RobustNet: Improving Domain Generalization in Urban-Scene Segmentation via Instance Selective Whitening

A. Supplementary Material

This supplementary section provides additional quantitative results to examine hyper-parameter impacts, further implementation details, and qualitative results.

Comparison of segmentation results is shown in Fig. 1. Our method makes reasonable predictions, while the baseline completely fails on them.

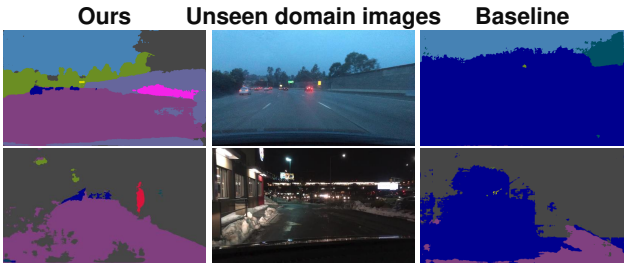

Figure 1. Segmentation results on BDD-100K with the models trained on Cityscapes. The upper image contains dust and water drops on the windshield, and the lower one has an extreme domain shift (*i.e.*, night and snow). Note that Cityscapes does not contain any images taken at night or under a snow condition.

A.1. Comparison with DA methods

We compare the result of our method with those reported from several domain adaptation (DA) methods under various settings. Fig. 2 shows the increase in mIoU from the baseline for each method. Although our method may not be the top performer, it shows comparable results to other DA methods. Note that DA methods require access to the target domain to solve DA problems. In contrast, our method is designed to improve generalization performance on an arbitrary *unseen* domain under the assumption of no access to the target domain, so we believe a comparison with DA methods under the same setting is impossible. However, we expect to solve DA by extending our key idea of *selectively* removing style-sensitive covariances to *selectively* matching such covariances between source and target domain.

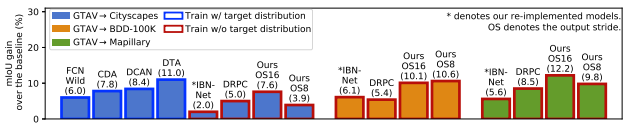

Figure 2. Comparison of mIoU gain(%) from the baseline for each method. Other methods compared to ours are FCN Wild [1], CDA [5], DCAN [4], DTA [2], IBN-Net [3], and DRPC [6].

A.2. Hyper-parameter Impacts

Criteria for separating covariance elements We adopt k-means clustering to separate covariance elements into two groups, domain-specific style and domain-invariant content, according to the variance of each covariance element across various photometric transformations such as color jittering and Gaussian blur. As specified in Section 4.3, after dividing the covariance elements into k clusters by the magnitude of the variance, the clusters from the first to the m-th are considered to be insensitive, and the remaining clusters are considered sensitive to photometric transformation. We set m to one and search the optimal k through the hyper-parameter search. Fig. 3 shows the threshold where the covariances are divided into two groups depending on the k value. Table 1 shows the changes in mIoU performance according to the k values, suggesting the optimal k as 3. Also, we can see that ours (ISW) performs better than IBN-Net or ours (IW) for all k values. Note that ours (IW) applies instance whitening loss to all covariance elements, while ours

Models (GTAV)	C	B	M	S	G
Baseline	28.95	25.14	28.18	26.23	**73.45**
Ours (ISW), k=2	35.46	35.00	39.38	27.70	72.08
Ours (ISW), k=3	**36.58**	**35.20**	**40.33**	**28.30**	72.10
Ours (ISW), k=5	34.84	33.58	39.25	27.52	72.31
Ours (ISW), k=10	33.58	33.76	38.96	27.68	72.24
Ours (ISW), k=20	33.66	33.29	38.70	27.47	72.10
Ours (IW)	33.21	32.67	37.35	27.57	72.06

Table 1. Comparison of mIoU(%) on five different validation sets according to k value. Cityscapes (C), BDD-100K (B), Mapillary (M), SYNTHIA (S), and GTAV (G). The models are trained on GTAV. ResNet-50 is adopted, and an output stride of 16 is used. † denotes re-implemented models. These experiments are conducted three times, and the average results are reported.

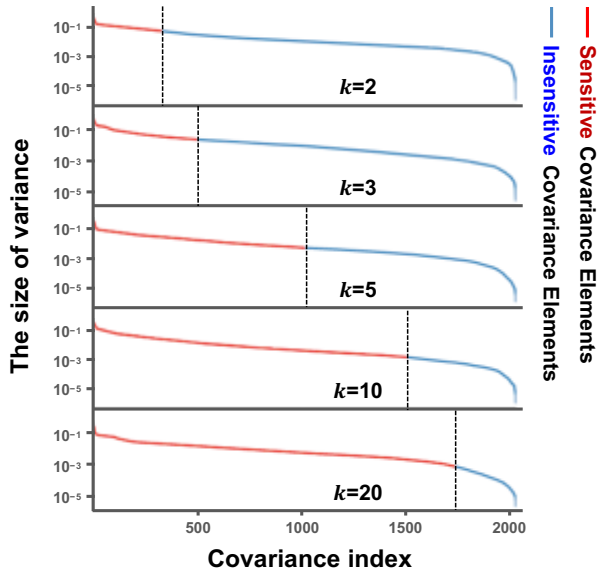

Figure 3. The curves denote the magnitude of the variance of each covariance element across the photometric transformations. The vertical dashed lines represent the threshold to separate the covariance elements. The magnitudes of the variance are extracted from the covariance matrix calculated in the input convolutional layer. The y-axis is in log-scale.

(ISW) applies it to a part of the covariance elements according to the k value.

Margin δ in instance-relaxed whitening (IRW) loss As described in Section 4.2, we propose margin-based relaxation of whitening loss. Table 2 shows the performance of ours (IRW) according to the margin δ.

Weight γ of instance-selective whitening (ISW) loss As described in Section 4.4, we empirically set the weight γ of the proposed ISW loss as 0.6. Table 3 shows the impact of changing γ.

Models (GTAV)	C	B	M	S	G
Baseline	28.95	25.14	28.18	26.23	**73.45**
Ours (IRW), δ=1/16	32.49	32.53	37.51	**27.77**	72.18
Ours (IRW), δ=1/32	33.30	33.17	38.03	27.43	71.96
Ours (IRW), δ=1/64	**33.57**	**33.18**	**38.42**	27.29	71.96
Ours (IRW), δ=1/128	32.85	32.40	37.36	27.43	72.21
Ours (IRW), δ=1/256	32.45	32.32	37.93	27.48	72.12
Ours (IW)	33.21	32.67	37.35	27.57	72.06

Table 2. Comparison of mIoU(%) on five different validation sets according to δ value. The models are trained on GTAV train set. ResNet-50 is adopted and an output stride of 16 is used. These experiments are conducted three times, and the average results are reported.

Models (GTAV)	C	B	M	S	G
Ours (ISW), γ=0.4	35.60	34.07	38.98	28.10	71.96
Ours (ISW), γ=0.6	**36.58**	**35.20**	**40.33**	**28.30**	**72.10**
Ours (ISW), γ=0.8	35.73	34.01	39.69	27.44	71.96

Table 3. Comparison of mIoU(%) on five different validation sets according to γ value. The models are trained on GTAV train set. ResNet-50 is adopted and an output stride of 16 is used. These experiments are conducted three times, and the average results are reported.

A.3. Further Implementation Details

Fig. 4 shows the detailed architecture of the semantic segmentation networks based on ResNet and DeepLabV3+. We adopt the auxiliary per-pixel cross-entropy loss proposed in PSPNet [6] and concatenate the low-level features from the ResNet stage 1 to the high-level features according to the encoder-decoder architecture proposed in DeepLabV3+. Instance normalization (IN) with ISW loss replaces batch normalization (BN) in the input convolutional layer, and these ones are added after the skip-connection of the last residual block for each ResNet stage. As IBN-Net [3] pointed out, earlier layers tend to encode the style information, hence we only adopt the ISW loss to the input convolutional layer and ResNet stage 1 and 2. In the end, the final loss $\mathcal{L}_{\text{Total}}$ is formulated as,

$$\mathcal{L}_{\text{Total}} = \mathcal{L}_{\text{Task (main)}} + \gamma_1 \mathcal{L}_{\text{Task (aux.)}} + \gamma_2 (\frac{1}{3} \sum_{i=1}^{3} \mathcal{L}_{\text{ISW}}^{i}),$$

where the γ_1 is 0.4 and the γ_2 is 0.6. We set the batch size to 8 for Cityscapes and 16 for GTA. For the photometric transformation, we apply Gaussian blur and color jittering implemented in Pytorch with a brightness of 0.8, contrast of 0.8, saturation of 0.8, and hue of 0.3.

A.4. Additional Qualitative Results

This section demonstrates additional qualitative results. We first present the comparison of the segmentation results on a *seen* domain (*i.e.,* Cityscapes) and diverse driving conditions in BDD-100K, and then show the failure cases of our method. Besides, we show the effects of the whitening by comparing the reconstructed images from our proposed approach and the baseline. Finally, we provide the tendency of images from the most sensitive and insensitive covariance elements to the photometric transformation.

Comparison of segmentation results To qualitatively describe the effect of our method, we compare the segmentation results from the baseline and ours. Fig. 6 presents the segmentation results on a *seen* domain (*i.e.,* Cityscapes). Similar to the quantitative results reported in Section 5,

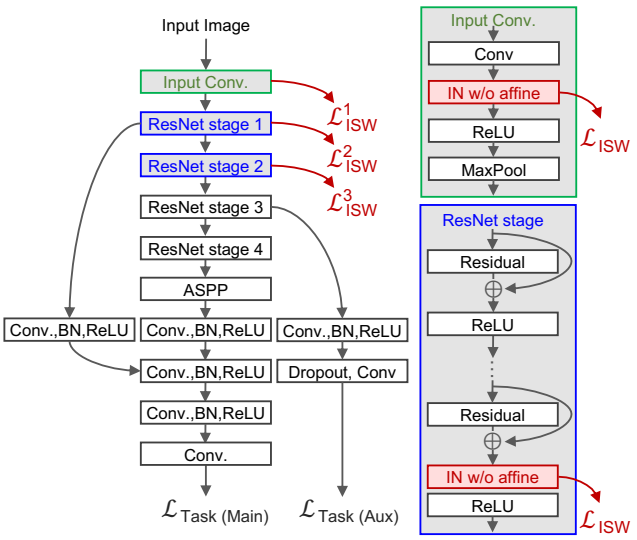

Figure 4. Detailed architecture of the segmentation model.

even with qualitative results, our model shows comparable performance to the baseline model on the *seen* domain. Fig. 7 shows the segmentation results under illumination changes on an *unseen* domain (*i.e.,* BDD-100K). Note that Cityscapes dataset only contains images taken at the daytime. The first group images are taken at the dusk. We can see that the baseline model is vulnerable to these changes, but in contrast, our model outputs less damaged maps and reasonably predicts roads and cars. In extreme cases such as at night, both models fail to predict the sky, but our method still finds key components such as roads and cars well. In addition, our method produces reasonable segmentation results even for drastic changes in lighting such as shadows, as seen in the third group. Fig. 8 shows the segmentation results under the adverse weather conditions, unseen structures, and lush vegetation. Our model successfully predicts a partially snowy sidewalk, whereas the baseline model incorrectly predicts it as a building. The second case in the first group shows a foggy urban scene. The baseline fails to cope with these weather changes, while ours still shows fair results. Under the structural changes as shown in the second group, our method finds the road and sidewalk better than the baseline. Moreover, the baseline totally fails to detect the parking lot. In the last case, which is lush vegetation, the baseline produces noisy segmentation results and confused the road as a car. On the other hand, our model shows reasonable performance in both cases. Fig. 5 shows the failure cases caused by a large domain shift.

Covariance effects in images To reveal the information that the covariance represents, we first identify the most sensitive and insensitive covariances to the photometric transformation. Then, we sort the BDD-100K images according to the magnitude of the identified covariances. The results are described in Fig. 9. In the left group, the images are getting dark as the most sensitive covariance is getting smaller. We conjecture that the corresponding covariance tends to represent the *illumination* information. On the other hand, the right group shows the sorted images along with the most insensitive covariance. The scenes are getting simpler as the covariance gets smaller, which implies that the most insensitive covariance tends to represent the *scene complexity*.

References

[1] Judy Hoffman, Dequan Wang, Fisher Yu, and Trevor Darrell. Fcns in the wild: Pixel-level adversarial and constraint-based adaptation. *arXiv preprint arXiv:1612.02649*, 2016. 1

[2] Seungmin Lee, Dongwan Kim, Namil Kim, and Seong-Gyun Jeong. Drop to adapt: Learning discriminative features for unsupervised domain adaptation. In *International Conference on Computer Vision (ICCV)*, 2019. 1

[3] Xingang Pan, Ping Luo, Jianping Shi, and Xiaoou Tang. Two at once: Enhancing learning and generalization capacities via ibn-net. In *European Conference on Computer Vision (ECCV)*, 2018. 1, 2

[4] Zuxuan Wu, Xintong Han, Yen-Liang Lin, Mustafa Gokhan Uzunbas, Tom Goldstein, Ser Nam Lim, and Larry S Davis. Dcan: Dual channel-wise alignment networks for unsupervised scene adaptation. In *European Conference on Computer Vision (ECCV)*, 2018. 1

[5] Yang Zhang, Philip David, and Boqing Gong. Curriculum domain adaptation for semantic segmentation of urban scenes. In *International Conference on Computer Vision (ICCV)*, 2017. 1

[6] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 2

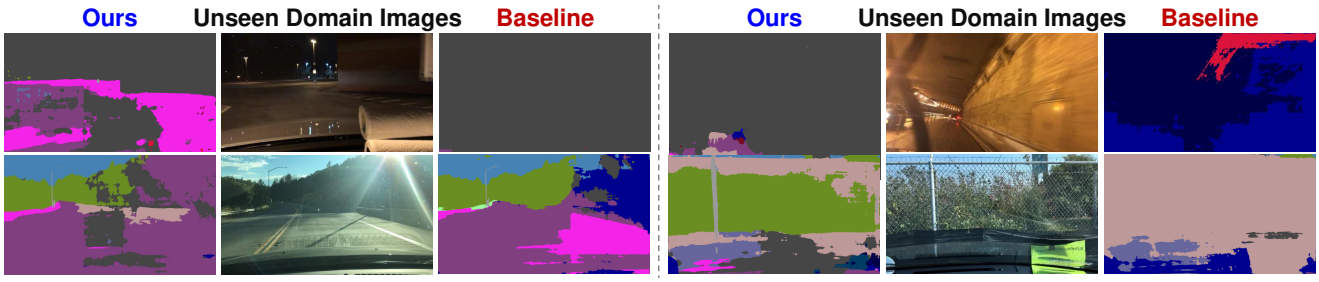

Figure 5. Comparison of failure cases of our method and the baseline.

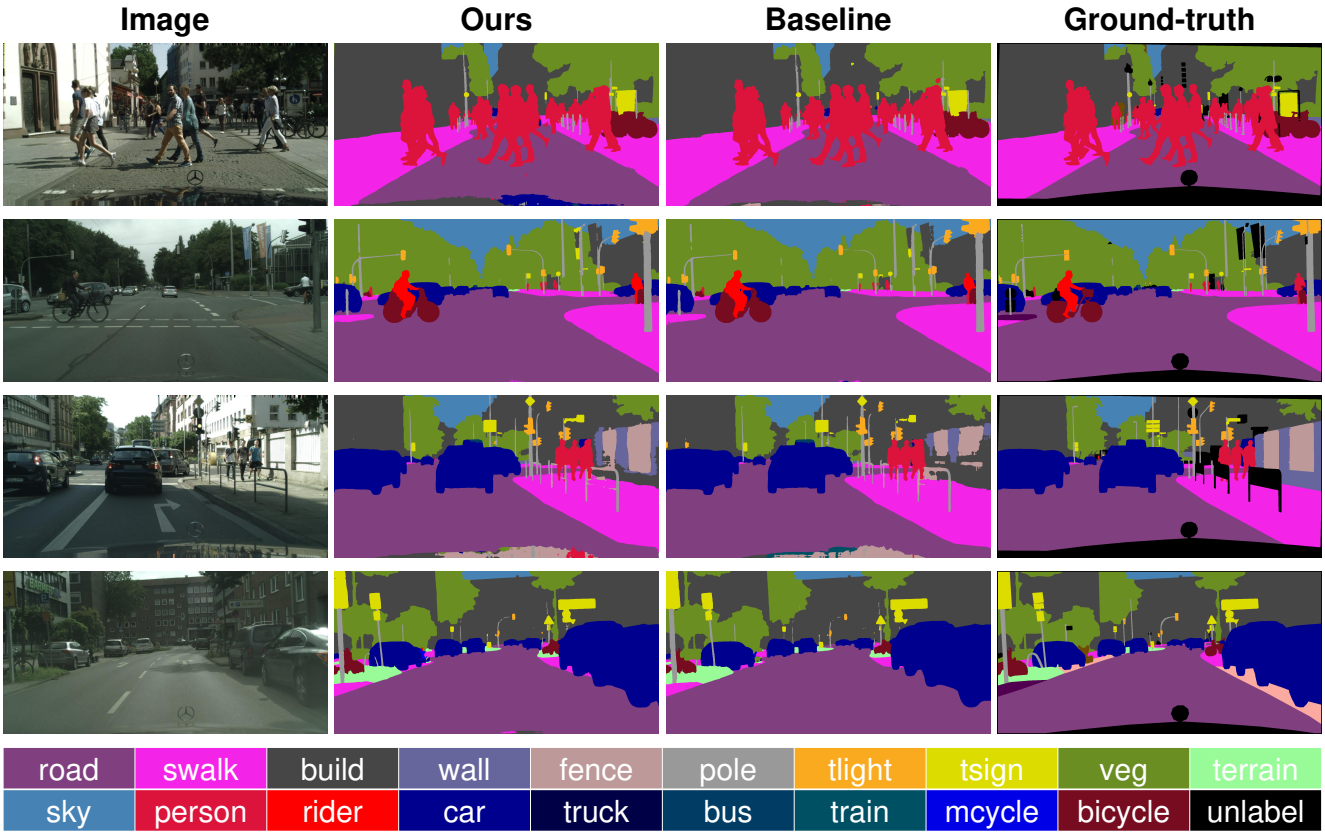

Figure 6. Segmentation results on *seen* domain images (*i.e.,* Cityscapes).

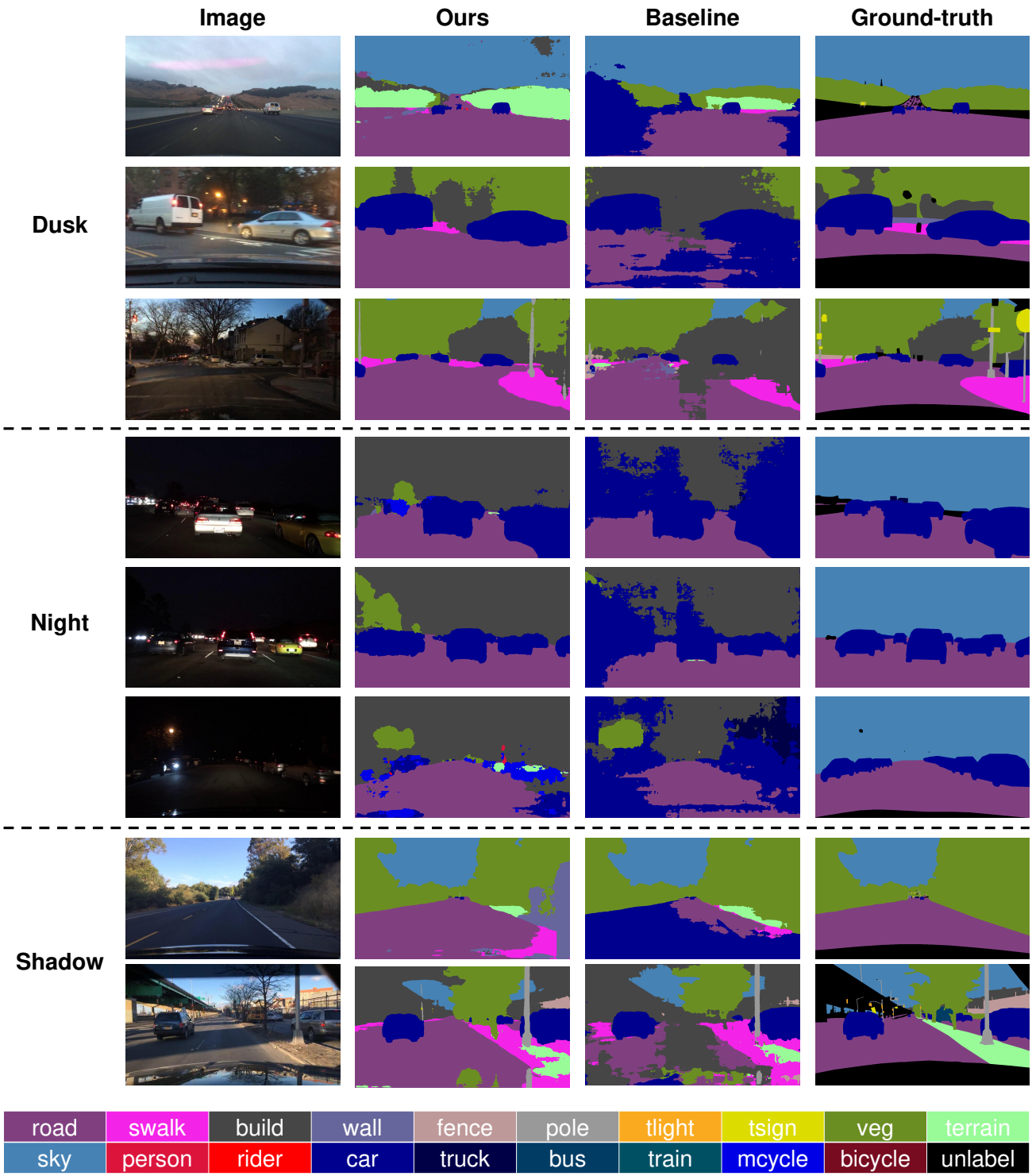

Figure 7. Segmentation results under illumination changes (*i.e.,* dusk, night, and shadow) in BDD-100K with the models trained on Cityscapes.

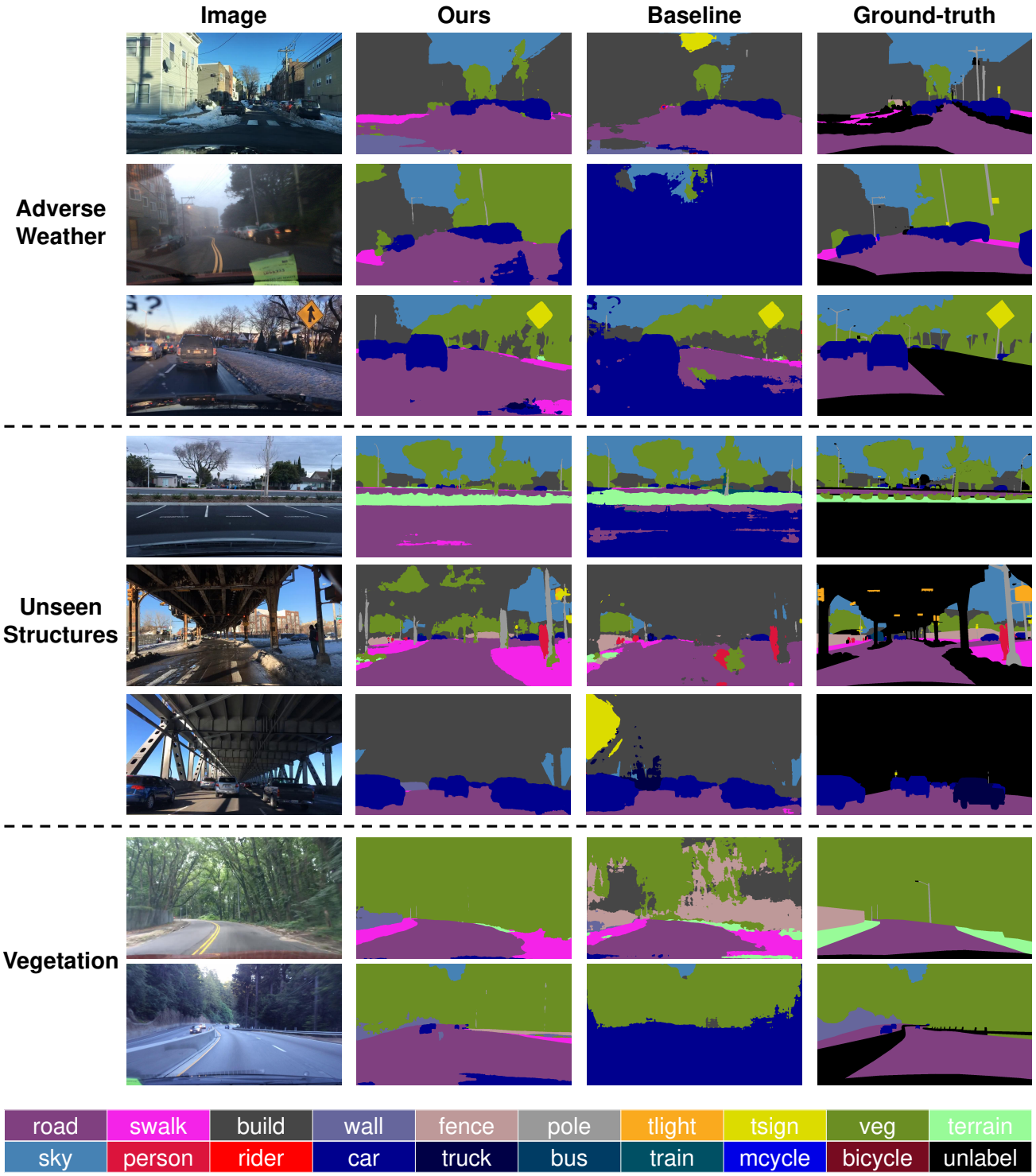

Figure 8. Segmentation results under various circumstances in BDD-100K with the models trained on Cityscapes. Circumstances include adverse weather conditions (*i.e.,* snow and fog), unseen structures (*i.e.,* parking lot and overpass), and vegetation.

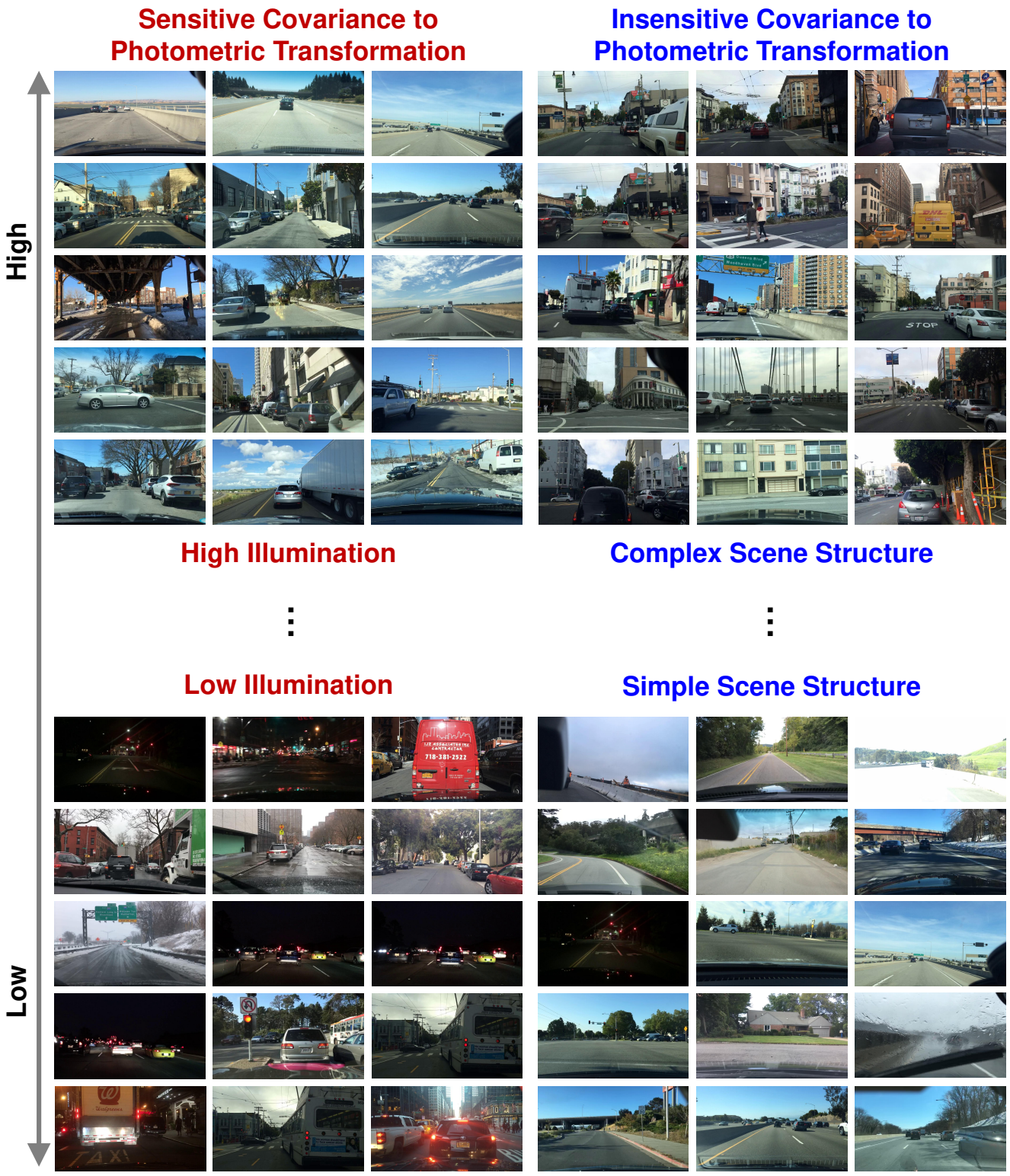

Figure 9. Tendency of images in BDD-100K dataset along with the covariance changes.