CAR: Class-aware Regularization for Semantic Segmentation Supplementary

Ye Huang[1], Di Kang[2], Liang Chen[3], Xuefei Zhe[2], Wenjing Jia[1], Linchao Bao[2], and Xiangjian He[4⋆]

[1] University of Technology Sydney, Australia
[2] Tencent AI Lab
[3] Fujian Normal University
[4] University of Nottingham Ningbo China

A Appendix

A.1 Extra technical details

A.1.1 Deterministic Control variables are very important for all scientific research. In computer vision, we always use the same backbones and the same datasets when verifying the difference between two methods.

Without using "deterministic" technology, all operations in neural networks contain some randomness. Nowadays, with the latest deterministic technology and fixed seeds, experiments can be conducted in a fully-controlled environment. This means that the performance difference between different settings (*i.e.*, w/ and w/o CAR) is not affected by this randomness any more but faithfully reflects the effectiveness of different methods.

In Tab. 1, we report the performance of our proposed CAR (ResNet-50 + Self-attention and Swin-Tiny + UperNet) with different seeds for readers who are interested in how our CAR performs when trained with different random seeds. As it is shown, our CAR consistently improves the mIOU over its baseline using different random seeds, demonstrating the effectiveness of the CAR.

A.2 Extra experiments

A.2.1 Ablation studies on batch class center In our experiments, we calculated the class centers using all the training images in a *batch* to alleviate the negative impact of noisy images. Here, we investigate the impact of using the class center of each individual image for class-aware regularizations.

A.2.2 CAR using Moving Average. We also implemented a moving average version of CAR which tracks the class center $\boldsymbol{\mu}$ with moving average similar to BatchNorm. As shown in Tab. 3, we find this moving average version of CAR negatively impacts both ResNet-50 + Self-Attention and Swin-Tiny + Uper.

⋆ Corresponding author

Table 1: Ablation studies of our proposed CAR using different random seeds on the Pascal Context dataset.

Methods	Seed (mIOU%)		
	0 (default)	1	2
ResNet-50 + Self-Attention	48.32	47.54	47.69
ResNet-50 + Self-Attention + CAR	50.50(+2.18)	50.20(+2.66)	50.59(+2.90)
Swin-Tiny + UperNet	49.62	49.24	49.54
Swin-Tiny + UperNet + CAR	50.78(+1.16)	50.57(+1.33)	50.75(+1.21)

Table 2: Comparison of mIOUs (%) obtained when using batch class center vs image class center in CAR.

Methods	Baseline	CAR	
		Image Class Center	Batch Class Center
ResNet-50 + Self-Attention	48.32	49.78	50.50
Swin-Tiny + UperNet	49.62	49.45	50.78

A.2.3 CAR without intra-class loss. Table 4 below shows that, including only inter-c2c loss and inter-c2p still improves the result.

A.2.4 Exceeding state-of-the-art (SOTA) in Pascal Context The main motivation of our CAR is to utilize class-level information as regularizations during training to boost the performance of all existing methods. However, following the convention and also for readers who are interested, we compare with state-of-the-art methods in Tab. 5 regardless their architectures are related to ours or not. Since Swin [5] is not compatible with dilation, we use JPU [8] as the substitution to obtain features with output stride = 8. Uper contains an FPN [4] module that can obtain features with output stride = 4.

Boosted by our CAR, the strong model ConvNext-Large [6] + CAA [3] achieved the performance of 62.70% mIOU under single-scale testing, and 63.91% under multi-scale testing. Also, we found increasing the training iterations from the default 30K to 40K when using Adam optimizer can further increase performance in Pascal Context dataset. Thus, the SOTA single model performance has now been boosted to 62.97% under single-scale testing, and 64.12% under multi-scale testing. This has outperformed the previous SOTA single model, *i.e.*, EfficientNetB7 + CAA, by a large margin.

A.2.5 Exceeding SOTA performance in COCOStuff-10K Simliar to Sec. A.2.4, in Tab. 6, boosted by our CAR, the strong model ConvNext-Large [6] + CAA achieved the performance of 49.03% mIOU under single-scale testing, and 50.01% under multi-scale testing. This has also outperformed the previous SOTA single model, *i.e.*, EfficientNetB7 + CAA, by a large margin.

Table 3: Ablation studies of adding moving average to CAR on Pascal Context. Decay rate stands for the effect of old class center.

Methods	CAR	CAR (Moving Average)		
		0.8	0.9	0.99
ResNet-50 + Self-Attention	50.50	49.80(−0.70)	50.26(−0.24)	49.96(−0.54)
Swin-Tiny + UperNet	50.78	49.56(−1.22)	50.03(−0.75)	48.93(−1.85)

Table 4: Extra ablation studies for CAR without intra-class loss.

	$\mathcal{L}_{\text{intra-c2p}}$	$\mathcal{L}_{\text{inter-c2c}}$	$\mathcal{L}_{\text{inter-c2p}}$	A	mIOU
ResNet-50 + Self-Attention	-	-			48.32
				✓	48.56
+ CAR		✓	✓		49.31
		✓	✓	✓	50.23

A.3 Extra Visualizations

A.3.1 Visualization of OCRNet in Pascal Context Similar to the main paper, in Fig. 1, we visualize the pixel-to-pixel relation energy maps obtained with HRNetW48 [9] + OCR [9]. This figure shows that our CAR can further improve the robustness of class center based models by making better use of the class center. Interestingly, as shown in C12 of Fig. 1 and Fig. 1 shown in our main paper what is predicted by ResNet-50 + Self-Attention, we find cow/sheep/dog misclassification is a common issue in many semantic segmentation models, especially when *i.e.* grass and cow co-exist frequently during training. This issue is better addressed by our CAR due to its reduced inter-class dependency.

A.3.2 Visualization of DeepLab in Pascal Context We also visualize the pixel-to-pixel relation energy map of ResNet-50 [2] + DeepLabV3 [1] in Fig. 2. These visualizations clearly show that the reduced inter-class dependency helps to correct the classification.

Table 5: Experiments on boosting the SOTA single-model performance on Pascal Context by our CAR. See Sec. A.2.4 for the details. *§*: We report previous SOTA score as reference. *SS*: mIOU on Single scale without flipping. *MF*: Multi-scale with flipping. JPU is used to get features with output stride = 8. *Aux*: Apply auxiliary loss during training, see [10]. *Iters*: training iterations. *WP*:Linear learning rate warmup

Methods	Backbone	Aux	Optimizer	Iters	WP	SS(%)	MF(%)
CAA§	EfficientNet-B7-D8	✓	SGD	30K		-	60.30
UperNet	Swin-Large		SGD	30K		57.48	59.45
UperNet + CAR	Swin-Large		SGD	30K		58.97	60.76
CAA	Swin-Large + JPU		SGD	30K		58.31	59.75
CAA + CAR	Swin-Large + JPU		SGD	30K		59.84	61.46
CAA + CAR	Swin-Large + JPU		Adam	30K		60.68	62.21
CAA	ConvNeXt-Large + JPU		SGD	30K		60.48	61.80
CAA + CAR	ConvNeXt-Large + JPU		SGD	30K		61.40	62.69
CAA + CAR	ConvNeXt-Large + JPU		Adam	30K		62.65	63.77
CAA + CAR	ConvNeXt-Large + JPU	✓	Adam	30K		62.70	63.91
CAA + CAR	ConvNeXt-Large + JPU	✓	Adam	40K		62.97	**6**4.12
CAA + CAR	ConvNeXt-Large + JPU	✓	Adam	40K	✓	63.13	64.17

Table 6: Experiments on boosting SOTA on COCOStuff10k, levering the previous single model SOTA and boosted by our CAR. See Sec. A.2.5 for details. *§*: We report the original SOTA scores. *SS*: Single scale without flipping. *MF*: Multi-scale with flipping. *Aux* Apply auxiliary loss during training, see [10].

Methods	Backbone	Aux	Optimizer	SS mIOU(%)	MF mIOU(%)
CAA§	EfficientNet-B7-D8	✓	SGD	-	45.40
UperNet	Swin-Large		SGD	44.25	46.10
UperNet + CAR	Swin-Large		SGD	44.88	46.64
CAA	Swin-Large + JPU		SGD	44.22	45.31
CAA + CAR	Swin-Large + JPU		SGD	45.48	46.99
CAA	ConvNeXt-Large + JPU		SGD	46.49	47.23
CAA + CAR	ConvNeXt-Large + JPU		SGD	46.70	47.77
CAA + CAR	ConvNeXt-Large + JPU		Adam	48.20	48.83
CAA + CAR	ConvNeXt-Large + JPU	✓	Adam	**49.03**	**50.01**

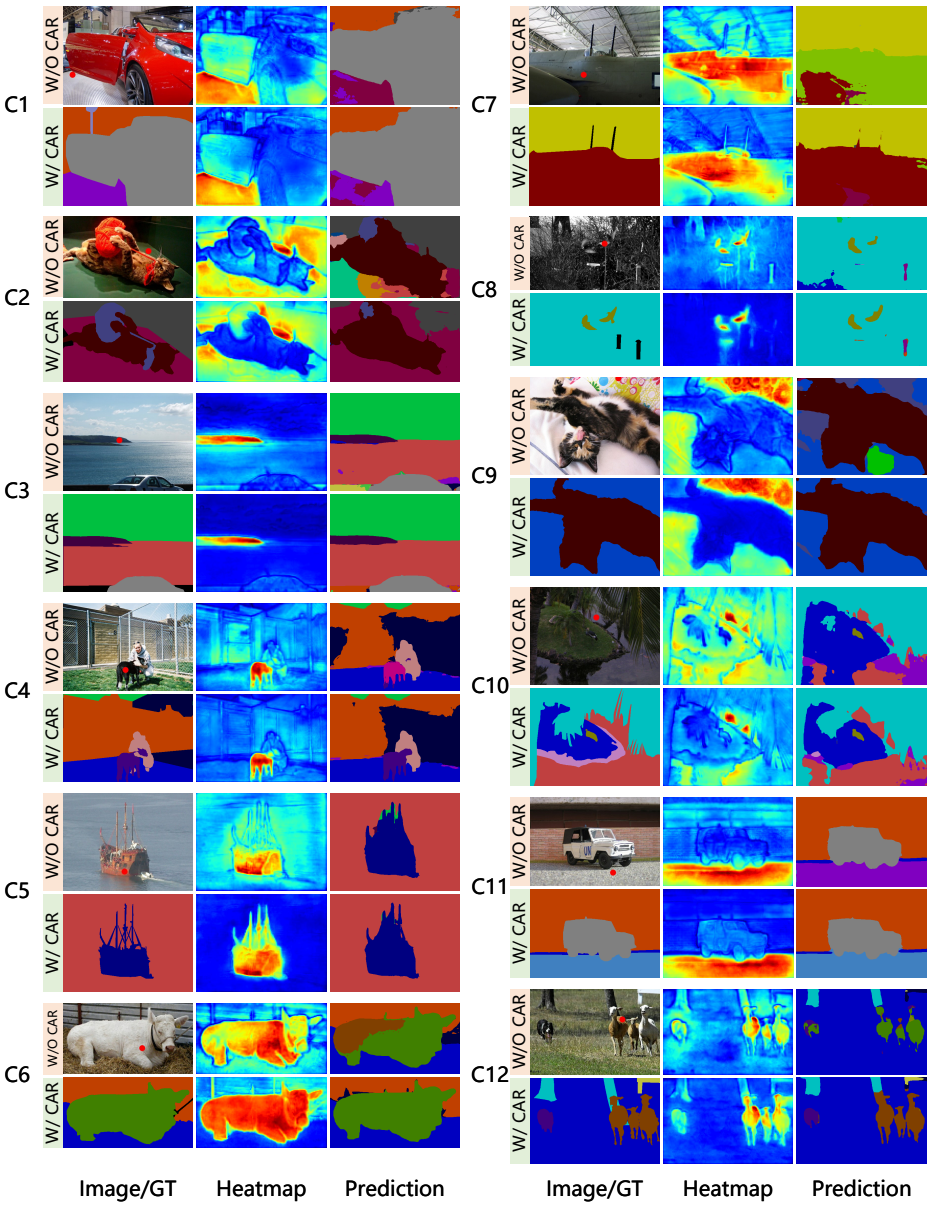

Fig. 1: Visualization of the feature similarity between a given pixel (marked with a red dot in the image) and all other pixels, as well as the segmentation results of **HRNetW48** [7] + **OCR** [9] on Pascal Context test set. A hotter color denotes a greater similarity value.

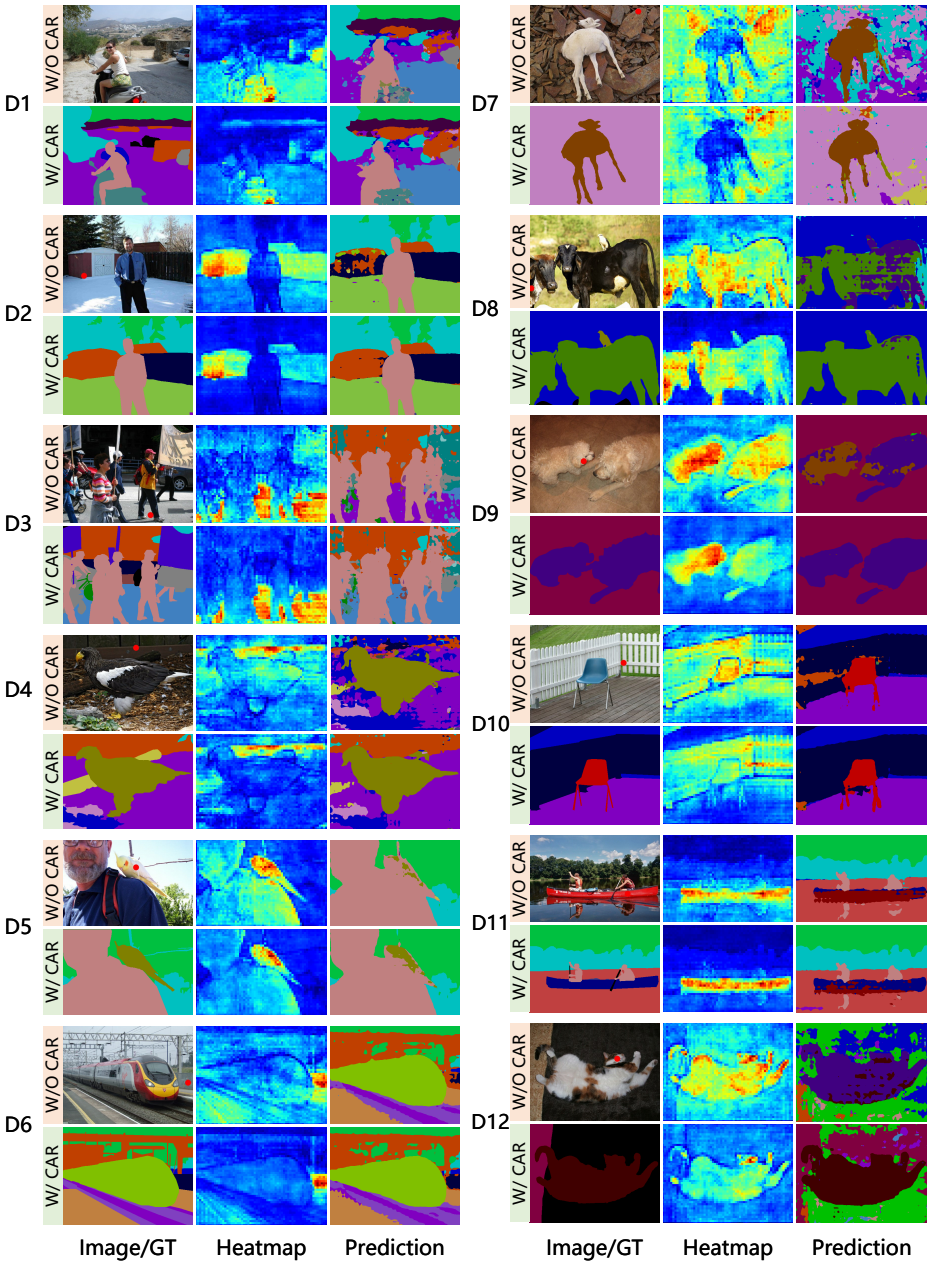

Fig. 2: Visualization of the feature similarity between a given pixel (marked with a red dot in the image) and all pixels, as well as the segmentation results of **ResNet-50 [2] + DeepLab [1]** on Pascal Context test set. A hotter color denotes a greater similarity value.

References

1. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation (2017)
2. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (2016)
3. Huang, Y., Kang, D., Jia, W., He, X., liu, L.: Channelized axial attention - considering channel relation within spatial attention for semantic segmentation. In: AAAI (2022)
4. Lin, T.Y., Dollá, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (2017)
5. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: ICCV (2021)
6. Liu, Z., Mao, H., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie, S.: A convnet for the 2020s. In: IEEE Conference on Computer Vision and Pattern Recognition (2022)
7. Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., Liu, W., Xiao, B.: Deep high-resolution representation learning for visual recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (2020)
8. Wu, H., Zhang, J., Huang, K., Liang, K., Yizhou, Y.: Fastfcn: Rethinking dilated convolution in the backbone for semantic segmentation (2019)
9. Yuan, Y., Chen, X., Wang, J.: Object-contextual representations for semantic segmentation. In: European Conference on Computer Vision (2020)
10. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: IEEE Conference on Computer Vision and Pattern Recognition (2017)