
3. Berclaz, J., Fleuret, F., Turetken, E., Fua, P.: Multiple object tracking using k-shortest paths optimization. PAMI **33**(9), 1806–1819 (2011)
4. Brostow, G.J., Fauqueur, J., Cipolla, R.: Semantic object classes in video: a high-definition ground truth database. Pattern Recogn. Lett. **30**(2), 88–97 (2009)
5. Caesar, H., et al.: nuScenes: a multimodal dataset for autonomous driving. In: CVPR (2020)
6. Chang, M.F., et al.: Argoverse: 3D tracking and forecasting with rich maps. In: CVPR (2019)
7. Chavdarova, T., et al.: Wildtrack: a multi-camera HD dataset for dense unscripted pedestrian detection. In: CVPR (2018)
8. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Semantic image segmentation with deep convolutional nets and fully connected CRFs. In: ICLR (2015)
9. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. TPAMI **40**(4), 834–848 (2017)
10. Chen, Y., et al.: Geosim: realistic video simulation via geometry-aware composition for self-driving. In: CVPR (2021)
11. Cheng, B., et al.: Panoptic-deeplab: a simple, strong, and fast baseline for bottom-up panoptic segmentation. In: CVPR (2020)
12. Cordts, M., et al.: The cityscapes dataset for semantic urban scene understanding. In: CVPR (2016)
13. Dehghan, A., Modiri Assari, S., Shah, M.: GMMCP tracker: globally optimal generalized maximum multi clique problem for multiple object tracking. In: CVPR (2015)
14. Dendorfer, P., et al.: MOTChallenge: a benchmark for single-camera multiple target tracking. IJCV **129**(4), 845–888 (2020)
15. Eshel, R., Moses, Y.: Homography based multiple camera detection and tracking of people in a dense crowd. In: CVPR (2008)
16. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes (VOC) challenge. IJCV **88**(2), 303–338 (2010)
17. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. IJCV **59**(2), 167–181 (2004)
18. Ferryman, J., Shahrokni, A.: Pets 2009: dataset and challenge. In: 2009 Twelfth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, pp. 1–6. IEEE (2009)
19. Fleuret, F., Berclaz, J., Lengagne, R., Fua, P.: Multicamera people tracking with a probabilistic occupancy map. PAMI **30**(2), 267–282 (2007)
20. Gao, N., et al.: SSAP: single-shot instance segmentation with affinity pyramid. In: ICCV (2019)
21. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? The kitti vision benchmark suite. In: CVPR (2012)
22. Geyer, J., et al.: A2d2: Audi autonomous driving dataset. arXiv preprint arXiv:2004.06320 (2020)

23. Han, X., et al.: MMPTRACK: large-scale densely annotated multi-camera multiple people tracking benchmark (2021)
24. Hariharan, B., Arbeláez, P., Girshick, R., Malik, J.: Simultaneous detection and segmentation. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8695, pp. 297–312. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10584-0_20
25. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
26. He, X., Zemel, R.S., Carreira-Perpiñán, M.Á.: Multiscale conditional random fields for image labeling. In: CVPR (2004)
27. Hofmann, M., Wolf, D., Rigoll, G.: Hypergraphs for joint multi-view reconstruction and multi-object tracking. In: CVPR (2013)
28. Huang, X., Wang, P., Cheng, X., Zhou, D., Geng, Q., Yang, R.: The apolloscape open dataset for autonomous driving and its application. TPAMI **42**(10), 2702–2719 (2019)
29. Huang, X., Wang, P., Cheng, X., Zhou, D., Geng, Q., Yang, R.: The apolloscape open dataset for autonomous driving and its application. PAMI **42**(10), 2702–2719 (2020)
30. Jaus, A., Yang, K., Stiefelhagen, R.: Panoramic panoptic segmentation: towards complete surrounding understanding via unsupervised contrastive learning. In: 2021 IEEE Intelligent Vehicles Symposium (IV), pp. 1421–1427. IEEE (2021)
31. Kendall, A., Gal, Y., Cipolla, R.: Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In: CVPR (2018)
32. Kim, D., Woo, S., Lee, J.Y., Kweon, I.S.: Video Panoptic Segmentation. In: CVPR (2020)
33. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. In: ICLR (2015)
34. Kirillov, A., Girshick, R., He, K., Dollár, P.: Panoptic feature pyramid networks. In: CVPR (2019)
35. Kirillov, A., He, K., Girshick, R., Rother, C., Dollár, P.: Panoptic segmentation. In: CVPR (2019)
36. Kuo, C.-H., Huang, C., Nevatia, R.: Inter-camera association of multi-target tracks by on-line learned appearance affinity models. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6311, pp. 383–396. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15549-9_28
37. Ladický, Ľ, Sturgess, P., Alahari, K., Russell, C., Torr, P.H.S.: What, where and how many? Combining object detectors and CRFs. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6314, pp. 424–437. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15561-1_31
38. Li, Y., et al.: Attention-guided unified network for panoptic segmentation. In: CVPR (2019)
39. Liang, J., Homayounfar, N., Ma, W.C., Xiong, Y., Hu, R., Urtasun, R.: Polytransform: deep polygon transformer for instance segmentation. In: CVPR (2020)
40. Liao, Y., Xie, J., Geiger, A.: Kitti-360: a novel dataset and benchmarks for urban scene understanding in 2D and 3D. arXiv:2109.13410 (2021)
41. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
42. Ling, H., Acuna, D., Kreis, K., Kim, S.W., Fidler, S.: Variational amodal object completion. In: NeurIPS (2020)
43. Liu1, H., et al.: An end-to-end network for panoptic segmentation. In: CVPR (2019)

44. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR (2015)
45. Luiten, J., et al.: HOTA: a higher order metric for evaluating multi-object tracking. In: IJCV (2020)
46. Mallya, A., Wang, T.-C., Sapra, K., Liu, M.-Y.: World-consistent video-to-video synthesis. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12353, pp. 359–378. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58598-3_22
47. Miao, J., Wei, Y., Wu, Y., Liang, C., Li, G., Yang, Y.: VSPW: a large-scale dataset for video scene parsing in the wild. In: CVPR (2021)
48. Narioka, K., Nishimura, H., Itamochi, T., Inomata, T.: Understanding 3D semantic structure around the vehicle with monocular cameras. In: IEEE Intelligent Vehicles Symposium (IV), pp. 132–137. IEEE (2018)
49. Neuhold, G., Ollmann, T., Bulò, S.R., Kontschieder, P.: The mapillary vistas dataset for semantic understanding of street scenes. In: ICCV (2017)
50. Petrovai, A., Nedevschi, S.: Semantic cameras for 360-degree environment perception in automated urban driving. IEEE Trans. Intell. Transp. Syst. (2022)
51. Philion, J., Fidler, S.: Lift, splat, shoot: encoding images from arbitrary camera rigs by implicitly unprojecting to 3D. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12359, pp. 194–210. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58568-6_12
52. Porzi, L., Bulò, S.R., Colovic, A., Kontschieder, P.: Seamless scene segmentation. In: CVPR (2019)
53. Qi, C.R., et al.: Offboard 3D object detection from point cloud sequences. In: CVPR (2021)
54. Qiao, S., Zhu, Y., Adam, H., Yuille, A., Chen, L.C.: ViP-DeepLab: learning visual perception with depth-aware video panoptic segmentation. In: CVPR (2021)
55. Ristani, E., Solera, F., Zou, R., Cucchiara, R., Tomasi, C.: Performance measures and a data set for multi-target, multi-camera tracking. In: Hua, G., Jégou, H. (eds.) ECCV 2016. LNCS, vol. 9914, pp. 17–35. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-48881-3_2
56. Ristani, E., Tomasi, C.: Features for multi-target multi-camera tracking and re-identification. In: CVPR (2018)
57. Roddick, T., Cipolla, R.: Predicting semantic map representations from images using pyramid occupancy networks. In: CVPR (2020)
58. Roshan Zamir, A., Dehghan, A., Shah, M.: GMCP-tracker: global multi-object tracking using generalized minimum clique graphs. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7573, pp. 343–356. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33709-3_25
59. Russakovsky, O., et al.: ImageNet large scale visual recognition challenge. Int. J. Comput. Vision **115**(3), 211–252 (2015). https://doi.org/10.1007/s11263-015-0816-y
60. Schönberger, J.L., Zheng, E., Frahm, J.-M., Pollefeys, M.: Pixelwise view selection for unstructured multi-view stereo. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9907, pp. 501–518. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46487-9_31
61. Shi, J., Malik, J.: Normalized cuts and image segmentation. PAMI **22**(8), 888–905 (2000)
62. Song, S., Zeng, A., Chang, A.X., Savva, M., Savarese, S., Funkhouser, T.: Im2Pano3D: extrapolating 360 structure and semantics beyond the field of view. In: CVPR (2018)

63. Su, Y.C., Grauman, K.: Making 360 video watchable in 2D: learning videography for click free viewing. In: CVPR (2017)
64. Sun, P., et al.: Scalability in perception for autonomous driving: Waymo open dataset. In: CVPR (2020)
65. Tang, Z., et al.: Cityflow: a city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification. In: CVPR (2019)
66. Tateno, K., Navab, N., Tombari, F.: Distortion-aware convolutional filters for dense prediction in panoramic images. In: ECCV (2018)
67. Thrun, S., Montemerlo, M.: The graph slam algorithm with applications to large-scale mapping of urban structures. Int. J. Robot. Res. **25**(5–6), 403–429 (2006)
68. Tu, Z., Chen, X., Yuille, A.L., Zhu, S.C.: Image parsing: unifying segmentation, detection, and recognition. IJCV **63**(2), 113–140 (2005)
69. Voigtlaender, P., et al.: MOTS: multi-object tracking and segmentation. In: CVPR (2019)
70. Wang, H., Luo, R., Maire, M., Shakhnarovich, G.: Pixel consensus voting for panoptic segmentation. In: CVPR (2020)
71. Wang, H., Zhu, Y., Green, B., Adam, H., Yuille, A., Chen, L.-C.: Axial-DeepLab: stand-alone axial-attention for panoptic segmentation. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12349, pp. 108–126. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58548-8_7
72. Weber, M., Luiten, J., Leibe, B.: Single-shot panoptic segmentation. In: IROS (2020)
73. Weber, M., et al.: DeepLab2: A TensorFlow Library for Deep Labeling. arXiv: 2106.09748 (2021)
74. Weber, M., et al.: Step: segmenting and tracking every pixel. In: NeurIPS Track on Datasets and Benchmarks (2021)
75. Wu, Y., Lim, J., Yang, M.H.: Online object tracking: a benchmark. In: CVPR (2013)
76. Xiong, Y., et al.: UPSNet: a unified panoptic segmentation network. In: CVPR (2019)
77. Xu, C., Xiong, C., Corso, J.J.: Streaming hierarchical video segmentation. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7577, pp. 626–639. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33783-3_45
78. Xu, Y., Liu, X., Liu, Y., Zhu, S.C.: Multi-view people tracking via hierarchical trajectory composition. In: CVPR (2016)
79. Yang, B., Bai, M., Liang, M., Zeng, W., Urtasun, R.: Auto4d: learning to label 4D objects from sequential point clouds. arXiv preprint arXiv:2101.06586 (2021)
80. Yang, K., Hu, X., Bergasa, L.M., Romera, E., Wang, K.: Pass: Panoramic annular semantic segmentation. IEEE Trans. Intell. Transp. Syst. **21**(10), 4171–4185 (2019)
81. Yang, K., Zhang, J., Reiß, S., Hu, X., Stiefelhagen, R.: Capturing omni-range context for omnidirectional segmentation. In: CVPR (2021)
82. Yang, L., Fan, Y., Xu, N.: Video instance segmentation. In: ICCV (2019)
83. Yang, T.J., et al.: DeeperLab: Single-Shot Image Parser. arXiv:1902.05093 (2019)
84. Yao, J., Fidler, S., Urtasun, R.: Describing the scene as a whole: joint object detection, scene classification and semantic segmentation. In: CVPR (2012)
85. Yogamani, S., et al.: Woodscape: a multi-task, multi-camera fisheye dataset for autonomous driving. In: ICCV (2019)
86. Yu, F., et al.: BDD100K: a diverse driving dataset for heterogeneous multitask learning. In: CVPR (2020)

87. Zakharov, S., Kehl, W., Bhargava, A., Gaidon, A.: Autolabeling 3D objects with differentiable rendering of SDF shape priors. In: CVPR (2020)
88. Zendel, O., Schörghuber, M., Rainer, B., Murschitz, M., Beleznai, C.: Unifying panoptic segmentation for autonomous driving. In: CVPR (2022)
89. Zhang, C., Liwicki, S., Smith, W., Cipolla, R.: Orientation-aware semantic segmentation on icosahedron spheres. In: ICCV (2019)