Supplementary Material for To Supervise or Not to Supervise: Understanding and Addressing the Key Challenges of Point Cloud Transfer Learning

Souhail Hadgi[1], Lei Li[2], and Maks Ovsjanikov[1]

[1] LIX, Ecole Polytechnique, IP Paris
[2] Technical University of Munich

This supplementary document provides additional details and results about the frameworks employed in our work. Specifically, we first describe the architectures that were used in our experiments in Sec. A. Then we detail in Sec. B other pre-training approaches evaluated in our study with their respective configurations. In Sec. C, we describe our transfer learning protocols and the corresponding optimization parameters. In Sec. D, we extend the analysis of the discriminative ability of early layers and gradient norm to other configurations. Finally, in Sec. E we provide additional results on our geometric regularization approach.

A Architectures

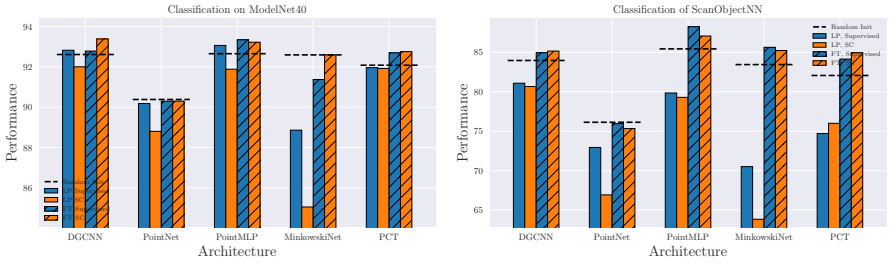
The architectures used in our experiments were selected to be as similar as possible to original implementations, some changes had to be made in order to properly implement pre-training strategies and to keep a consistent encoder architecture across them. These modifications were not intended to tailor the architectures to get the best performance, but just to keep consistency.

Unless specified otherwise, encoders share a feature embedding dimension of 1024 which is projected to a 128 per-point feature dimension for point-level contrastive-learning and 256 feature dimension for shape-level contrastive learning (see Sec. B).

DGCNN: For both pre-training and fine-tuning, we employed the six-layer deep part-segmentation architecture as proposed in the original DGCNN work [11].

PointNet: Modifications were applied to the PointNet architecture to achieve a uniform encoder suitable for both classification and segmentation tasks, resulting in performance differences from the original implementation.

PointMLP: We used the standard (non-elite) version of the original PointMLP [7] implementation of the encoder. Although no benchmark or implementation was initially provided for the semantic segmentation task, we add a simple decoder using the PointNet feature propagation module implemented in PointNet++ [9] to evaluate PointMLP on the semantic segmentation downstream task, enable point-level contrastive learning and apply layer-wise geometric regularization.



(a) Modelnet40 **(b)** ScanObjectNN

Fig. 7: Evaluation on (a) ModelNet40 and (b) ScanObjectNN classification tasks of SC pre-trained models in comparison to supervised pre-trained model, using linear probing (LP – solid bars) and fine-tuning (FT – dashed bars) settings. Random Init is a randomly initialized model.

MinkowskiNet: We adopt the SR-UNet architecture originally proposed in [4] and used in PointContrast [13]. Though this architecture is originally tuned for segmentation tasks, its encoder can be effectively used for classification tasks. We take 32 as the point feature embedding dimension, to be consistent with PointContrast implementation.

PCT: We adopt the baseline PCT architecture [5] used for classification (with downsampling module) for the classification downstream tasks. For the segmentation task, we implement a version without downsampling as proposed in the original paper since the implementation was missing. Each variation underwent its own pre-training process.

Maintaining architectural consistency while exploring various pre-training approaches posed significant challenges. Consequently, we prioritized maintaining consistent parameterization across architectures over achieving the highest possible performance.

B Pre-training

In addition to the supervised and point-level contrastive approach presented in the main paper, we provide additional results on 2 other pre-training strategies:

- **Shape-level contrastive learning (SC):** Inspired by the commonly used contrastive learning approach in 2D [3], which contrasts between entire images, we apply the same view generation technique used in the point-level contrastive learning approach (denoted as PC) to generate a positive pair of shapes. We add an MLP projection head to the encoder for contrastive loss computation.
- **Point-DAE:** We follow the Point-DAE work [14], which investigates denoising auto-encoders for self-supervised pre-training on 3D point clouds. They propose several corruption settings, but we select the affine + masking corruption for its performance and generalizability and see if observations

Table 5: Evaluation of denoising auto-encoder pre-training strategy. Shape classification on ModelNet40 and ScanObjectNN, and 3D scene segmentation on S3DIS across different pre-training strategies and architectures. Accuracy metric used for classification and mIoU metric for semantic segmentation. Bolded results represent best evaluation metric for a specific dataset and architecture setting. RI is a randomly initialized model.

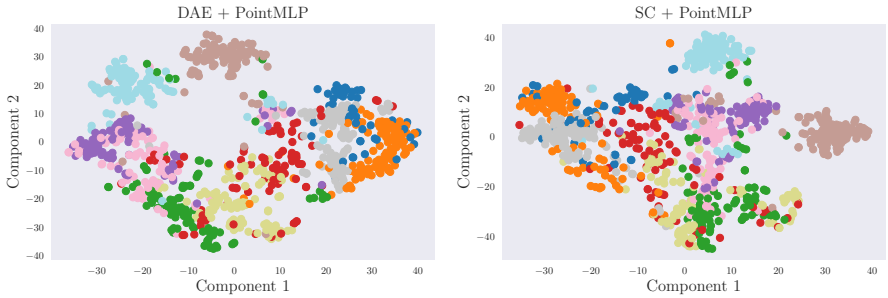
Pre-training Strategy	ModelNet40 [12]	ScanObjectNN [10]	S3DIS [1]
Linear Probing			
DAE + DGCNN	92.00	72.31	-
DAE + PointMLP	91.15	73.46	-
DAE + PCT	89.33	65.82	-
Fine-tuning			
DAE + DGCNN	**92.69**	**84.98**	**51.4**
RI + DGCNN	92.61	83.96	49.82
DAE + PointMLP	**93.18**	**85.5**	**59.24**
RI + PointMLP	92.65	85.43	56.59
DAE + PCT	**91.4**	**77.65**	**50.87**
RI + PCT	91.19	76.72	50.8

made in our work are still relevant for this type of pre-training. One advantage of this simple scheme is that it can be used with different type of 3D backbones, unlike other masked pre-training methods that are exclusive to vanilla or vision transformers (*e.g.* Point-MAE [8])

Although we focus on contrastive vs supervised pre-training, our evaluations span supervised, point-level, and shape-level contrastive pre-training methods, alongside a reconstruction-based approach. We concentrate on single modality (3D) pre-training, but multi-modal pre-training methods are also viable for comparison.

Efforts were made to standardize pre-training settings across architectures and methods to reduce biases due to parameter differences.

When using contrastive pre-training, we introduce a set of invariances by applying data augmentation. We start by normalizing the point cloud and perform random geometric transformations, including translation with a magnitude of 0.5, scaling between 80% and 125% and rotation of magnitude 45°. These transformations and their values are typically used in data augmentation schemes for object-level datasets. We additionally simulate partial data to obtain complex shapes that are more challenging for the contrastive pretext task. This is done by cropping the original shape to a certain percentage. We experimented with several crop ratios ranging from 0.2 to 0.8 and found that 0.5 gives the best evaluation performance.



(a) Point-DAE **(b)** SC

Fig. 8: t-SNE plots of the first-layer feature activation for PointMLP under (a) Point-DAE and (b) SC pre-training. Both pre-training strategies produce discriminative early layers.

The ADAM optimizer [6], with a learning rate of 10^{-3} and a weight decay of 10^{-6}, is used for 100 epochs and a batch size of 32. Although this epoch count may seem low for Point-DAE pre-training, it was chosen to maintain consistency in comparisons. Temperature settings for SC and PC pre-training are set at 0.1 and 0.4, respectively, following conventions established in PointContrast [13]. For MinkowskiNet, a voxel size of 0.1 was selected after testing ranges from 0.1 to 0.5, showing optimal pre-training outcomes. Pre-training is conducted on the entire ShapeNetCoreV2 dataset [2].

C Evaluation

C.1 Settings

We maintained consistent linear probing and fine-tuning optimization parameters across downstream classification tasks, utilizing an SGD optimizer with a learning rate of 10^{-3}, a weight decay of 10^{-6}, training for 200 epochs, and a batch size of 32. For semantic segmentation tasks on S3DIS, the batch size was adjusted to 24, and the training duration was set to 100 epochs. For MinkowskiNet, we applied a voxel size of 0.1 for shape classification and 0.5 for semantic segmentation on S3DIS.

Data augmentation for ModelNet40 classification included random geometric transformations such as translations (up to 0.2 units) and scalings (between two-thirds and one and a half times the original size), with additional rotations for ScanObjectNN classification. For MinkowskiNet, we adopted the PointContrast data augmentation scheme, excluding transformations for semantic segmentation tasks.

In our primary study, linear probing was not conducted for semantic segmentation due to the absence of a pre-trained decoder for supervised and SC contrastive pre-training. This approach is less common for this task, as encoder

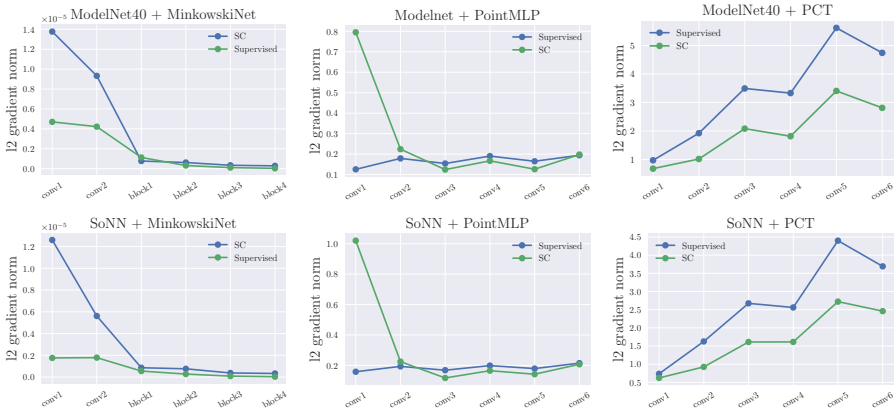


Fig. 9: Gradient norm analysis for pre-trained models. Evaluation includes SC and Supervised pre-training across MinkowskiNet, PointMLP, and PCT. The x-axis represents convolutional layers at various depths.

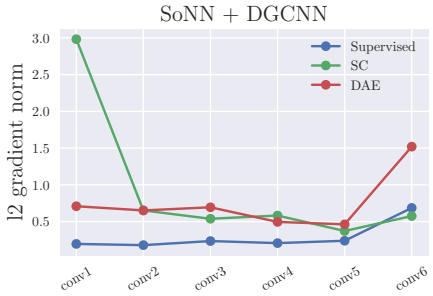


Fig. 10: Gradient norm analysis for pre-trained models. Evaluation includes SC, Point-DAE and Supervised pre-training. The x-axis represents convolutional layers at various depths.

weights typically generate feature embeddings for the entire input, not point-wise. For PC pre-training with DGCNN, keeping the encoder and decoder frozen while training only the last layer resulted in a significantly lower mIoU score (24.09%) compared to fine-tuning (49.99%).

C.2 Additional results

Shape-level contrastive pre-training (SC). We evaluate transfer learning performance of shape-level contrastive learning in comparison to supervised pre-training in Figure 7. Compared to the conclusions made for point-level contrastive learning (PC) being better for fine-tuning but worse for linear probing than supervised pre-training, we find that SC follows the same pattern except in cases like MinkowskiNet where it under-performs. This showcases that deep

Table 6: Fine-tuning of geometric regularized pre-trained models on different downstream data/tasks. Shape classification on ModelNet40 and ScanObjectNN with more architectures. Regularization can improve downstream performance of supervised pre-training.

Pre-training Strategy	PointNet	MinkowskiNet	PointMLP	PCT
ModelNet40 accuracy				
Supervised	90.30	91.37	92.65	91.56
Supervised + regularization	**90.34**	**92.54**	**92.94**	**91.64**
ScanObjectNN accuracy				
Supervised	75.95	85.63	**88.24**	**78.07**
Supervised + regularization	**76.68**	**85.81**	87.23	77.69

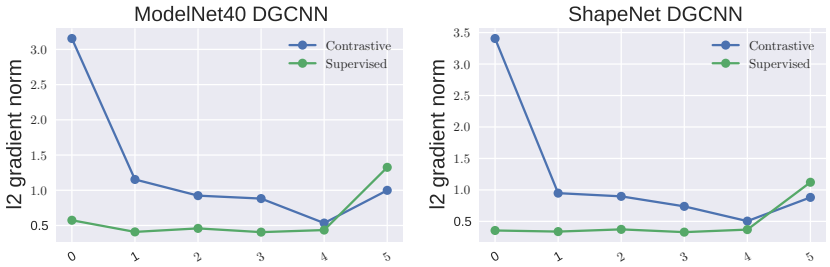


Fig. 11: Layer-wise gradient norm of a model pre-trained on ScanObjectNN for two different downstream datasets.

sparse architectures prefer point-wise pretext tasks.

Reconstruction type pre-training. Results for Point-DAE, a denoising autoencoder (with affine and masking corruptions), indicate modest linear probing performance but promising fine-tuning results, especially in semantic segmentation tasks, highlighting the potential of reconstruction-based pre-training. The superior performance in semantic segmentation is correlated to the nature of the pretext task, which is focused on the point-level rather than the shape-level, as for contrastive pre-training. We specifically avoided MinkowskiNet because of the variable number of points in an input which makes it incompatible with the baseline framework of Point-DAE, and PointNet because of its global architecture which overfits to the pretext task.

D Analysis

Discriminative capability of early layers. As depicted in Figure 8, both shape-level contrastive learning (SC) and reconstruction-type pre-training (Point-DAE) lead to distinguishable clusters in feature space, indicating that discriminative early layers are common across pre-training strategies. This effect, not observed in 2D vision as discussed in the main paper, can be attributed to the

Table 7: Fine-tuning of geometric regularized pre-trained models on semantic segmentation of S3DIS scenes. For relevant architectures, regularization can improve downstream performance of supervised pre-training.

Pre-training Strategy	PointMLP	PCT
	S3DIS mIoU	
Supervised	55.7	50.7
Supervised + regularization	**56.74**	**51**

Table 8: Accuracy evaluation of geometric regularization on different layers for DGCNN. Using only the first two layers of DGCNN provides the best features for fine-tuning on ModelNet40 classification task.

Layer regularized	conv1	conv2	conv3	conv4	Output layer
Supervised + layer-regularization	92.65	**93.34**	93.18	92.9	93.06

nature of 3D inputs.

Gradient norm of layer weights for other architectures. We generalize the results found in the main paper on gradient norm of layer weights for different architectures and other pre-training strategies. First, we see in Figure 9 that supervised pre-training produces low gradient norm value for first layers across all studied architectures, although the exact pattern of the gradient norm curve changes through different convolutional layers from architecture to another. Second, these results correspond to shape-level contrastive learning, where the results resemble the point-level contrastive learning one. Third, reconstruction-type pre-training strategies like Point-DAE result in higher gradient norm values, as shown in Figure 10, but still lower than those from contrastive learning strategies, potentially explaining the reduced performance of Point-DAE.

Gradient norm of layer weights for other source data. We highlight that our key observations on the difference of adaptability between pre-training methods are not unique to the ShapeNet source data. As shown in Figure 11, in the setting where source data (data used for pre-training) is ScanObjectNN, the early layers are also less adaptable for supervised pre-training compared to contrastive pre-training.

E Regularization

Our geometric regularization technique proved beneficial in enhancing the performance of supervised pre-training for the DGCNN architecture, as reported in Table 3 of the main document. Expanding upon this, Tables 6 and 7 illustrate the effectiveness of this method for additional architectures across various tasks. The observed improvements across different architectures indicate the versatility of

our regularization method in augmenting pre-training effectiveness, irrespective of the specific model.

Particularly, for architectures like PointMLP, which already show a preference for supervised over contrastive pre-training, geometric regularization further elevates performance in challenging scenarios, such as S3DIS scene segmentation, where supervised pre-training alone may fall short.

Through empirical testing to determine the optimal layers for regularization, we discovered that targeting early layers for regularization yields better outcomes than applying it to the encoder's output layer alone. This finding, presented in Table 8, suggests future research should carefully consider which layers to regularize, as early layers appear more beneficial for this purpose, potentially guiding more effective regularization strategies in 3D model pre-training.

References

1. Armeni, I., Sener, O., Zamir, A.R., Jiang, H., Brilakis, I., Fischer, M., Savarese, S.: 3D semantic parsing of large-scale indoor spaces. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1534–1543 (2016)
2. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., Xiao, J., Yi, L., Yu, F.: ShapeNet: An information-rich 3d model repository. Tech. rep. (2015)
3. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. pp. 1597–1607. PMLR (2020)
4. Choy, C., Gwak, J., Savarese, S.: 4d spatio-temporal convnets: Minkowski convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3075–3084 (2019)
5. Guo, M.H., Cai, J.X., Liu, Z.N., Mu, T.J., Martin, R.R., Hu, S.M.: Pct: Point cloud transformer. Computational Visual Media **7**, 187–199 (2021)
6. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
7. Ma, X., Qin, C., You, H., Ran, H., Fu, Y.: Rethinking network design and local geometry in point cloud: A simple residual mlp framework. arXiv preprint arXiv:2202.07123 (2022)
8. Pang, Y., Wang, W., Tay, F.E., Liu, W., Tian, Y., Yuan, L.: Masked autoencoders for point cloud self-supervised learning. In: European conference on computer vision. pp. 604–621. Springer (2022)
9. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems **30** (2017)
10. Uy, M.A., Pham, Q.H., Hua, B.S., Nguyen, T., Yeung, S.K.: Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 1588–1597 (2019)
11. Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M.: Dynamic graph cnn for learning on point clouds. Acm Transactions On Graphics (TOG) **38**(5), 1–12 (2019)

12. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3D shapenets: A deep representation for volumetric shapes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1912–1920 (2015)
13. Xie, S., Gu, J., Guo, D., Qi, C.R., Guibas, L., Litany, O.: Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In: European conference on computer vision. pp. 574–591 (2020)
14. Zhang, Y., Lin, J., Li, R., Jia, K., Zhang, L.: Point-dae: Denoising autoencoders for self-supervised point cloud learning. arXiv preprint arXiv:2211.06841 (2022)