

Coupling Self-Distillation with Test Time Augmentation for effective LiDAR-Based 3D Semantic Segmentation

D. Antonarakos¹, G. Zamanakos¹ , I. Papadeas¹  and I. Pratikakis^{1,2} 

¹Democritus University of Thrace, Department of Electrical and Computer Engineering, 67100 Xanthi, Greece

²ATHENA RC, Greece

Abstract

Effective 3D perception is fundamental for spatial awareness and safe navigation in modern autonomous systems, with 3D semantic segmentation of LiDAR point clouds being a critical perception task. Recent progress in 2D vision highlights the potential of non-architectural training and inference strategies to further boost model performance. Inspired by consistency-based learning and self-distillation, this work employs such a training pipeline for robust 3D semantic segmentation in street scene understanding. Specifically, we incorporate a teacher-student knowledge self-distillation framework that integrates Test-Time Augmentation to enhance the quality of the soft labels generated by the teacher model during training and to improve inference performance. We present a comparative study on the effectiveness of the employed framework across both convolutional and attention-enhanced networks. Experimental results on the Street3D benchmark dataset demonstrate that the adopted training framework coupled with attention-enhanced networks compares favorably with the state-of-the-art for 3D semantic segmentation in the context of autonomous driving. Code is available at https://github.com/DUTH-VCG/Self_Distillation_with_TTA-main

1. Introduction

Accurate 3D perception is a critical component of spatial awareness and safe navigation in modern domains such as robotics and autonomous vehicles. An important perception task is 3D semantic segmentation, where each structural element of a 3D scene is given a semantic label. Common 3D sensors are the Light Detection and Ranging (LiDAR) sensors as well as the RGB-D cameras. LiDAR sensors are often preferred because of their high spatial resolution and accuracy, regardless of lighting conditions. LiDARs capture a 3D scene in the form of a point cloud, which is a set of spatially scattered points that preserve the geometric structure of the scene. Unlike structured image data, a point cloud is inherently sparse and non-uniform, with a decreasing density, as the distance from the sensor increases. These characteristics necessitate specialized architectures tailored for processing sparse 3D data, rather than conventional models designed for dense 2D data inputs.

Various deep learning techniques are developed to handle point cloud data. Such architectures are categorized mainly on their input data representation. The pioneering works of PointNet [QSMG17] and PointNet++ [QYSG17] operate directly on unordered point sets while others, such as MinkowskiUnet [CGS19], discretize 3D space into voxels and then assign points to voxels. Focused in computational efficiency, certain architectures project points into structured 2D grids and construct range images, as in RangeNet++ [MVBS19]. Across these paradigms, network designs

have emerged by incorporating dedicated modules that enhance feature expressiveness via attention mechanisms, designed to focus on the spatial and contextual information of a 3D scene [VZP22].

Beyond architectural design, recent advances in 2D image analysis have demonstrated the effectiveness of dedicated training techniques and inference-level strategies in improving model performance without modifying the underlying network structure. Notable among these is knowledge distillation, where a smaller student model learns from the softened outputs of a larger teacher model [HVD15]. At inference time, strategies like Test-Time Augmentation (TTA) apply multiple geometric transformations to an input and average the resulting predictions to obtain a more stable output [CG24]. In the semi-supervised learning domain, methods such as FixMatch [SBC*20] combine consistency regularization with pseudo-labeling by enforcing agreement between weakly and strongly augmented views of the same input. Collectively, these strategies highlight the potential of non-architectural improvements to enhance performance across various vision tasks.

Inspired by prior work on consistency-based learning with self distillation in autonomous driving scenarios [LDD22], in this work we employ such a training pipeline aiming to facilitate improved and robust performance for the task of 3D semantic segmentation in street scene understanding. Specifically, we incorporate a teacher-student knowledge distillation framework that integrates TTA to improve the quality of the soft labels produced by the

teacher model. We demonstrate the effectiveness of the proposed framework for two types of network architectures, convolutional and attention enhanced networks, namely the MinkowskiUnet backbone network along with its variations that rely upon attention modules. We evaluate their performance on Street3D [KVB*20] benchmark dataset focused on 3D semantic segmentation for street scene understanding.

The contribution of the presented work lies on the consistent comparative study employed among convolutional-only and attention-enhanced deep learning architectures that use a joint knowledge distillation and TTA training framework for the task of 3D semantic segmentation in LiDAR data for street scene understanding.

2. Related Work

3D semantic segmentation involves the prediction and assignment of a semantic class to each individual point of a point cloud. The architectures designed for this task are mainly categorized based on their input data representation into point-based, projection-based, voxel-based and dual representation.

Point-based methods operate directly on raw point sets without spatial discretization. The pioneering work of PointNet [QSMG17] firstly introduced this representation, by using a multi layer perceptron and a maximum pooling operation to extract a global feature vector for the point cloud. An improved version is PointNet++ [QYSG17], which creates point clusters locally in the scene and then uses the original PointNet architecture on each cluster to capture local dependencies.

Projection-based methods transform a point cloud into structured 2D representations, such as range images or bird's-eye views. This approach leverages the maturity and efficiency of standard 2D Convolutional Neural Networks (CNN) architectures while simplifying the irregular structure of a point cloud. One notable architecture is RangeNet++ [MVBS19], which projects LiDAR point clouds onto a spherical image plane and performs semantic segmentation using a standard 2D CNN. Although the projection process reduces the computational complexity, this can result in information loss due to quantization of the projected views.

Voxel-based methods convert irregular point cloud data into structured volumetric grids, also known as voxels. This regularization allows the network to learn spatial features across local neighborhoods through 3D CNN, similarly to typical 2D CNN. A representative example is VoxNet [MS15], one of the earliest architectures to apply 3D convolutions directly on voxelized data. While effective in capturing spatial context, these methods are often memory-intensive, especially at higher resolutions, due to the cubic growth of voxel grids. In order to alleviate this, a solution was built on the concept of Spatially Sparse Convolutions [Gra14]. Sparse 3D Convolutions [Gra15] (SpCpconv) apply convolution operations only if non-empty voxels are present in the 3D grid, which results in significant reduction on memory consumption. However, following this approach, new voxels can be activated in the output, potentially resulting in an increased memory allocation for deeper layers of the network. To mitigate this, Submanifold Sparse Convolutions [GvdM17] (SubSpConv) are introduced, by restricting op-

erations strictly to existing active voxels, therefore preserving the input sparsity and reducing computational overhead. A typical example of an architecture that utilizes the aforementioned efficient sparse convolutions, is MinkowskiUnet [CGS19].

Dual-based methods make use of two representations of a single point cloud, to extract fine-grained features while retaining a low computational overhead. Sparse Point Voxel CNN (SPVCNN) [TLZ*20] leverages both a point and voxel representation to retain the fine geometric details of a scene and extract more discriminative features. This is achieved through the integration of a point-voxel convolution branch, which projects the learned voxel features to points and vice versa.

Knowledge distillation is recently used to reduce the size of a network, by distilling the knowledge from a large teacher model, into a smaller student model. Such an approach is demonstrated for the task of 2D semantic segmentation [HVD15] and focuses on aligning the teacher and student models at feature level using pairwise similarities [LCL*19], the intra-class feature variation [WZJ*20], channel and spatial correlations [PH20] or self-attention [ALLX22].

Self-distillation is a form of knowledge distillation in which the student model learns from an earlier or temporally averaged version of itself, rather than from a separately pretrained teacher of a larger size. This internal teacher is continuously updated throughout training, making the approach both efficient and architecture-agnostic. For 3D point cloud data, recent approaches aim to distill knowledge using perturbed self-distillation [ZQX*21] and jointly self-distillation with TTA [LDD22].

Test Time Augmentation (TTA) has been widely used in 2D image tasks to improve the performance of trained models, by averaging the predictions of image variants [KKK20, LMA*20]. The image variants are constructed from a single image, by applying geometric transformations, such as rotation, translation and flipping. Recently, the same concept is been applied for 3D LiDAR data, by applying standard geometric transformations on the input point cloud [LDD22].

3. Proposed Methodology

In this work, we incorporate with minor modifications the training framework that integrates knowledge self-distillation with TTA, as proposed by [LDD22], to enhance the performance of a 3D sparse convolutional U-Net model [RFB15]. As a backbone network, the MinkowskiUnet [CGS19] is used, due to its popularity, simplicity and effectiveness. Additionally, we use attention enhanced networks, specifically the variations of MinkowskiUnet with Squeeze-and-Excitation (SE) [HSS18], Convolutional Block Attention Module (CBAM) [WPLK18] and Point Transformer (PT) [ZJJ*21] modules. The architecture of MinkowskiUnet is shown in Figure 2, while for the attention enhanced networks we follow the previous work of [VZP22] and place the attention modules after each convolutional block, as shown in Figure 3.

The pipeline of the proposed methodology begins with the acquisition of a point cloud. Initially, the raw point cloud is voxelized to convert the irregular input into a structured sparse tensor, where

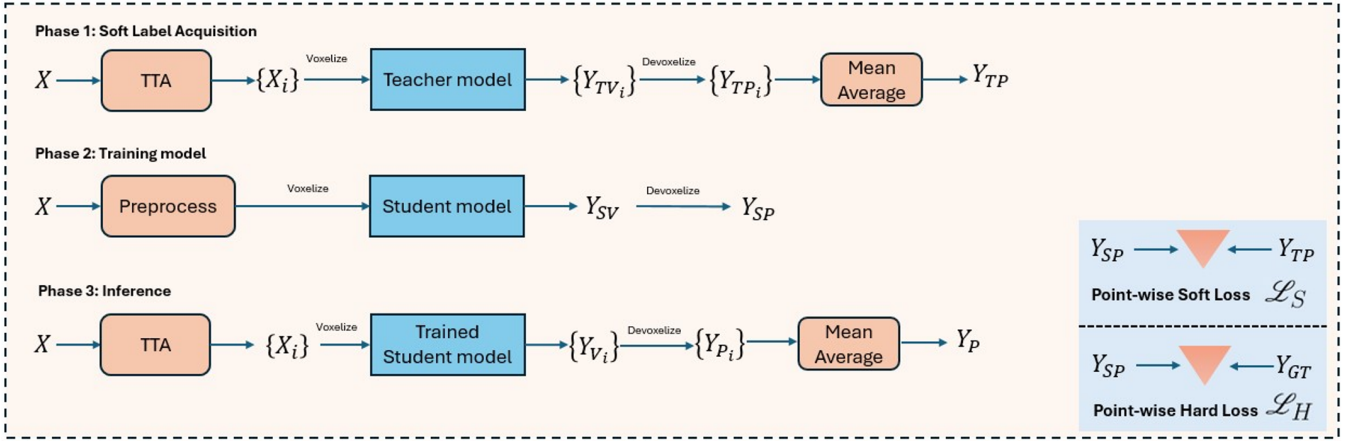


Figure 1: The training and inference pipeline. Training comprises of phase 1 & 2. Inference is shown in phase 3.

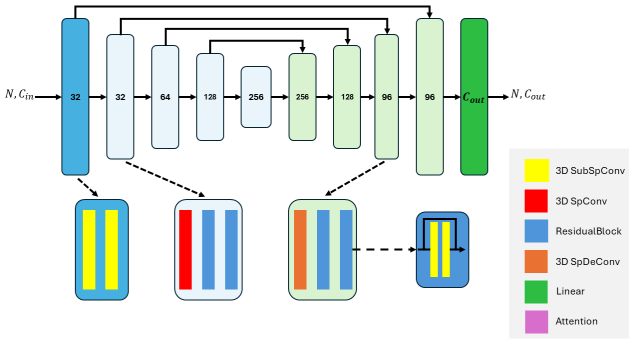


Figure 2: MinkowskiUNet backbone.

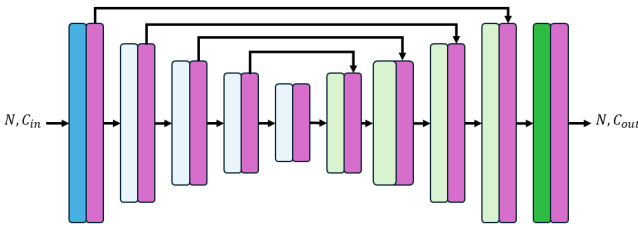


Figure 3: MinkowskiUnet backbone with added attention modules.

empty voxels are discarded. Following, the sparse representation is fed as an input to the deep learning network, which performs hierarchical feature extraction using sparse 3D convolutions. At the final layer, the network predicts a semantic class label for each non-empty voxel. The voxel-level predictions are subsequently projected back to the original point cloud via inverse mapping, where each point inherits the label of the voxel it resides in.

3.1. Self Distillation

In this work, we define the teacher model to be of a similar architecture as the student model. For both models, training starts from scratch, without using pretrained weights. At each step, the teacher is updated by simply copying the student model from the previous step.

To enable the teacher model to generate more informative soft labels for self-distillation, we incorporate a TTA strategy. Specifically, we apply four types of common augmentations used in the training phase of a 3D segmentation network, namely global scaling (Scale), random flipping along the X and Y axis (Flip), rotation along the Z axis (Rot) and translation (Tran). To address the TTA concept, we compose these augmentations into a single Compound Transformation (CT), similar to [LDD22] and realized in a sequential order, as follows:

$$CT(X) = \text{Tran}(\text{Rot}(\text{Flip}(\text{Scale}(X)))) \quad (1)$$

The resulting predictions are averaged at the logit level to produce more stable and informative soft labels for supervising the student model. A CT is also applied, multiple times, to the same input point cloud during inference. Similarly, the resulting predictions are averaged for an improved performance of the trained model.

3.2. Training Pipeline

The full training and inference pipeline is shown in Figure 1. Firstly, before passing a scene into the model, we create N_{train} augmented versions of the scene using the CT, forming a set of augmented inputs $\{X_i\}$, where $i = 1, 2, 3, \dots, N_{train}$. Each i^{th} input in the set is passed through the teacher model. The resulting voxel-wise predictions $\{Y_{TV_i}\}$ are mapped back to each point, resulting in a set of point-wise predictions $\{Y_{TP_i}\}$. The mean average of these logits forms the final teacher prediction Y_{TP} , which is used as soft labels for training. On the student side, the input is the original point cloud preprocessed only with global rotation and random scaling, excluding the full compound transformation. This preprocessed input is passed through the student model, resulting in a

voxel-wise output Y_{SV} , which is then mapped back to point-wise predictions Y_{SP} .

3.3. Loss Function

Our total loss \mathcal{L}_T consists of two components: the hard loss \mathcal{L}_H and the soft loss \mathcal{L}_S . The hard loss is defined as follows:

$$\mathcal{L}_H = \mathcal{L}_{ce'}(Y_{SP}, Y_{GT}) + \mathcal{L}_{\text{Lovasz}}(Y_{SP}, Y_{GT}) \quad (2)$$

where GT corresponds to ground truth labels. We use Lovász-Softmax loss [BRTB18] to directly optimize the IoU metric, improving segmentation performance, especially in class-imbalanced scenes. Furthermore, weighted cross-entropy (ce') is used with weights calculated as $W_c = \sqrt{\frac{1}{f_c}}$, where W_c is the weight for class c and f_c is the percentage of the appearance of class c on the total classes. Furthermore, the soft loss is defined as follows:

$$\mathcal{L}_S = \mathcal{L}_{ce}(Y_{SP}, Y_{TP}) \quad (3)$$

However, this time the cross-entropy (ce) is used without weights. Subsequently, the total loss is calculated as the sum of the two losses as:

$$\mathcal{L}_T = \mathcal{L}_H + \gamma \cdot \mathcal{L}_S \quad (4)$$

where $\gamma = \exp(\text{mIoU}(Y_{TP}, Y_{GT}))/2$. It is worthnoting that the term γ is used to control on how much the student can rely on the teacher's prediction.

3.4. Inference

During inference, only the trained student model is used to obtain the point-wise predictions. To enhance the model's performance, we apply TTA during inference by generating N_{inf} augmented versions $X_i, (i = 1, \dots, N_{inf})$ of an X point cloud scene. The resulting voxel-wise outputs $\{Y_{Vi}\}$ are then mapped back to point-wise predictions $\{Y_{Pi}\}$. The final prediction Y_P of the X scene, is obtained by computing the mean of all point-wise logits.

4. Experiments

4.1. Datasets

Street3D [KVB*20] is a LiDAR-based dataset developed by Cyclomedia Technology for use in the SHREC 2020 Track benchmark. The data were captured using a Velodyne HDL-32 sensor under clear weather and daylight conditions across various urban streets in Utrecht, Netherlands. The dataset comprises 80 annotated 3D scenes, each containing over two million points represented by their 3D spatial coordinates. For our experiments, 60 scenes are designated for training and 20 for testing, similar to the SHREC 2020 evaluation protocol. Each point is assigned one of six semantic classes, including an 'undefined' class. Evaluation metrics include Overall Accuracy (OA) and mean Intersection over Union (mIoU). In this work, we adhere to the official split for SHREC 2020 and report results on the 20 test scenes after training on the provided 60 training scenes.

4.2. Training Setup

All experiments were conducted on a desktop PC running Linux, using Python 3.9.18 and PyTorch 1.10.0. The system is equipped with an AMD Ryzen 9 3900X 12-Core Processor (24 threads) and 125 GB of RAM. A single NVIDIA GeForce RTX 3090 was used for acceleration via CUDA 11.1. The TorchSparse library [TLL*22] was employed for efficient sparse 3D convolution operations. For training, all networks are optimized using stochastic gradient descent (SGD) with Nesterov momentum [SMDH13] set to 0.9. Each model is trained for 15 epochs using a batch size of 2. The initial learning rate is set to 0.024 and is decayed over time using a cosine annealing [LH16] schedule. For the remainder training hyper-parameters, we refer the reader to our github repository given in the Abstract. All reported results correspond to the final (15th) training epoch.

All models are trained using the proposed self-distillation framework, with TTA applied to the teacher model with $N_{train} = 10$. During inference, TTA is also applied with $N_{inf} = 12$ augmented views. To investigate the impact of the augmentation size N_{train} and N_{inf} , an ablation study is conducted in Section 4.4.

4.3. Evaluation

The results for Street3D [KVB*20] are shown in Table 1 that comprise the IoU for each class along with OA and mIoU. At the same Table, the performance of SPVCNN enhanced with Point Transformer attention modules [VZP22] is shown, as it is the state of the art performing model for Street3D dataset.

Generally, it is apparent that the proposed methodology compares favorably against the baseline. The best overall performance is demonstrated by the MinkowskiUnet with PT, which results in +0.03% and +1.54% improvement compared to SPVCNN with PT, on OA and mIoU, respectively. The modest increase in OA, compared to the more substantial gain in mIoU, can be attributed to enhanced learning in classes with complex geometric structures, such as 'pole' and 'car'. In fact, across nearly all models, the most significant improvement is observed in the 'pole' class, while the 'vegetation' class consistently shows the least improvement. This pattern underscores the strength of the proposed methodology in capturing fine-grained structural details critical for identifying geometrically distinct classes. However, classes with simpler shapes and ambiguous boundaries—such as 'ground' and 'vegetation', tend to benefit less from the proposed method. In some rare cases, such classes may even be negatively affected, as seen with the MinkowskiUnet with PT, which led to a 1.54% decrease in IoU for the 'vegetation' class.

Concerning the impact of the proposed method on attention modules, it appears that PT-enhanced MinkowskiUnet benefits the most, with the SE and CBAM-enhanced models achieving smaller but consistent gains, particularly when combined with k-NN local pooling. Notably, the magnitude of improvement varies not only with the presence of attention mechanisms but also with the particular type used.

It is worthnoting that the largest overall improvement compared to the the baseline is observed by MinkowskiUnet without any attention modules, which achieves a +3.76% gain in mIoU. The same

Table 1: Evaluation of the proposed methodology for the MinkowskiUnet network and its enhanced with attention modules versions, in Street3D benchmark

Network	Attention	Type	OA	mIoU	Building	Car	Ground	Pole	Vegetation
SPVCNN* [TLZ*20]	PT {v}	baseline	98.09	90.29	94.26	88.32	98.20	75.95	94.73
MinkowskiUnet [CGS19]	-	baseline	97.49	87.51	92.59	87.47	97.57	66.93	93.00
		ours	98.16	91.27	94.81	89.42	98.17	79.64	94.3
		impr.	+0.67	+3.76	+2.22	+1.95	+0.60	+12.71	+1.30
	SE	baseline	97.83	87.19	94.06	86.79	97.94	63.24	93.94
		ours	98.22	88.93	95.38	90.23	98.23	66.23	94.56
		impr.	+0.39	+1.74	+1.32	+3.44	+0.29	+2.89	+0.62
	SE k-NN	baseline	97.81	88.10	93.42	87.99	97.78	66.57	95.36
		ours	98.24	89.56	95.20	88.77	98.32	70.71	94.81
		impr.	+0.43	+1.46	+1.78	+0.78	+0.54	+4.14	-0.55
	CBAM	baseline	98.01	86.67	94.71	87.85	98.47	59.04	93.29
		ours	98.10	87.88	95.16	89.89	98.27	62.49	93.58
		impr.	+0.09	+1.21	+0.45	+2.04	-0.20	+3.45	+0.29
	CBAM k-NN	baseline	98.10	88.42	94.76	87.07	98.14	66.95	95.17
		ours	98.44	91.13	95.55	90.15	98.47	75.87	95.59
		impr.	+0.34	+2.71	+0.79	+3.08	+0.33	+8.92	+0.42
	PT	baseline	97.98	89.17	94.39	87.34	97.81	71.30	95.03
		ours	98.12	91.83	95.72	89.48	97.84	82.63	93.49
		impr.	+0.14	+2.06	+1.33	+2.14	+0.03	+11.33	-1.54

*: Current best performing network in Street3D benchmark [VZP22].

Baseline networks are implemented by [VZP22].

impr.: Positive refers to performance improvement. Negative refers to performance deterioration.

{v}: Attention module is applied on the voxel branch of SPVCNN [TLZ*20].

baseline model also demonstrates the highest per-class improvement, with a +12.71% increase in IoU for the ‘pole’ class. The varying magnitudes of improvement across different model architectures suggest that, while the proposed method is generally effective, its compatibility with the underlying architecture plays a significant role in achieving optimal performance. Simple architectures, such as those without attention modules, achieve greater improvement due to the proposed training method, even if more complex models demonstrate a higher overall performance.

4.4. Ablation Studies

Ablation studies are conducted to identify the importance of the N_{train} and N_{inf} hyper-parameters. Experiments are performed for the MinkowskiUnet network, on Street3D dataset and are shown in Table 2 and Table 3 for N_{train} and N_{inf} , respectively. As it appears, during training, the teacher model benefits from a higher N_{train} value and provides more discriminative soft labels to the student. At inference, the student also benefits from a higher N_{inf} value, which results in an improved performance. Remarkably, it appears that there exists a connection between the N_{train} and N_{inf} hyper-parameters, since when both are used in their maximum values, the performance improvement is further boosted.

5. Conclusions

This paper has presented a consistent comparative study that examines a self-distillation training framework for a variety of attention-based deep learning architectures aiming to enhance the performance of 3D semantic segmentation models for LiDAR point clouds, a crucial task for robust spatial awareness in street scene

Table 2: Ablation study for the hyperparameter N_{train} for MinkowskiUnet, in Street3D dataset.

N_{train}	$N_{inf} = 0$		$N_{inf} = 12$	
	mIoU	Improvement	mIoU	Improvement
0	87.51	-	88.39	-
2	89.18	+1.67	90.73	+2.34
4	89.54	+2.03	90.35	+1.96
6	89.17	+1.66	90.75	+2.36
8	89.49	+1.98	90.69	+2.30
10	89.34	+1.83	91.27	+2.88

understanding. We addressed the inherent challenges of point cloud data by integrating a knowledge self-distillation paradigm with Test-Time Augmentation (TTA).

We rigorously evaluated the framework across a range of 3D sparse convolutional U-Net architectures, including the standard MinkowskiUnet and its variants augmented with SE, CBAM, and PT attention modules. Our comprehensive experimental work on the Street3D dataset demonstrates the consistent effectiveness of the proposed method. Notably, the MinkowskiUnet integrated with PT attention modules, when trained with the proposed framework, achieved state-of-the-art performance, surpassing existing methods. The more substantial improvement, particularly for geometrically complex classes like ‘pole’ and ‘car’, highlights the method’s strength in capturing fine-grained structural details. The ablation studies confirmed the positive impact of increased augmentation sizes in TTA, for both training and inference.

As a future work, we plan to explore the application of this

Table 3: Ablation study for the hyperparameter N_{inf} for MinkowskiUnet, in Street3D dataset.

N_{inf}	$N_{train} = 0$		$N_{train} = 10$	
	mIoU	Improvement	mIoU	Improvement
0	87.51	-	89.34	-
1	87.02	-0.49	89.60	+0.26
2	87.87	+0.36	90.43	+1.09
3	88.01	+0.50	90.81	+1.47
4	88.27	+0.76	91.03	+1.69
5	88.35	+0.84	91.06	+1.72
6	88.20	+0.69	91.14	+1.80
7	88.43	+0.92	91.19	+1.85
8	88.35	+0.84	91.17	+1.83
9	88.49	+0.98	91.24	+1.90
10	88.40	+0.84	91.26	+1.92
11	88.39	+0.88	91.25	+1.91
12	88.39	+0.88	91.27	+1.93

framework to larger and more diverse outdoor datasets, investigate the optimal trade-off between computational overhead and performance gains with even higher augmentation factors and delve into adaptive TTA strategies that can dynamically adjust based on scene characteristics or model uncertainty.

References

- [ALLX22] AN S., LIAO Q., LU Z., XUE J.-H.: Efficient semantic segmentation via self-attention and self-distillation. *IEEE Transactions on Intelligent Transportation Systems* 23, 9 (2022), 15256–15266. [2](#)
- [BRTB18] BERMAN M., RANNEN TRIKI A., BLASCHKO M. B.: The lovasz-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2018), pp. 4413–4421. [doi:10.1109/CVPR.2018.00464.4](#)
- [CG24] COHEN G., GIRYES R.: Simple post-training robustness using test time augmentations and random forest. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024), IEEE, p. 3996–4006. [1](#)
- [CGS19] CHOY C. B., GWAK J., SAVARESE S.: 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019), IEEE, pp. 3075–3084. [doi:10.1109/CVPR.2019.00319.1,2,5](#)
- [Gra14] GRAHAM B.: Spatially-sparse convolutional neural networks, 2014. arXiv preprint arXiv:1409.6070. [arXiv:1409.6070.2](#)
- [Gra15] GRAHAM B.: Sparse 3d convolutional neural networks, 2015. arXiv preprint arXiv:1505.02890. [arXiv:1505.02890.2](#)
- [GvdM17] GRAHAM B., VAN DER MAATEN L.: Submanifold sparse convolutional networks, 2017. arXiv preprint arXiv:1706.01307. [arXiv:1706.01307.2](#)
- [HSS18] HU J., SHEN L., SUN G.: Squeeze-and-excitation networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2018), IEEE, pp. 7132–7141. [doi:10.1109/CVPR.2018.00745.2](#)
- [HVD15] HINTON G., VINYALS O., DEAN J.: Distilling the knowledge in a neural network. In *Proceedings of Advances in Neural Information Processing Systems Workshop* (2015), pp. 1–9. [1,2](#)
- [KKK20] KIM I., KIM Y., KIM S.: Learning loss for test-time augmentation. *Advances in neural information processing systems* 33 (2020), 4163–4174. [2](#)
- [KVB*20] KU T., VELTKAMP R. C., BOOM B., DUQUE-ARIAS D., VELASCO-FORERO S., DESCHAUD J.-E., GOULETTE F., MARCOTEGUI B., ORTEGA S., TRUJILLO A., ET AL.: Shrec 2020: 3d point cloud semantic segmentation for street scenes. *Computers & Graphics* 93 (2020), 13–24. [2,4](#)
- [LCL*19] LIU Y., CHEN K., LIU C., QIN Z., LUO Z., WANG J.: Structured knowledge distillation for semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2019), pp. 2604–2613. [2](#)
- [LDD22] LI J., DAI H., DING Y.: Self-distillation for robust lidar semantic segmentation in autonomous driving. In *European Conference on Computer Vision* (2022), Springer, pp. 659–676. [1,2,3](#)
- [LH16] LOSHCHILOV I., HUTTER F.: SGDR: Stochastic gradient descent with warm restarts, 2016. arXiv preprint arXiv:1608.03983. [arXiv:1608.03983.4](#)
- [LMA*20] LYZHOV A., MOLCHANOVA Y., ASHUKHA A., MOLCHANOV D., VETROV D.: Greedy policy search: A simple baseline for learnable test-time augmentation. In *Conference on uncertainty in artificial intelligence* (2020), PMLR, pp. 1308–1317. [2](#)
- [MS15] MATURANA D., SCHERER S.: Voxnet: A 3d convolutional neural network for real-time object recognition. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2015), IEEE, pp. 922–928. [doi:10.1109/IROS.2015.7353481.2](#)
- [MVBS19] MILIOTO A., VIZZO I., BEHLEY J., STACHNISS C.: Rangenet++: Fast and accurate lidar semantic segmentation. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2019), IEEE, pp. 4213–4220. [1,2](#)
- [PH20] PARK S., HEO Y. S.: Knowledge distillation for semantic segmentation using channel and spatial correlations and adaptive cross entropy. *Sensors* 20, 16 (2020), 4616. [2](#)
- [QSMG17] QI C. R., SU H., MO K., GUIBAS L. J.: Pointnet: Deep learning on point sets for 3D classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 652–660. [1,2](#)
- [QYSG17] QI C. R., YI L., SU H., GUIBAS L. J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems* (2017), Guyon I., Luxburg U. V., Bengio S., Wallach H., Fergus R., Vishwanathan S., Garnett R., (Eds.), vol. 30, Curran Associates, Inc., pp. 5099–5108. [1,2](#)
- [RFB15] RONNEBERGER O., FISCHER P., BROX T.: U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015* (Cham, 2015), vol. 9351 of *Lecture Notes in Computer Science*, Springer, pp. 234–241. [doi:10.1007/978-3-319-24574-4_28.2](#)
- [SBC*20] SOHN K., BERTHELOT D., CARLINI N., ZHANG Z., ZHANG H., RAFFEL C., CUBUK E. D., KURAKIN A., LI C.-L.: Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in Neural Information Processing Systems (NeurIPS)* (2020). [arXiv:2001.07685.1](#)
- [SMDH13] SUTSKEVER I., MARTENS J., DAHL G., HINTON G.: On the importance of initialization and momentum in deep learning. In *Proceedings of the 30th International Conference on Machine Learning (ICML)* (2013), PMLR, pp. 1139–1147. [4](#)
- [TLL*22] TANG H., LIU S., LI X., LIN Y., HAN S.: Torchsparse: Efficient point cloud inference engine. In *Conference on Machine Learning and Systems (MLSys)* (2022). [4](#)
- [TLZ*20] TANG H., LIU Z., ZHAO S., LIN Y., LIN J., WANG H., HAN S.: Searching efficient 3D architectures with sparse point-voxel convolution. In *European Conference on Computer Vision* (2020), Springer, pp. 685–702. [2,5](#)

- [VZP22] VANIAN V., ZAMANAKOS G., PRATIKAKIS I.: Improving performance of deep learning models for 3d point cloud semantic segmentation via attention mechanisms. *Computers & Graphics* 107 (2022), 63–74. doi:10.1016/j.cag.2022.06.010. 1, 2, 4, 5
- [WPLK18] WOO S., PARK J., LEE J.-Y., KWEON I. S.: Cbam: Convolutional block attention module. In *European Conference on Computer Vision (ECCV)* (2018), vol. 11211 of *Lecture Notes in Computer Science*, Springer, pp. 3–19. doi:10.1007/978-3-030-01234-2_1. 2
- [WZJ*20] WANG Y., ZHOU W., JIANG T., BAI X., XU Y.: Intra-class feature variation distillation for semantic segmentation. In *European Conference on Computer Vision* (2020), Springer, pp. 346–362. 2
- [ZJJ*21] ZHAO H., JIANG L., JIA J., TORR P. H. S., KOLTUN V.: Point transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2021), IEEE, pp. 16259–16268. doi:10.1109/ICCV48922.2021.01595. 2
- [ZQX*21] ZHANG Y., QU Y., XIE Y., LI Z., ZHENG S., LI C.: Perturbed self-distillation: Weakly supervised large-scale point cloud semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision* (2021), pp. 15520–15528. 2