Slot-VPS: Object-centric Representation Learning for Video Panoptic Segmentation
Supplementary Material

Yi Zhou[1], Hui Zhang[1], Hana Lee[2], Shuyang Sun[3], Pingjun Li[1], Yangguang Zhu[1],
ByungIn Yoo[2], Xiaojuan Qi[4*], Jae-Joon Han[2*]
[1]Samsung Research China - Beijing (SRC-B)
[2]Samsung Advanced Institute of Technology (SAIT), South Korea
[3]University of Oxford [4]The University of Hong Kong
{yi0813.zhou, hui123.zhang, hana.hn.lee, byungin.yoo, jae-joon.han}@samsung.com
kevinsun@robots.ox.ac.uk, xjqi@eee.hku.hk

A. Multi-scale Features Utilization.

To exploit sufficient spatio-temporal information, in Slot-VPS, VPR (Video Panoptic Retriever) is stacked for multiple stages and employed with multi-scale features. As shown in Figure S1, each stage, consisting of multiple VPR modules, is applied on features of certain scale, and features of different scales are fused through Fuse module across stages. Sinusoidal position embedding [1] is generated based on the input features of each stage. In Fuse module, the lower-resolution feature map is up-sampled, concatenated with the higher-resolution feature map, and fused via a 1×1 convolutional layer. The fused feature map will be fed into the subsequent stage.

With this multi-scale learning strategy, the difficulty of learning the unified slot representations is mitigated and the ability of handling multi-scale objects is improved.

B. Comparison between Slot-VPS and Related Methods.

The differences among Slot-VPS, Slot Attention [3], DETR [1], and previous VPS methods [2, 4] are shown in Table S1. The biggest advantage of Slot-VPS is that its VPR, consisting of Panoptic Retriever and Video Retriever, helps the panoptic slots acquire object information in each frame and makes it become consistent for the same object across frames. (a) On the image level, DETR's attention applies the softmax along the spatial dimension, discriminating only pixels instead of objects ($-$**4.2 PQ** *vs.* Panoptic Retriever). Slot Attention's learnable parameters are the mean and variance of the normal distribution, failing to handle complex objects in the real world ($-$**26 PQ** *vs.* Panoptic Retriever). (b) On the video level, previous VPS methods apply temporal attention on the feature level. This leads the temporal object information to be affected by the background features. Experiments show that replacing our Video Retriever with this strategy leads to **1.2 VPQ** drop.

		Learnable object params	Softmax dim	Temporal attn	Video object representation	Scenario
Image level	Slot Attention	distribution level	**slot**	NONE	NO	synthetic
	DETR	**object level**	spatial	NONE	NO	real-world
Video level	Prev VPS methods	NONE	NONE	feature	NO	real-world
	Slot-VPS (Ours)	**object level**	**slot**	**slot**	**YES**	**real-world**

Table S1. Comparison between Slot-VPS and related methods.

C. Visualization of Result Comparison on Cityscapes-VPS and VIPER.

Result comparison between VPSNet [2] and the proposed Slot-VPS on Cityscapes-VPS and VIPER are visualized in Figure S2, Figure S3, Figure S4 and Figure S5. It validates that our method can handle objects with different scales, achieve richer details in single frame and better temporal consistency across frames.

D. Limitation and Broader Impact.

(1) Slot-VPS currently predicts the object ID based on the correlations of panoptic slots across frames. Advanced architecture, loss, and regularization technologies may be explored to improve VPS performance without an ID head. (2) Slot-VPS unifies the video panoptic segmentation in terms of representations, however, it does not fully unify the entire training pipeline since the learning targets are still separated and individual losses and manually tuned loss weights are needed. This problem is challenging but maybe potentially solved by designing a new unified loss to the whole VPS task. (3) Current metric does not fully consider

*Corresponding author.

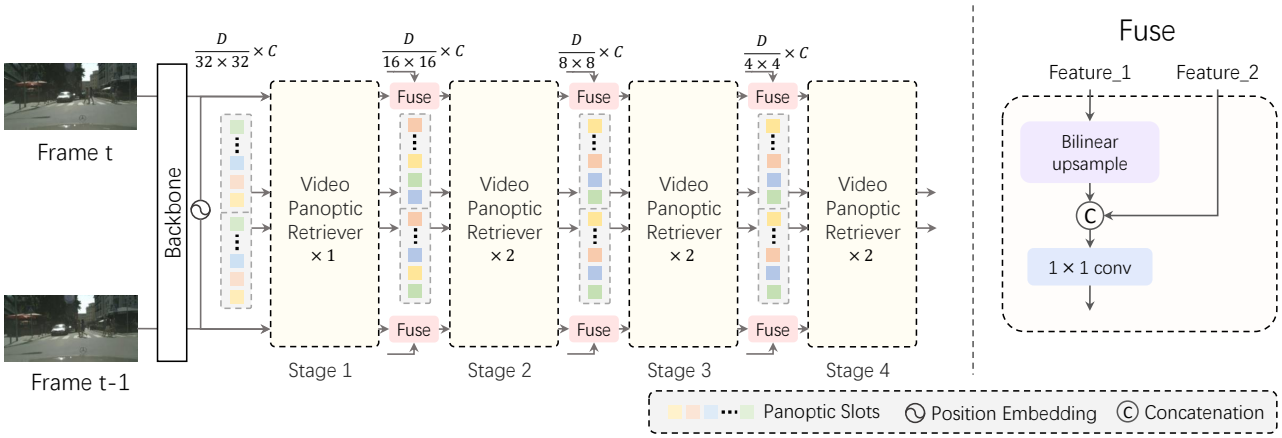

Figure S1. **Multi-scale feature utilization in the Slot-VPS.** Take two frames (t and t-1) as an example, four stages of Video Panoptic Retriever (VPR), consisting of $1, 2, 2, 2$ VPR modules respectively, are consecutively applied on four scales of multi-scale features extracted from the backbone. In each stage, VPR takes the features with its position embedding and panoptic slots as input and output the spatio-temporal coherent panoptic slots. Across stages, Fuse module is applied to generate fused features for later stage. Note that position embedding for later stages are omitted for brevity. D, C refer to the spatial size (height $\times$ width) and the number of channels of feature maps respectively.

the severity of prediction error, which is also an interesting research direction.

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020. 1

[2] Dahun Kim, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Video panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9859–9868, 2020. 1

[3] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *arXiv preprint arXiv:2006.15055*, 2020. 1

[4] Siyuan Qiao, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Vip-deeplab: Learning visual perception with depth-aware video panoptic segmentation. *arXiv preprint arXiv:2012.05258*, 2020. 1

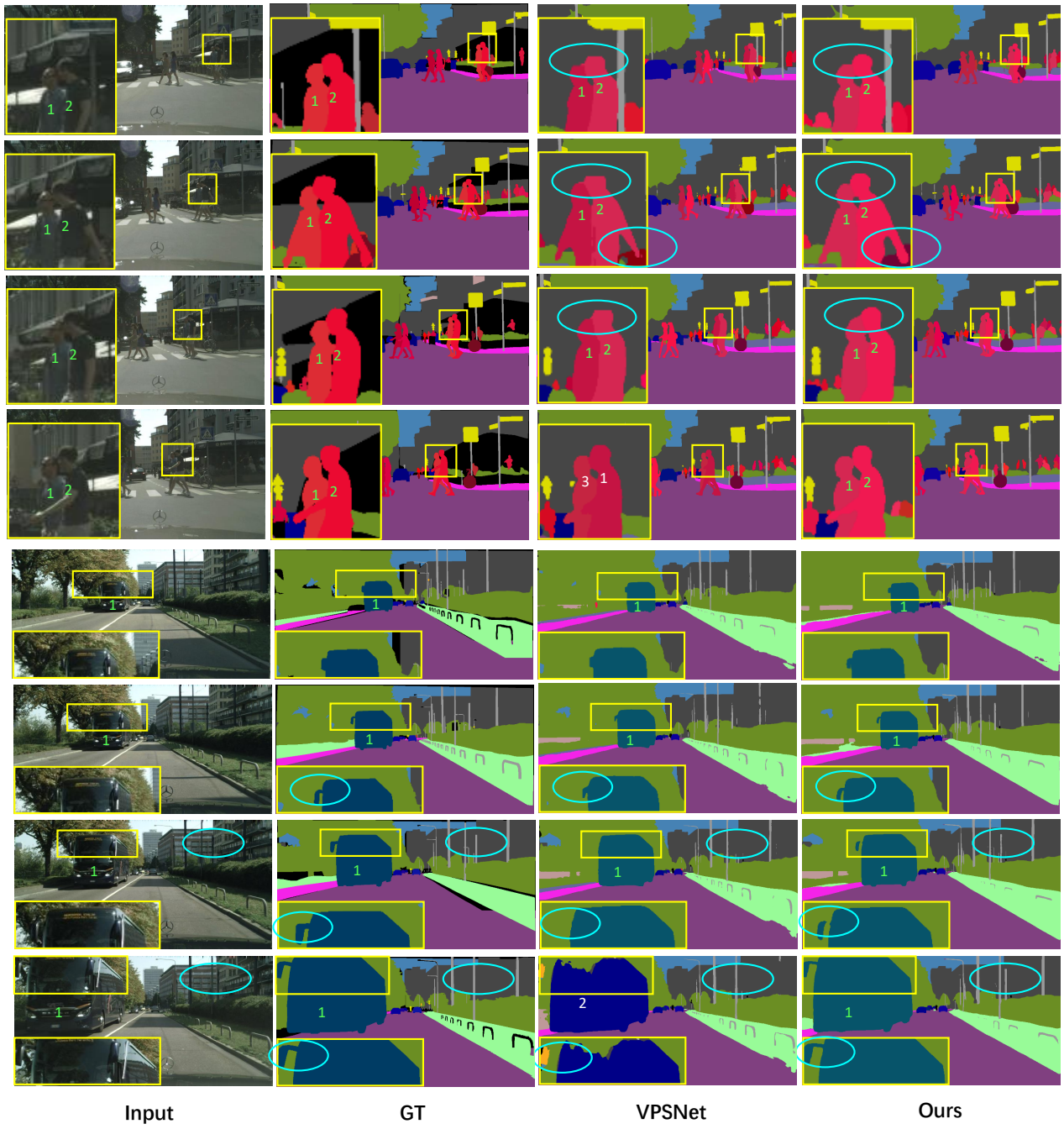

Figure S2. **Visualization of result comparison on Cityscapes-VPS.** From left to right: input frame, GT annotation, VPSNet predictions, and our predictions. Each case contains four consecutive frames (from top to bottom) from a video. Key regions are cropped in yellow rectangles. The matched instances are tagged with the same number across frames. Green and white numbers represent the consistent and inconsistent ID predictions respectively. Note that the bus in the fourth frame of bottom case is mistakenly classified as car by VPSNet. In comparison, our results have better temporal consistency even though people are very close to each other or the bus varies greatly in size. Moreover, as emphasized by the blue ovals, our segmentation masks have richer details (*e.g.* the head and hand of person in top case, the bus mirror in bottom case). Best viewed in color.

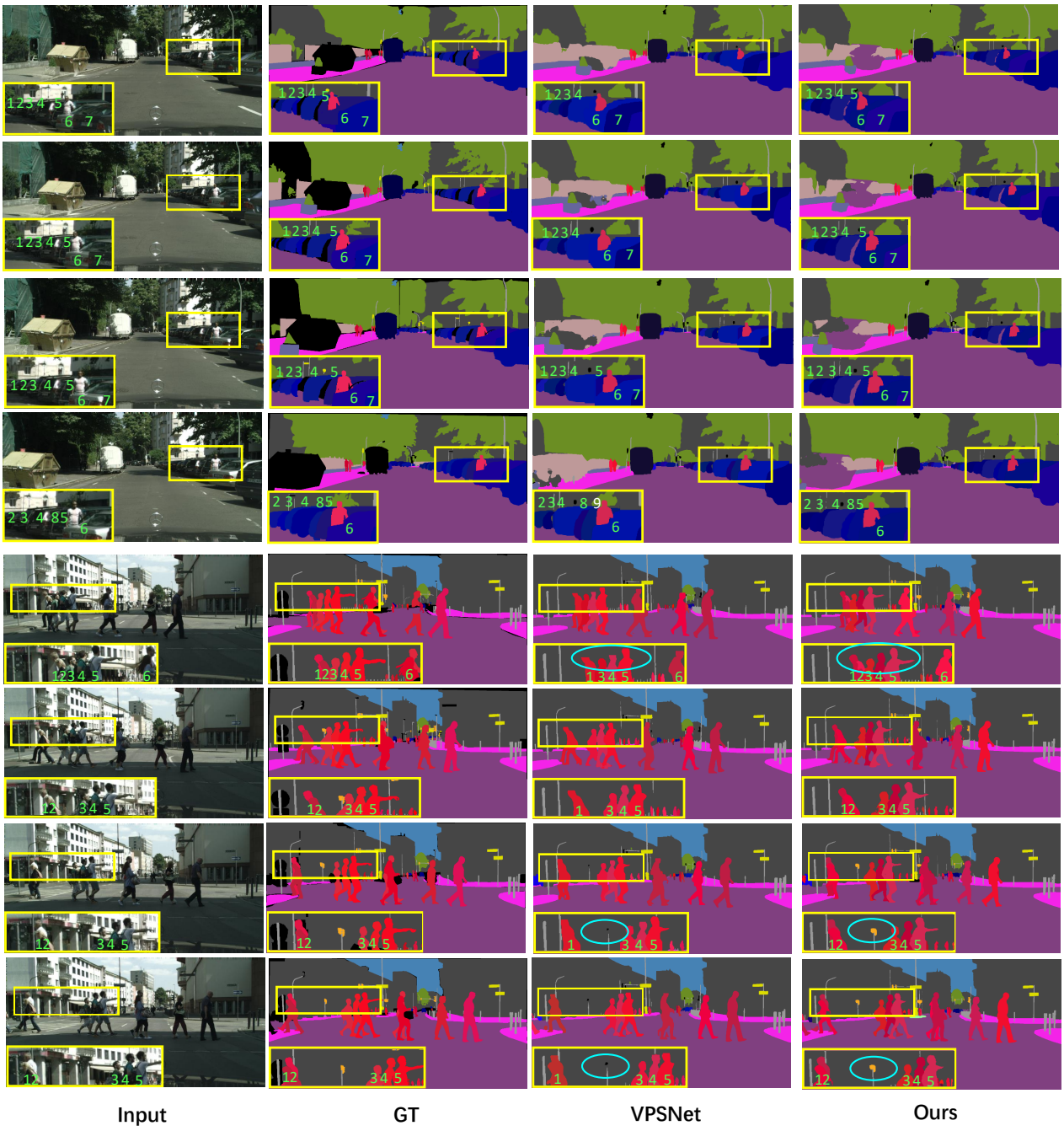

Figure S3. **Visualization of result comparison on Cityscapes-VPS.** From left to right: input frame, GT annotation, VPSNet predictions, and our predictions. Each case contains four consecutive frames (from top to bottom) from a video. Key regions are cropped in yellow rectangles. The matched instances are tagged with the same number across frames. Green and white numbers represent the consistent and inconsistent ID predictions respectively. For this kind of dense situations, VPSNet easily missed or mistakenly assigned IDs to objects (*e.g.* the car with ID 5 in the top case is correctly segmented and identified only in the third frame, the person with ID 2 in the bottom case is missed for all frames.). While our predictions are always consistent across all frames. Moreover, as emphasized by the blue ovals, our segmentation masks have richer details (*e.g.* the head and hand of person, the traffic sign in bottom case). Best viewed in color.

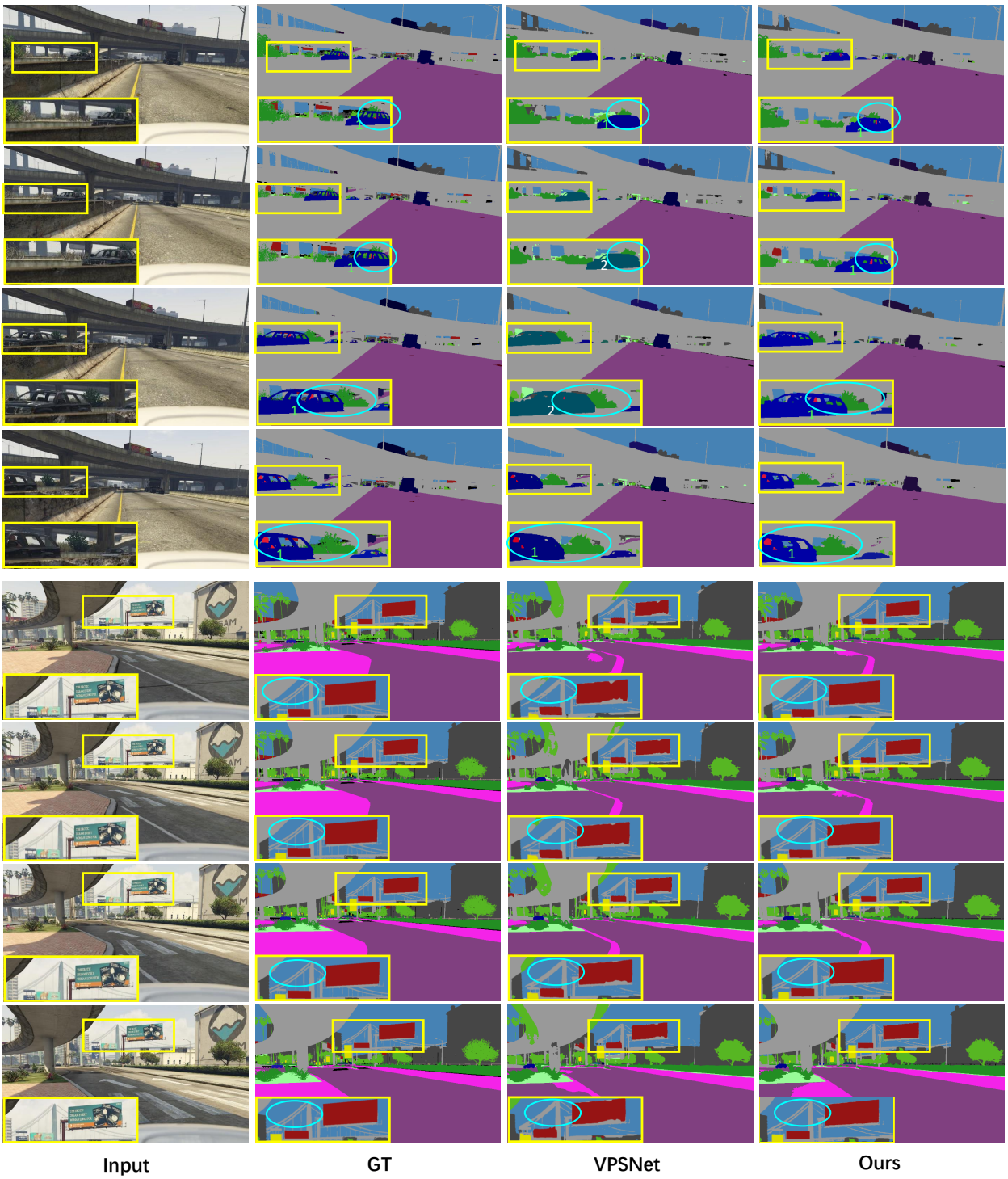

Figure S4. **Visualization of result comparison on VIPER.** From left to right: input frame, GT annotation, VPSNet predictions, and our predictions. Each case contains four consecutive frames (from top to bottom) from a video. Key regions are cropped in yellow rectangles. The matched instances are tagged with the same number across frames. Green and white numbers represent the consistent and inconsistent ID predictions respectively. As emphasized by the blue ovals, our segmentation masks have richer details (*e.g.* the person in the car in the top case, the bridge cable in bottom case). Best viewed in color.

Figure S5. **Visualization of result comparison on VIPER.** From left to right: input frame, GT annotation, VPSNet predictions, and our predictions. Each case contains four consecutive frames (from top to bottom) from a video. Key regions are cropped in yellow rectangles. The matched instances are tagged with the same number across frames. Green and white numbers represent the consistent and inconsistent ID predictions respectively. As emphasized by the blue ovals, our segmentation masks have richer details (*e.g.* the car mirror in the top case, the fence in bottom case). Note that VPSNet can barely segment the details inside cars in these cases but our predictions have sufficient details. Best viewed in color.