FairScene: Learning Class-Disentangled 2D/3D Representations for Semantic Scene Completion Supplementary Material

Dian Jia[1] Pei Yu[2] Wei Tang[1]
[1]University of Illinois Chicago, Chicago, IL, USA
[2]Microsoft, Redmond, WA, USA
{djia7,tangw}@uic.edu, pei.yu@microsoft.com

1. Losses

We adopt the scene-class affinity loss ℓ^{SCAL} from MonoScene [2]. It optimizes the class-wise differentiable (P)recision, (R)ecall and (S)pecificity, where t^{P} and t^{R} measure the performance of similar voxels, and t^{S} measures the performance of dissimilar voxels. Let $\boldsymbol{p}_i$ and $\hat{\boldsymbol{p}}_i$ be the ground truth label and the predicted probability of voxel i, respectively. $\mathcal{V}$ is the set of all voxels. C denotes the number of classes. Let $p_{i,j}$ be the j-th element of $\boldsymbol{p}_i$. We have the following definitions:

$$t^{\text{P}} = \frac{\sum_{i\in\mathcal{V}} \boldsymbol{p}_i \cdot \hat{\boldsymbol{p}}_i}{\sum_{i\in\mathcal{V}} \sum_{j=1}^{C} \hat{p}_{i,j}}, \tag{1}$$

$$t^{\text{R}} = \frac{\sum_{i\in\mathcal{V}} \boldsymbol{p}_i \cdot \hat{\boldsymbol{p}}_i}{\sum_{i\in\mathcal{V}} \sum_{j=1}^{C} p_{i,j}}, \tag{2}$$

$$t^{\text{S}} = \frac{\sum_{i\in\mathcal{V}} (1-\boldsymbol{p}_i) \cdot (1-\hat{\boldsymbol{p}}_i)}{\sum_{i\in\mathcal{V}} \sum_{j=1}^{C} (1-p_{i,j})}, \tag{3}$$

The scene-class affinity loss is defined as:

$$\begin{aligned}\ell^{\text{SCAL}} = \ell^{\text{BCE}}(t^{\text{P}}, \hat{t}^{\text{P}}) + \ell^{\text{BCE}}(t^{\text{R}}, \hat{t}^{\text{R}}) \\ + \ell^{\text{BCE}}(t^{\text{S}}, \hat{t}^{\text{S}})\end{aligned} \tag{4}$$

where ℓ^{BCE} denotes the binary cross-entropy loss, and $\hat{t}^{\text{P}}$, $\hat{t}^{\text{R}}$, and $\hat{t}^{\text{S}}$ are respectively target precision, recall, and specificity calculated by replacing the prediction in Eqs. (1)-(3) with ground truth. Following MonoScene [2], the semantic scene completion loss ℓ^{SSC} includes the scene-class affinity loss on both the semantic label and geometric label, along with a standard cross entropy loss:

$$\ell^{\text{SSC}} = \ell^{\text{SCAL}}_{\text{SEM}} + \ell^{\text{SCAL}}_{\text{GEO}} + \ell^{\text{CE}} \tag{5}$$

For 3D semantic guidance, we apply standard cross-entropy loss and lovasz loss [1]:

$$\ell^{\text{3D}} = \ell^{\text{CE}} + \ell^{\text{LOVASZ}} \tag{6}$$

$$\ell^{\text{LOVASZ}} = \frac{1}{C} \sum_{c=1}^{C} \frac{1}{|\mathcal{P}_c|} \sum_{k=1}^{|\mathcal{P}_c|} \Delta J_c(k)\, \ell_c(k) \tag{7}$$

where $|\mathcal{P}_c|$ denotes the total number of voxels considered for category c, $\ell_c(k)$ is the k-th largest error (e.g., hinge losses) for category c, sorted in descending order, $\Delta J_c(k)$ represents the change in the Jaccard index (IoU) for category c when including the k-th error, and C is the number of classes.

For a given class c, the Lovász loss is computed as follows. First, we calculate the error for each voxel i belonging to class c. These errors are then sorted in descending order. Next, we compute the Jaccard (IoU) increment $\Delta J_c(k)$ for the k-th voxel, which quantifies the change in the Jaccard index when including this voxel's error. Finally, the Lovász loss for class c is defined as a weighted sum of the sorted errors.

For 2D semantic guidance ℓ^{2D}, we only apply standard pixel-wise cross-entropy loss. The final loss functions for training is:

$$\ell = \ell^{\text{SSC}} + \ell^{\text{3D}} + \ell^{\text{2D}}. \tag{8}$$

2. Implementation Details

Algorithm 1 provides more details about OccMix. In summary, it consists of three key components: (1) View-Occupancy Mixing; (2) Voxel Proposal Fusion; and (3) Mixed Feature Lifting. It is worth noting that, in the 2D-to-3D feature lifting process, instead of extracting 2D features from two training samples separately through the backbone network, we lift the mixed 2D features $\tilde{\boldsymbol{F}}^{2D}$ twice using different camera matrices. This design choice avoids running the backbone network twice on different images, thereby reducing computational overhead.

3. Additional Results

Method	Input	Supervision	IoU	mIoU	road (15.30%)	sidewalk (11.13%)	parking (1.12%)	other-grnd. (0.56%)	building (14.1%)	car (3.92%)	truck (0.16%)	bicycle (0.03%)	motorcycle(0.03%)	other-veh. (0.20%)	vegetation (39.3%)	trunk (0.51%)	terrain (9.17%)	person (0.07%)	bicyclist (0.07%)	motorcyclist (0.05%)	fence (3.90%)	pole (0.29%)	traf.-sign (0.08%)
CGFormer [8]	Stereo	Voxel + Lidar	45.99	16.87	65.51	32.31	20.82	0.16	23.52	34.32	19.44	4.61	2.71	7.67	26.93	8.83	39.54	2.38	4.08	0.00	9.20	10.67	7.84
MonoScene [2]	Mono	Voxel	37.12	11.50	57.47	27.05	15.72	0.87	14.24	23.55	7.83	0.20	0.77	3.59	18.12	2.57	30.76	1.79	1.03	0.00	6.39	4.11	2.48
TPVFormer [3]	Mono	Voxel	35.61	11.36	56.50	25.87	**20.60**	0.85	13.88	23.81	8.08	0.36	0.05	4.35	16.92	2.26	30.38	0.51	0.89	0.00	5.94	3.14	1.52
OccFormer [9]	Mono	Voxel	36.50	13.46	58.85	26.88	19.61	0.31	14.40	25.09	25.53	0.81	1.19	8.52	19.63	3.93	32.62	2.78	2.82	0.00	5.61	4.26	2.86
VoxFormer [5]	Stereo	Voxel	44.02	12.35	54.76	26.35	15.50	0.70	17.65	25.79	5.63	0.59	0.51	3.77	24.39	5.08	29.96	1.78	3.32	0.00	7.64	7.11	4.18
Symphonize [4]	Stereo	Voxel	41.92	14.89	56.37	27.58	15.28	0.95	**21.64**	28.68	20.44	2.54	2.82	**13.89**	25.72	6.60	30.87	3.52	2.24	0.00	8.40	9.57	5.76
HASSC [7]	Stereo	Voxel	**44.82**	13.48	57.05	28.25	15.90	**1.04**	19.05	27.23	9.91	0.92	0.86	5.61	25.48	6.15	32.94	2.80	**4.71**	0.00	6.58	7.68	4.05
SGN [6]	Stereo	Voxel	43.60	14.55	**59.32**	**30.51**	18.46	0.42	21.43	**31.88**	13.18	0.58	0.17	5.68	25.98	7.43	**34.42**	1.28	1.49	0.00	9.66	9.83	4.71
FairScene (ours)	Stereo	Voxel	43.63	**16.06**	57.21	30.02	18.64	0.45	21.63	31.42	**26.52**	**3.19**	**3.01**	7.98	**26.74**	**7.69**	33.8	**5.16**	1.56	**0.01**	**10.81**	**12.06**	**7.26**

Table 1. Results evaluated on SemanticKITTI **val** set. The method with the best performance is showcased in **bold** and the second best is showcased in underline.

Methods	IoU	mIoU	Parameters (M)
Symphonize [4]	42.19	15.04	59.31
VoxFormer [5]	42.95	12.20	57.81
HASSC [7]	**43.40**	13.34	58.43
FairScene	43.00	**15.76**	54.10

Table 2. Quantitative results of different SSC methods in terms of IoU, mIoU, and model size. The IoU and mIoU results are obtained on the SemanticKITTI **test** set. Our model achieves superior performance with a relatively lightweight architecture.

Methods	IoU	mIoU
SGN	43.60	14.55
SGN + OccMix	43.30	**14.99**
VoxFormer	44.02	12.35
VoxFormer + OccMix	44.83	**12.74**

Table 3. Our proposed OccMix can be seamlessly integrated into existing camera-based SSC frameworks. Results are reported on the SemanticKITTI **val** set.

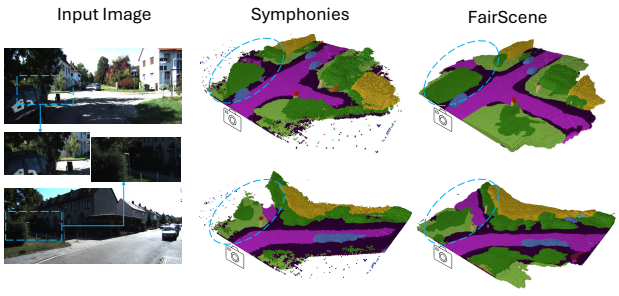

Figure 1. FairScene can better handle occlusions than Symphonize

Algorithm 1 OccMix

Require: Training sample $B_i = (\boldsymbol{I}_i, \boldsymbol{Y}_i, \boldsymbol{E}_i, \boldsymbol{R}_i, \boldsymbol{O}_i)$; mixing hyper-parameter λ

if $np.random.random() < 0.3$ **then**
  for each random training sample index $j \neq i$ **do**
    if $\boldsymbol{E}_i = \boldsymbol{E}_j$ **then**
      $B_j = (\boldsymbol{I}_j, \boldsymbol{Y}_j, \boldsymbol{E}_j, \boldsymbol{R}_j, \boldsymbol{O}_j)$
      Break
    end if
  end for
  // View-Occupancy Mixing
  $\tilde{\boldsymbol{I}} \leftarrow \lambda \cdot \boldsymbol{I}_i + (1-\lambda) \cdot \boldsymbol{I}_j$
  $\tilde{\boldsymbol{Y}} \leftarrow \lambda \cdot \boldsymbol{Y}_i + (1-\lambda) \cdot \boldsymbol{Y}_j$
  // Voxel Proposal Fusion
  $\tilde{\boldsymbol{O}} \leftarrow \boldsymbol{O}_i \cup \boldsymbol{O}_j$
  // Mixed Feature Lifting
  $\tilde{\boldsymbol{F}}^{2D} \leftarrow$ Backbone$(\tilde{\boldsymbol{I}})$; $\tilde{\boldsymbol{E}} = \boldsymbol{E}_i = \boldsymbol{E}_j$
  $\boldsymbol{F}_i^{3D} \leftarrow$ 2D-to-3D Feature Lifting$(\tilde{\boldsymbol{F}}^{2D}, \tilde{\boldsymbol{E}}, \boldsymbol{R}_i)$
  $\boldsymbol{F}_j^{3D} \leftarrow$ 2D-to-3D Feature Lifting$(\tilde{\boldsymbol{F}}^{2D}, \tilde{\boldsymbol{E}}, \boldsymbol{R}_j)$
  $\tilde{\boldsymbol{F}}^{3D} \leftarrow \lambda \cdot \boldsymbol{F}_i^{3D} + (1-\lambda) \cdot \boldsymbol{F}_j^{3D}$
end if
return $\tilde{B}_i = (\tilde{\boldsymbol{I}}, \tilde{\boldsymbol{Y}}, \tilde{\boldsymbol{E}}, \tilde{\boldsymbol{R}}, \tilde{\boldsymbol{O}})$, $\tilde{\boldsymbol{R}} = (\boldsymbol{R}_i, \boldsymbol{R}_j)$, $\tilde{\boldsymbol{F}}^{3D}$

Tab. 2 shows that FairScene achieves competitive performance while maintaining model efficiency. Specifically, FairScene achieves an IoU of 43.00 and an mIoU of 15.76, outperforming Symphonize in both IoU and mIoU while using fewer parameters (54.10M vs. 59.31M). These results demonstrate the effectiveness of our proposed architecture in balancing accuracy and model complexity. Tab. 1 presents results on the SemanticKITTI validation set, where FairScene achieves an mIoU of 16.06, outperforming all other methods in this metric. Tab. 3 highlights the effectiveness of our proposed OccMix by integrating it into existing camera-based SSC frameworks. When applied to SGN, Oc-

cMix improves the mIoU from 14.55 to 14.99. Similarly, incorporating OccMix into VoxFormer results in consistent improvements in both IoU (44.02 to 44.83) and mIoU (12.35 to 12.74). These results validate that OccMix can be seamlessly integrated into different architectures and consistently enhance their performance on the SemanticKITTI validation set, demonstrating its generalizability and effectiveness as a plug-in module for camera-based SSC tasks.

We also provide more visualization reults. Fig. 1 show that FairScene can better handle occlusions than Symphonize [4].

References

[1] Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4413–4421, 2018. 1

[2] Anh-Quan Cao and Raoul De Charette. Monoscene: Monocular 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3991–4001, 2022. 1, 2

[3] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9223–9232, 2023. 2

[4] Haoyi Jiang, Tianheng Cheng, Naiyu Gao, Haoyang Zhang, Tianwei Lin, Wenyu Liu, and Xinggang Wang. Symphonize 3d semantic scene completion with contextual instance queries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20258–20267, 2024. 2, 3

[5] Yiming Li, Zhiding Yu, Christopher Choy, Chaowei Xiao, Jose M Alvarez, Sanja Fidler, Chen Feng, and Anima Anandkumar. Voxformer: Sparse voxel transformer for camera-based 3d semantic scene completion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9087–9098, 2023. 2

[6] Jianbiao Mei, Yu Yang, Mengmeng Wang, Junyu Zhu, Jongwon Ra, Yukai Ma, Laijian Li, and Yong Liu. Camera-based 3d semantic scene completion with sparse guidance network. *IEEE Transactions on Image Processing*, 2024. 2

[7] Song Wang, Jiawei Yu, Wentong Li, Wenyu Liu, Xiaolu Liu, Junbo Chen, and Jianke Zhu. Not All Voxels are Equal: Hardness-Aware Semantic Scene Completion with Self-Distillation . In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14792–14801, Los Alamitos, CA, USA, 2024. IEEE Computer Society. 2

[8] Zhu Yu, Runmin Zhang, Jiacheng Ying, Junchen Yu, Xiaohai Hu, Lun Luo, Si-Yuan Cao, and Hui-Liang Shen. Context and geometry aware voxel transformer for semantic scene completion. *Advances in Neural Information Processing Systems*, 37: 1531–1555, 2025. 2

[9] Yunpeng Zhang, Zheng Zhu, and Dalong Du. Occformer: Dual-path transformer for vision-based 3d semantic occupancy prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9433–9443, 2023. 2