Supplementary Material
Self-Supervised Representation Learning from Flow Equivariance

Yuwen Xiong Mengye Ren Wenyuan Zeng Raquel Urtasun
Waabi*, University of Toronto
{yuwen, mren, wenyuan, urtasun}@cs.toronto.edu

1. Additional ablation experiments

1.1. Noisier guidance from flow

To test our framework's generalizability and robustness, here we experiment with a weaker flow model, which produces worse flow prediction. Specifically, we use PWC-Net [2] which is pretrained on Flying-chair, Flying-things, and Sintel. Following ablation experiments in the main paper, we train our model on UrbanCity with 16,000 iterations. The results are shown in Table 1. We can see that using a weaker flow model in our framework only has a minor impact on the model performance.

Flow model	mIoU	mAP	mIoU†	mAP†
PWC-Net	37.2	**3.9**	52.4	16.4
RAFT (main paper)	**37.9**	3.8	**53.2**	**16.5**

Table 1: Semantic segmentation and instance segmentation readout results on UrbanCity with a weaker flow model PWC-Net [2]

1.2. Training BYOL on driving videos:

BYOL [1] shows good performance on ImageNet data which is highly curated and carefully constructed for image-level recognition, and its training protocol is proven to work very well on ImageNet. It treats the entire image as a single instance and random crops two patches on the images and minimizes the two patches' dissimilarity. However, images in the wild may be more complex; in this case, the two random crops may not cover the same objects, leading to potential performance degradation.

For reference, we adapt BYOL on the driving video data. The results are shown in Table 2. For the simplest variant of BYOL, we simply treat video frames like ImageNet images. The random crop augmentation has no spatial constraint; thus, two crops from the same image may cover different objects on the street. We can see that the performance is as bad as a randomly initialized encoder, indicating no meaningful representation is learned in the model.

Method	UrbanCity		BDD100K	
	mIoU†	mAP†	mIoU†	mAP†
BYOL [1]	19.6	5.0	21.9	4.5
BYOL (pre-crop)	26.6	5.1	18.2	4.2
BYOL (pre-crop) w/ video	13.0	2.0	10.7	2.1
FlowE (Ours)	**61.7**	**19.0**	**49.8**	**24.9**

Table 2: BYOL trained on UrbanCity and BDD100K.

We then try to slightly modify it by pre-cropping a 480×480 patch on the original image to limit the movement of the random crop, denoted as "BYOL (pre-crop)". However, the performance is not good either. We also try to change the pre-cropping size, but it does not help. We think one reason might be the higher similarity and less diversity of street scene images compared to ImageNet images: even though the smaller pre-cropped image can reduce the complexity of the image and help the two patches cover the same object, the street scene images are very similar, and patches from different images may share the same semantic meaning (e.g., building, road, sky), making the model hard to distinguish them and learn meaningful representations.

Furthermore, we also try BYOL using two neighboring video frames (BYOL (pre-crop) w/ video), but it is even worse due to extra object movement across frames. These results indicate that the popular BYOL training protocol on ImageNet is not ideal for raw driving videos. Instead, our method can utilize the raw driving video and learn meaningful representations effectively.

2. Additional visualization results

We provide additional visualization results of UrbanCity, BDD100K as well as Cityscapes. Instance segmentation and object detection results are shown in Figure 1, and semantic segmentation results are shown in Figure 2, respectively. For UrbanCity and BDD100K, models are trained on the corresponding datasets. For Cityscapes, the model is

*This work was done by all authors while at Uber ATG

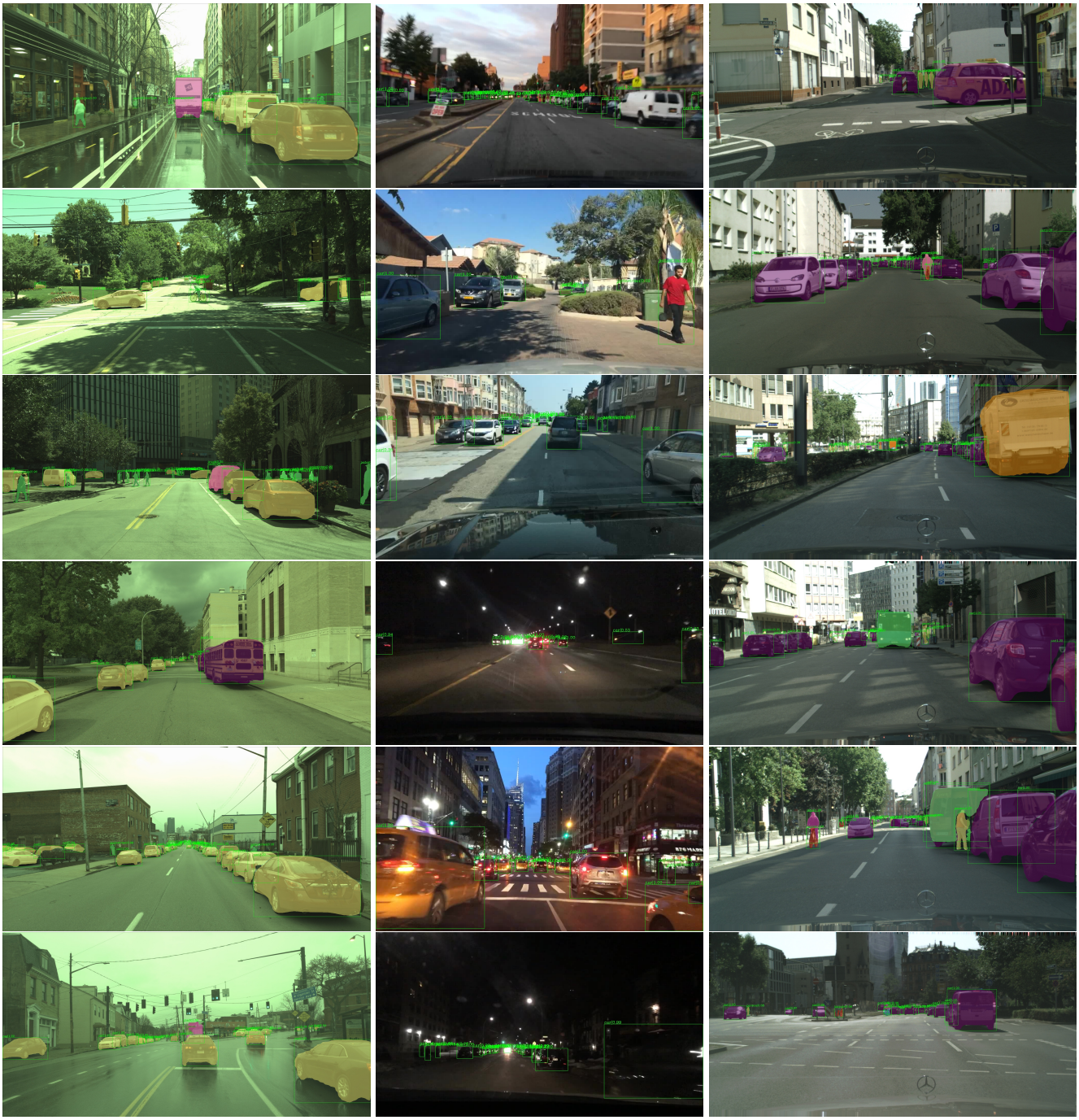

Figure 1: Instance segmentation / object detection visualization results on UrbanCity (left), BDD100K (middle), and Cityscapes (right). We use the heavier header (ResNet-FPN) for readout.

trained on UrbanCity. We use a heavier header to perform instance segmentation/object detection readout tasks and a standard header for semantic segmentation readout task.

We also include a demo video as part of our supplementary material. The file "flowe-demo.mp4" shows semantic segmentation results on the Cityscapes dataset. In this video, we use FlowE trained on UrbanCity, and train a linear readout (1×1 convolution) layer on Cityscapes training set to produce classification logits. We can see that the model can produce impressive results while only have one linear layer learned from the labeled data, indicating that our method can exploit unlabeled driving videos well and learn semantically meaningful representations from them.

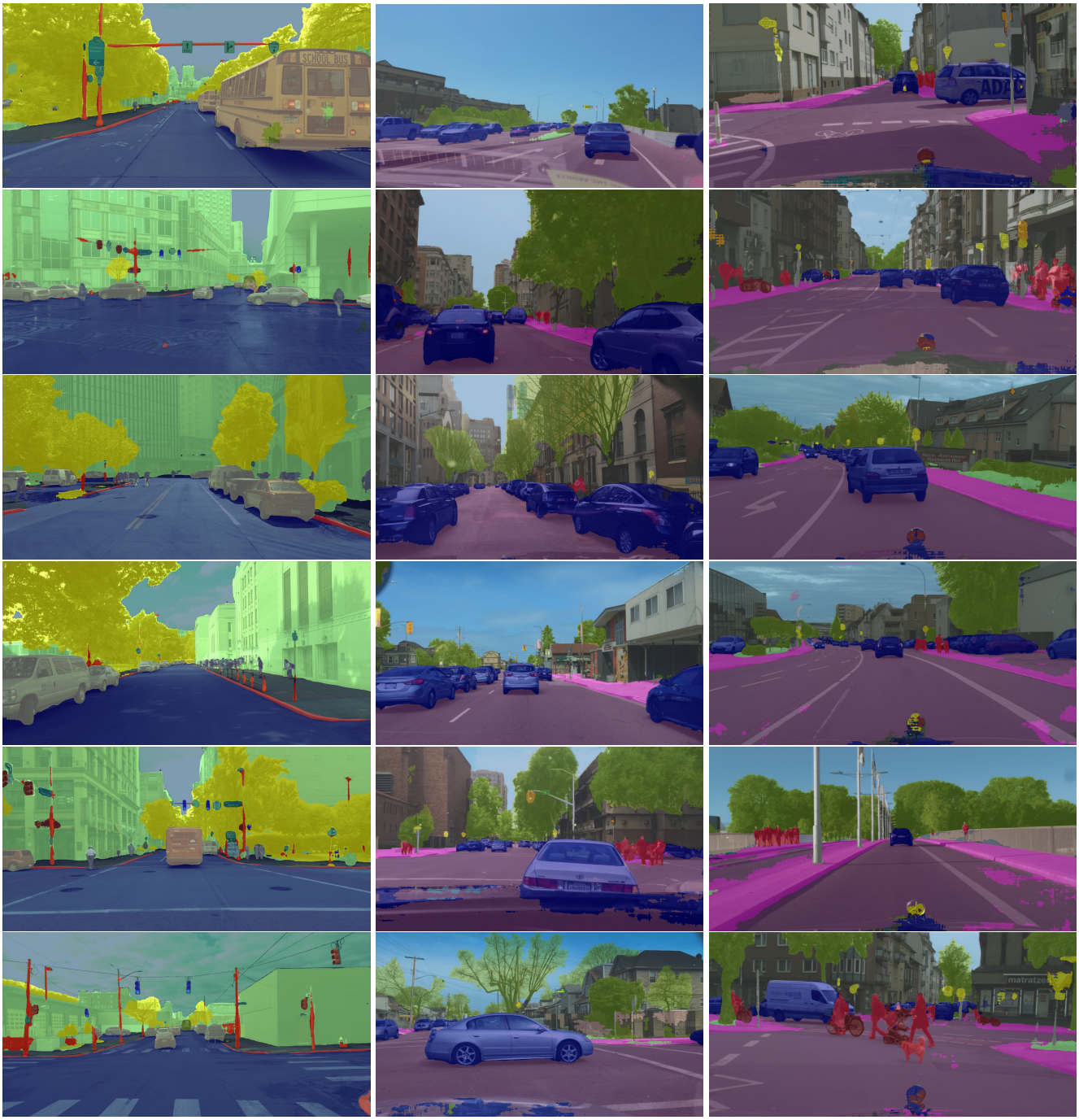

Figure 2: Semantic visualization results on UrbanCity (left), BDD100K (middle), as well as Cityscapes (right). For UrbanCity and BDD100K, models are trained on the corresponding dataset. For Cityscapes, we use a UrbanCity pretrained model. We use the standard header (1 convolutional layer) for readout.

References

[1] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020. 1

[2] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 1