RESSCAL3D: RESOLUTION SCALABLE 3D SEMANTIC SEGMENTATION OF POINT CLOUDS

Remco Royen⋆†*, Adrian Munteanu*⋆†

⋆Department ETRO, Vrije Universiteit Brussel, Pleinlaan 2, B-1050 Brussels, Belgium
†imec, Kapeldreef 75, B-3001 Leuven, Belgium
Email: {*remco.royen, adrian.munteanu*}@vub.be

ABSTRACT

While deep learning-based methods have demonstrated outstanding results in numerous domains, some important functionalities are missing. Resolution scalability is one of them. In this work, we introduce a novel architecture, dubbed RESSCAL3D, providing resolution-scalable 3D semantic segmentation of point clouds. In contrast to existing works, the proposed method does not require the whole point cloud to be available to start inference. Once a low-resolution version of the input point cloud is available, first semantic predictions can be generated in an extremely fast manner. This enables early decision-making in subsequent processing steps. As additional points become available, these are processed in parallel. To improve performance, features from previously computed scales are employed as prior knowledge at the current scale. Our experiments show that RESSCAL3D is 31-62% faster than the non-scalable baseline while keeping a limited impact on performance. To the best of our knowledge, the proposed method is the first to propose a resolution-scalable approach for 3D semantic segmentation of point clouds based on deep learning.

Index Terms— Resolution scalability, point cloud processing, semantic segmentation, scalable data acquisition

1. INTRODUCTION

In recent years, deep learning has shown great potential in different domains such as compression [1, 2], 6D pose estimation [3, 4] and semantic segmentation [5, 6]. While most papers focus on pure performance and are able to outperform traditional methods significantly, less attention has been given to practical features. One such feature is scalability.

Scalability is a broad term that can be applied on different aspects of deep learning, leading to different subdomains. In [7, 8], techniques are proposed allowing the selection of the model complexity at runtime depending on the available computing resources, thus achieving complexity scalability. In [9], a novel layer, called MaskLayer, is proposed that provides quality scalability with applications presented in compression and semantic hashing. The domain of resolution scalability allows operating at different resolutions, dependent on the application or available data. It has proven to be an important feature in traditional compression algorithms [10, 11, 12] and, more recently, in a point cloud geometry codec [13]. While all of these methods provide various scalability functionalities, full-resolution point cloud data is required to be available at the start of inference as input for these methods. Consequently, existing methods are not able to handle varying spatial resolutions of the input point cloud.

In addition, existing methods are not able to progressively process additional points in the input point cloud as they become available over time. The recent advent of scalable 3D acquisition devices [14, 15] enables the acquisition of point clouds of which the densities increase progressively over time. Such resolution-scalable 3D scanning devices generate a low spatial resolution of the scene with extremely small latency, and progressively increase the resolution of the acquired point cloud over time. An important advantage of this new 3D scanning paradigm is that it enables processing the sparse point cloud while higher resolutions are captured. Once new points are captured, the results are refined.

In this work, we propose a novel method, dubbed RESSCAL3D, that allows processing the point cloud data in a resolution-scalable manner. It allows processing low resolution 3D point clouds while higher resolutions are still being captured by the scanning device. When extra points become available, instead of restarting processing for all points, leading to large delays, the proposed method processes only the new points. To improve performance, processing of any given spatial resolution employs the information obtained at the lower spatial resolutions as prior information. This reduces the processing time. Another advantage is that early
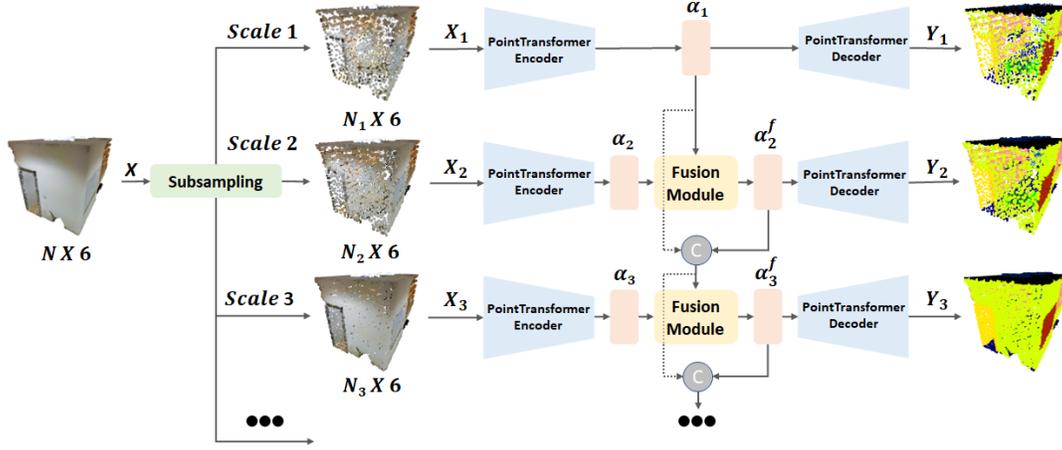
Fig. 1: The RESSCAL3D architecture. The grey circle with 'C' stands for concatenation.
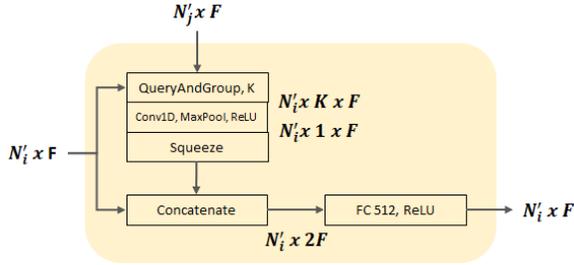

Fig. 2: RESSCAL3D fusion module

decision-making is enabled as intermediate predictions on the lower resolutions are retrieved very fast.

To evaluate the proposed architecture, semantic segmentation was chosen as target application. 3D scene understanding is of critical importance for many application domains, such as virtual reality, autonomous driving, and robotics, where timing is crucial. To this end, a fundamental component is 3D semantic segmentation [6, 16, 17, 18, 5, 19]. We highlight PointTransformer [5] which yields state of the art results by using the Transformer architecture for this task.

Summarized, our main contributions are as follows:

- The first deep learning-based approach, to the best of our knowledge, that provides resolution scalable 3D semantic segmentation

- A fusion module that fuses features from different resolution levels

- An experimental analysis on S3DIS. While minimizing the cost of scalability, RESSCAL3D is 31-62% faster than the non-scalable baseline at the highest spatial resolution. Additionally, intermediate results are generated, the fastest after only 6% of the total inference time of the baseline.

The paper is structured as follows: Sec. 2 introduces the proposed approach. Sec. 3 and Sec. 4 present our experimental results and ablation study, respectively. Finally, Sec. 5 concludes this work.

2. PROPOSED METHOD

Overview of the proposed method. The RESSCAL3D architecture is illustrated in Fig. 1. To retrieve the multi-resolution data, the complete input sample $\boldsymbol{X} \in \mathbb{R}^{N \times C}$, with N and C the number of points and channels, respectively, is subsampled in s different, non-overlapping partitions. We will denote these partitions as $\boldsymbol{X}_i \in \mathbb{R}^{N_i \times C}$ with $i \in [1, s]$ and $N_1 < ... < N_s < N$. The employed subsampling method is described in Sec. 3.

Firstly, the partition with the lowest resolution, $\boldsymbol{X}_1$, is processed by a PointTransformer [5], resulting in a prediction $\boldsymbol{Y}_1 \in \mathbb{R}^{N_1}$. As $N_1 << N$, the computational complexity of this first scale is low and a fast prediction can be obtained. The second scale receives as input $\boldsymbol{X}_2$, which is processed by another PointTransformer encoder to produce the features $\boldsymbol{\alpha}_2 \in \mathbb{R}^{N_2' \times F}$, with N_2' and F the number of subsampled points by the encoder and features, respectively. In order to improve performance, those features are fused with the already computed features of lower scales by a fusion module. The resulting multi-resolution features $\boldsymbol{\alpha}_2^f$ are employed by the decoder to obtain $\boldsymbol{Y}_2$. At higher scales, the input of the fusion module is the concatenation of the fused features of previous scales. Once all scales are processed, $\boldsymbol{Y} = \{\boldsymbol{Y}_1, \boldsymbol{Y}_2, \ldots, \boldsymbol{Y}_s\} \in \mathbb{R}^N$ is obtained.

Regarding computational complexity, the presented approach has a benefit over handling all the data at once. Since PointTransformer uses an attention mechanism that requires the computation of the K-Nearest Neighbors (KNN), the complexity of processing the input as a whole can be expressed as: $O(N^2) = O((N_1 + ... + N_s)^2) = O(N_1^2 + ... + N_s^2 +$

$2 \sum_{k=1}^{s} \sum_{p=1, p \neq k}^{s} N_k N_p)$, with $N = N_1 + ... + N_s$, and considering s scales.

With RESSCAL3D, the attention mechanism is applied in parallel on the partitions, leading to complexity of order $O(N_1^2 + ... + N_s^2)$. Compared to the non-scalable approach, RESSCAL3D substantially lowers complexity with a factor proportional to:

$$\sum_{k=1}^{s} \sum_{p=1, p \neq k}^{s} N_k N_p. \quad (1)$$

With a larger s, this effect becomes more pronounced as the partition sizes become smaller. It should be noted that sequential processing of scales also brings some computational redundancy, though the KNN is the most computational expensive operation. For large N and a large amount of scales, the effect of the double product elimination becomes significant. Experimental validation of the introduced concepts is further reported in Sec. 3.
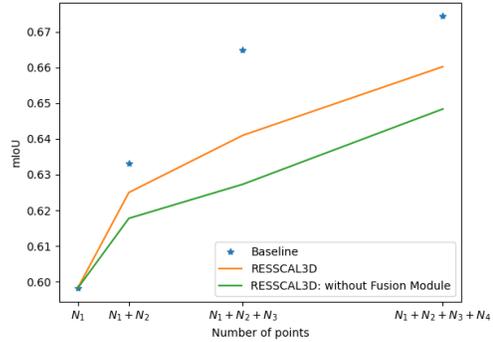
Fusion Module. Let $\boldsymbol{\alpha}_{i-1}^c \in \mathbb{R}^{N_j' \times F}$ be the concatenated features from the lower scales with N_j' the number of concatenated points in feature space. Given $\boldsymbol{\alpha}_{i-1}^c$ and $\boldsymbol{\alpha}_i$, the fusion module combines the multi-scale information into a single feature matrix which is used for decoding. The fusion architecture is depicted in Fig. 2. In more detail, the fusion module firstly retrieves the relevant features from the previous scales. This is done with a KNN algorithm on the points associated to the features in $\boldsymbol{\alpha}_i$. In other words, for each feature vector in $\boldsymbol{\alpha}_i$, the features of the K nearest neighbors in $\boldsymbol{\alpha}_{i-1}^c$ are utilized. As these features are originating from different resolution scales, the acquired feature matrices contain multi-resolution information. In a next step, these neighborhoods are processed by a Conv1D, followed by a MaxPool layer. After concatenation with the original scale features, $\boldsymbol{\alpha}_i$, a fully-connected layer encodes the information back to the original feature size.

Training. RESSCAL3D is trained scale by scale. All weights from previous scales are freezed while training an extra scale and the loss-function is computed only on the results from the current scale. This allows the PointTransformer backbone to achieve maximal results for each resolution.

3. EXPERIMENTS

Dataset and Evaluation metrics. The Stanford 3D Indoor Scene dataset (S3DIS) [20] consists of 6 large-scale indoor areas with in total 271 rooms. Each point has been annotated with one of the 13 semantic categories. Area-5 has been captured in a different building than the other areas and is therefore often selected as test set [5, 18, 19]. As evaluation metrics, the mean intersection over union (mIoU), mean accuracy (mAcc) and overall accuracy (oAcc) are being used. All presented results are averaged over the Area-5 testset.
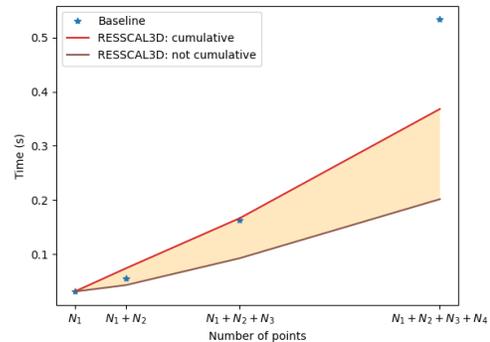
Implementation details. PointTransformer [5] has been selected as backbone architecture as it achieves state of the art
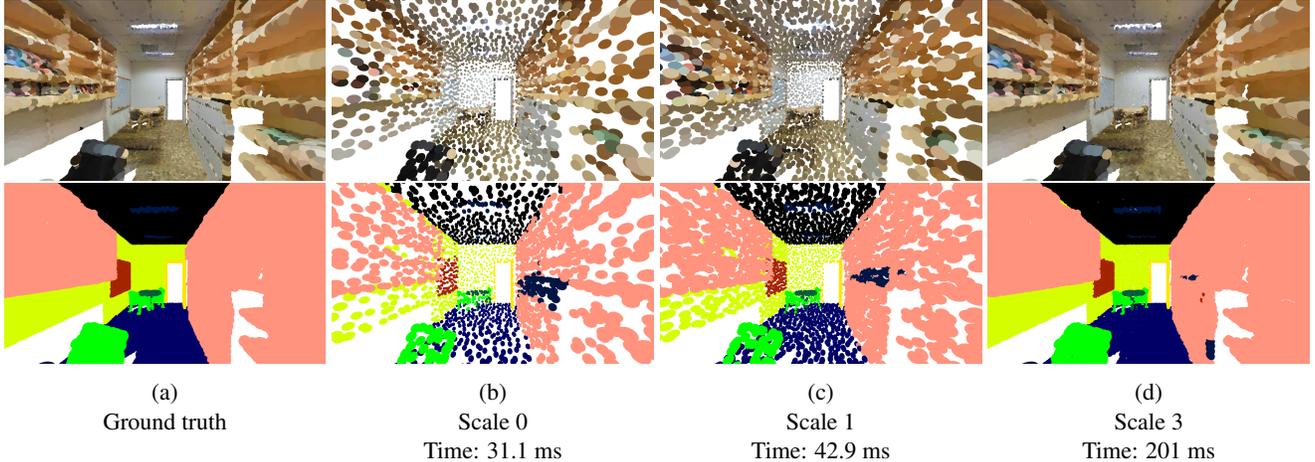

Fig. 3: Ablation study and comparison of the scalable RESSCAL3D with the non-scalable baseline

performance and has official, publicly available code. Each scale was trained for 34 epochs with a batch size of 4. Other training and network parameters are the same as in [5]. Each input point is represented by a 6-dimensional vector: $xyzrgb$. To obtain the multi-resolution data, $\boldsymbol{X}$ is voxelized s times with s different voxel sizes. Subsequently, one point per voxel is randomly selected while making sure a point is not present in multiple partitions. More specifically, we have opted to employ 4 scales with voxel sizes [0.16, 0.12, 0.08, 0.06].

S3DIS semantic segmentation. RESSCAL3D is, to the best of our knowledge, the first method that performs resolution-scalable 3D semantic segmentation of point clouds. Consequently, no quantitative or qualitative comparison with existing methods can be made. Nevertheless, in order to characterize its performance and inference time, we compare the proposed method with the non-scalable baseline, which employs the same semantic segmentation backbone for the different scales. Our scalable approach processes the additional

Fig. 4: Comparison of RESSCAL3D with the non-scalable baseline in inference time. The actual inference latency is bounded to the yellow zone. The displayed non-scalable baseline timing results are not cumulative.

(a)	(b)	(c)	(d)
Ground truth	Scale 0 Time: 31.1 ms	Scale 1 Time: 42.9 ms	Scale 3 Time: 201 ms

Fig. 5: Visualization of S3DIS results for RESSCAL3D. The input data and semantic prediction are visualized on the top and bottom row, respectively. Non-cumulative time is used.

points at each scale, using side information from the previous scales, the baseline processes the whole point cloud at that scale. Thus, the latter can only be launched when all data is available and does not process data in a progressive manner. Also, no intermediate results are obtained.

The results in terms of mIoU of our scalable approach, with and without the fusion module, and the non-scalable baseline are shown in Fig. 3. At the first scale, all methods operate in an identical manner and thus, achieve equal performance. At higher scales, using the proposed fusion module reduces the performance gap between the scalable approach and non-scalable baseline. At the highest scale, the performance gap in mIoU is only 2.1% of the total performance. Although the proposed method is not able to achieve the same performance as the baseline at the highest scale, the resulting difference is deemed small.

On the other hand, the scalable approach presents an important advantage in inference time. In Fig. 4, the inference time can be compared. Important to note is that the latency invoked by RESSCAL3D depends on the data availability. Since the scalable approach can start processing lower resolutions while higher resolutions are being acquired, it utilizes the otherwise lost acquisition time, while the non-scalable baseline can only start once all point cloud data is available. Therefore we have opted to present the upper and lower bounds of the induced latency by RESSCAL3D. When operating on the upper bound, all data is available at the start and all inference timings are cumulated. For the latter, the processing of the previous scale is finished before the point cloud data for the current resolution becomes available. Therefore, the latency introduced by RESSCAL3D will be in the yellow zone (see Fig. 4) and is mainly lower than the baseline. Important to note is that even if operating on the upper bound, RESSCAL3D is able to attain a 31% decrease in inference time at the highest scale with respect to the

Scale	Method	Performance			Time (ms)
		oAcc	mAcc	mIoU	
0	Without Fusion	85.7	67.9	59.8	31.1
	Fusion	85.7	67.9	59.8	31.1
1	Without Fusion	86.5	69.6	61.8	**73.8**
	Fusion	**87.0**	**70.0**	**62.5**	73.9
3	Without Fusion	87.0	70.5	62.7	167
	Fusion	**87.6**	**71.2**	**64.1**	167
4	Without Fusion	87.8	72.4	64.8	368
	Fusion	**88.5**	**73.0**	**66.0**	368

Table 1: Ablation of fusion module. Cumulative time is used.

baseline. The main reason is the reduced complexity of the attention modules as explained in Sec. 2. In the lower bound case, RESSCAL3D achieves an impressive 61% decrease in inference time. When operating with higher number of points, the gain will become even more pronounced (Eq. (1)).

Qualitative results are presented in Fig. 5. Overall, one can notice very accurate segmentation, with some errors on the lower scales corrected at the higher scales. An example is the erroneous dark segmentation on the bookcase on the right.

4. ABLATION STUDY

In this section, the effect and value of our fusion module is analysed. The removal of the fusion module leads to the loss of multi-resolution processing and scales which are processed independently. In Fig. 3 and Tab. 1 is shown that employing the fusion module consistently leads to better results. The added inference time is negligible.

5. CONCLUSION

In this paper, we propose RESSCAL3D, a novel architecture allowing resolution scalable 3D semantic segmentation of point clouds. The experiments show that our scale-by-scale approach allows significantly faster inference while maintaining a limited impact on performance relative to the non-scalable baseline.

6. REFERENCES

[1] Yiqun Xu, Qian Yin, Shanshe Wang, Xinfeng Zhang, Siwei Ma, and Wen Gao, "Multi-scale end-to-end learning for point cloud geometry compression," in *2022 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2022, pp. 2107–2111. 1

[2] Ionut Schiopu and Adrian Munteanu, "Deep-learning-based lossless image coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 7, pp. 1829–1842, 2019. 1

[3] Lei Ke, Shichao Li, Yanan Sun, Yu-Wing Tai, and Chi-Keung Tang, "Gsnet: Joint vehicle pose and shape reconstruction with geometrical and scene-aware supervision," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV 16*. Springer, 2020, pp. 515–532. 1

[4] Yangxintong Lyu, Remco Royen, and Adrian Munteanu, "Mono6d: Monocular vehicle 6d pose estimation with 3d priors," in *2022 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2022, pp. 2187–2191. 1

[5] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun, "Point transformer," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 16259–16268. 1, 2, 3

[6] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in neural information processing systems*, vol. 30, 2017. 1, 2

[7] Mingxing Tan, Ruoming Pang, and Quoc V Le, "Efficientdet: Scalable and efficient object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10781–10790. 1

[8] Ji Lin, Yongming Rao, Jiwen Lu, and Jie Zhou, "Runtime neural pruning," *Advances in neural information processing systems*, vol. 30, 2017. 1

[9] Remco Royen, Leon Denis, Quentin Bolsee, Pengpeng Hu, and Adrian Munteanu, "Masklayer: Enabling scalable deep learning solutions by training embedded feature sets," *Neural Networks*, vol. 137, pp. 43–53, 2021. 1

[10] Leon Denis, Shahid M Satti, Adrian Munteanu, Jan Cornelis, and Peter Schelkens, "Scalable intraband and composite wavelet-based coding of semiregular meshes," *IEEE Transactions on Multimedia*, vol. 12, no. 8, pp. 773–789, 2010. 1

[11] Jonathan Taquet and Claude Labit, "Hierarchical oriented predictions for resolution scalable lossless and near-lossless compression of ct and mri biomedical images," *IEEE Transactions on image processing*, vol. 21, no. 5, pp. 2641–2652, 2012. 1

[12] Jonas El Sayeh Khalil, Adrian Munteanu, and Peter Lambert, "Scalable wavelet-based coding of irregular meshes with interactive region-of-interest support," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 7, pp. 2067–2081, 2018. 1

[13] André FR Guarda, Nuno MM Rodrigues, and Fernando Pereira, "Deep learning-based point cloud geometry coding with resolution scalability," in *2020 IEEE 22nd International Workshop on Multimedia Signal Processing (MMSP)*. IEEE, 2020, pp. 1–6. 1

[14] Ward van der Tempel, Robert Collier, Kostas Pataridis, Ségolène Rogge, Arman Alaie, Jean-Sébastien Staelens, Mahmoud Shahin, Johannes Peeters, André Miodezky, and Christian Mourad, "Low power, low latency 3d perception for xr," presented at SPIE AR—VR—MR, 2023. 1

[15] "Voxelsensors," https://voxelsensors.com/, Accessed: 2023-02-17. 1

[16] Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhan Di, and Baoquan Chen, "Pointcnn: Convolution on x-transformed points," *Advances in neural information processing systems*, vol. 31, 2018. 2

[17] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas, "Kpconv: Flexible and deformable convolution for point clouds," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6411–6420. 2

[18] Hengshuang Zhao, Li Jiang, Chi-Wing Fu, and Jiaya Jia, "Pointweb: Enhancing local neighborhood features for point cloud processing," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 5565–5573. 2, 3

[19] Guocheng Qian, Yuchen Li, Houwen Peng, Jinjie Mai, Hasan Abed Al Kader Hammoud, Mohamed Elhoseiny, and Bernard Ghanem, "Pointnext: Revisiting pointnet++ with improved training and scaling strategies," *arXiv preprint arXiv:2206.04670*, 2022. 2, 3

[20] Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese, "3d semantic parsing of large-scale indoor spaces," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1534–1543. 3