How Many Tokens Do 3D Point Cloud Transformer Architectures Really Need?

Tuan Anh Tran [1] **Duy Minh Ho Nguyen** [2 1 3] **Hoai-Chau Tran** [4] **Michael Barz** [1] **Khoa D. Doan** [4]
Roger Wattenhofer [5] **Vien Anh Ngo** [6] **Mathias Niepert** [2 3] **Daniel Sonntag** [1 7] **Paul Swoboda** [8]

Abstract

Recent advances in 3D point cloud transformers have led to state-of-the-art results in tasks such as semantic segmentation and reconstruction. However, these models typically rely on dense token representations, incurring high computational and memory costs during training and inference. In this work, we present an efficient token merging strategy that drastically reduces the token count by up to 90–95% while preserving competitive performance. Our approach estimates token importance by leveraging spatial structures within the 3D point cloud, enabling aggressive token reduction with minimal degradation in accuracy. This finding challenges the prevailing assumption that more tokens inherently yield better performance and highlights that many current models are over-tokenized and under-optimized for scalability. We validate our method across multiple 3D vision tasks and show consistent improvements in computational efficiency. Our ongoing work will release code and detailed benchmarks to support reproducibility and further system-level exploration of efficient foundation models for 3D data. We release our implementations at this Github.

1. Introduction

The rise of transformer-based architectures has significantly advanced the field of 3D point cloud understanding (Guo et al., 2020; Zhang et al., 2022b; Fang et al., 2023), particularly in tasks such as semantic segmentation (Lai et al., 2022; Wang, 2023; Yang et al., 2025; Lai et al., 2023), and reconstruction (Kong et al., 2024; Chen et al., 2024; Tang et al., 2024; Chen et al., 2025). Inspired by the success of attention mechanisms in NLP (Vaswani et al., 2017; Devlin et al., 2019; Achiam et al., 2023) and 2D vision (Dosovitskiy et al., 2021; Carion et al., 2020; Liu et al., 2021; Kirillov et al., 2023), Point Transformer (PTv) models introduced self-attention tailored for point clouds (Zhao et al., 2021; Wu et al., 2022; 2024). Among them, PTv-3 has emerged as a powerful backbone, thanks to its hierarchical attention design and ability to capture both local and global spatial dependencies. It excels in dense 3D segmentation (Fan et al., 2024; Wu et al., 2025), reconstruction (Chen et al., 2025; Fan et al., 2024).

Despite these strengths, our analysis reveals that PTv-3 significantly overuses tokens during self-attention. Even with architectural optimizations - like replacing KNN operations and removing image-relative positional encodings - PTv-3 still remains token-heavy. *Strikingly, we find that retaining only 5–10% of the most spatially informative tokens is sufficient to preserve performance across diverse 3D tasks*. This challenges the common belief that dense tokenization is essential (Guo et al., 2021; Zhao et al., 2021; Liu et al., 2019; Lahoud et al., 2022). To our knowledge, this is the first systematic study to expose the redundancy in token usage within 3D point cloud transformers, highlighting new opportunities for improving efficiency without sacrificing accuracy.

We conducted extensive experiments by integrating several token reduction techniques - originally developed for vision transformers - into the PTv-3 framework. These included Token Merging (ToMe) (Bolya et al., 2023; Bolya & Hoffman, 2023; Bonnaerens & Dambre, 2023), Token Pruning (Yin et al., 2022; Zhou et al., 2022; Kim et al., 2024), ALGM (Norouzi et al., 2024), and PiToMe (Tran et al., 2024). By applying before each attention layer and followed by token unmerging, these methods pruned 10–50% of tokens during inference. Across PTv-3 and its variants (e.g., Sonata (Wu et al., 2025) pre-trained on 140k point cloud, Splatformer (Chen et al., 2025)), performance remained largely unaffected even with substantial token reduction on benchmarks like ScanNet (Dai et al., 2017), ScanNet200 (Yeshwanth

[1]German Research Centre for Artificial Intelligence (DFKI) [2]Max Planck Research School for Intelligent Systems (IMPRS-IS) [3]University of Stuttgart [4]College of Engineering and Computer Science, VinUniversity [5]ETH Zurich [6]VinRobotics, Hanoi, Vietnam [7]University of Oldenburg [8]Heinrich Heine University Düsseldorf . Correspondence to: Tuan Anh Tran <tuan.tran@dfki.de>, Paul Swoboda <paul.swoboda@hhu.de>.

Proceedings of the ES-FoMo III: 3rd Workshop on Efficient Systems for Foundation Models at the 42nd International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

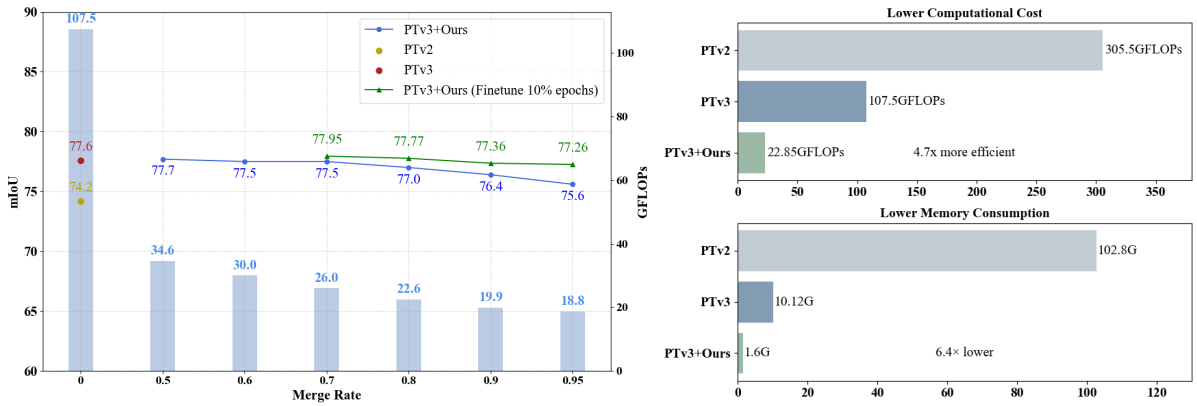

Figure 1. Impact of token merging on performance and efficiency. **Left**: mIoU (%) vs. merge rate (left y-axis) and GFLOPs (right y-axis) between original PTv-3 with our proposed token merging method (PTv-3 + Ours). At each merging rate, we report blue bars as corresponding GFLOPs. Despite this aggressive token reduction, performance degradation is minimal with the off-the-shelf setting (blue line, left plot) and nearly negligible when fine-tuned (green line, right plot) with only 10% total epochs. **Right**: Our approach achieves a **4.7× reduction in FLOPs** (from 107.5 GFLOPs to 22.85 GFLOPs) and a **6.4× reduction in memory usage** (from 10.12 GB to 1.6 GB). PTv2 and PTv3 baselines are shown for reference.

.

et al., 2023), S3DIS (Armeni et al., 2016), nuScenes (Caesar et al., 2020), GSO (Downs et al., 2022a), Objaverse (Deitke et al., 2023), while significantly lowering FLOPs and memory usage. Motivated by this, we developed a 3D-specific token merging strategy that leverages spatial locality and attention relevance (Section 4). Our method merges up to 95% of tokens without retraining, maintaining strong performance even with 95% token removal while saving largely FLOPs and memory usage with large margins (Figure 1. With just 10% fine-tuning, it fully recovers or even surpasses baseline performance, particularly on datasets like ScanNet and S3DIS, underscoring its potential for efficient and scalable deployment. In summary, our contributions are:

- We reveal that up to 90–95% of tokens in point cloud transformers are redundant, challenging the assumption that dense tokenization is necessary.
- We introduce a 3D-specific token merging strategy based on geometric structure and attention saliency, enabling efficient token reduction with minimal accuracy loss.
- We validate our method across segmentation and reconstruction tasks achieving significant efficiency gains and occasionally surpassing baseline performance with limited fine-tuning.

2. Related Work

3D Point Cloud Architectures. 3D point cloud understanding has progressed from projection-based (Chen et al., 2017; Lang et al., 2019; Li et al., 2016; Su et al., 2015) and voxel-based (Maturana & Scherer, 2015; Song et al., 2017; Choy et al., 2019; Graham et al., 2018; Wang et al., 2017) methods to point-based approaches like PointNet (Qi et al., 2017a), PointNet++ (Qi et al., 2017b), and PointMLP (Ma et al., 2022), which operate directly on raw points. While these preserve spatial detail, they often struggle with global context. Transformer-based models have emerged to address this, with the Point Transformer (PTv) series - PTv-1 (Zhao et al., 2021), PTv-2 (Wu et al., 2022), and PTv-3 (Wu et al., 2024) - achieving state-of-the-art results in segmentation (Lai et al., 2022; Wang, 2023; Yang et al., 2025; Lai et al., 2023), and reconstruction (Kong et al., 2024; Chen et al., 2024; Tang et al., 2024; Chen et al., 2025). Recent variants like PTv-3 Sonata (Wu et al., 2025) and Splatformer (Chen et al., 2025) further extend this success.

In this work, we revisit a fundamental question: *"Is PTv-3 efficient in its token usage?"* Interestingly, our results show that it is highly redundant, where retaining only a small fraction of tokens still achieves comparable accuracy, paving the way for lightweight and memory-efficient 3D transformers.

Token Redundancy and Sparsity in Transformers. To improve transformer efficiency, prior work has explored approximating attention via hashing (Daras et al., 2020; Kitaev et al., 2020), low-rank factorization (Likhosherstov et al., 2021; Fan et al., 2021), sparsity (Ren et al., 2021; Shen et al., 2021), head pruning (Meng et al., 2022; Fayyaz et al., 2022), and domain-specific modules (Liu et al., 2021; 2022; Wang et al., 2023b). However, these often require retraining or heavy fine-tuning. In contrast, token reduction methods like pruning (Yin et al., 2022; Zhou et al., 2022; Wang et al., 2024; Kim et al., 2024) and merging (Bolya et al., 2023; Bolya & Hoffman, 2023; Bonnaerens & Dambre, 2023; Tran et al., 2024) aim to speed up inference by reducing token count, typically with minor accuracy loss. Techniques like ToMe (Bolya et al., 2023; Bolya & Hoffman, 2023;

Norouzi et al., 2024) use soft matching to merge similar tokens, while clustering (Bianchi et al., 2020; Marin et al., 2023) and graph-based methods (Wang et al., 2024; Xu et al., 2024; Tran et al., 2024) offer more systematic merging but can introduce extra overhead.

In this work, we adapt several of these methods to PTv-3 and its variants, finding that performance remains robust under substantial token reduction. Building on this, we propose a 3D-specific merging strategy leveraging spatial structure and density, enabling up to 90-95% token reduction with minimal or no loss, offering a path toward scalable and efficient 3D transformers.

3D Point Cloud Compression and Efficiency. Several works aim to design efficient 3D architectures, such as MinkUNet (Choy et al., 2019), Sparse Point Transformer (Tang et al., 2020; Sun et al., 2022; Wang et al., 2023a), PTv3 (Wu et al., 2024), and others (Lai et al., 2022; Feng et al., 2023; Huang et al., 2023), which reduce computation via efficient architectural design. However, they often require training from scratch, limiting compatibility with pre-trained models. Separately, point cloud reduction methods like Random Sampling (Hu et al., 2020; Lu et al., 2020; Zhang et al., 2022a), Farthest Point Sampling (FPS) (Eldar et al., 1997; Qi et al., 2017b; Li et al., 2018; 2022), and VoxelGrid Downsampling (Que et al., 2021; Lyu et al., 2024; Vizzo et al., 2023) reduce input size but are rule-based and task-agnostic, often leading to suboptimal results.

In contrast, our token merging strategy operates at the feature level, enabling dynamic token compression during inference without retraining and integrating seamlessly with existing architectures like PVTv3. It consistently outperforms traditional downsampling methods in both efficiency and accuracy across indoor and outdoor segmentation as well as reconstruction tasks.

3. Analyzing Token Redundancy in 3D Transformers

3.1. Point Transformer v3 architecture

PTv3 introduces a simplified and efficient framework for 3D point cloud processing by replacing KNN-based grouping with a *1D serialization strategy*, where points are ordered via space-filling curves to preserve spatial locality. The model follows a *U-Net-style encoder-decoder architecture* with skip connections, enabling hierarchical feature learning. At each resolution, the serialized sequence is partitioned into disjoint local groups, and *self-attention* is applied independently to each group to capture local context. PTv3 evenly divides the input point set $\mathcal{X} = \{x_1, x_2, \ldots, x_N\}$ into K disjoint subsets (partitions) $\{\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_K\}$ such that $\bigcup_{k=1}^{K} \mathcal{P}_k = \mathcal{X}$, $\mathcal{P}_i \cap \mathcal{P}_j = \varnothing$ for $i \neq j$, and $|\mathcal{P}_k| = 1024$. Self-attention is then applied independently within each partition to capture local geometric structure.

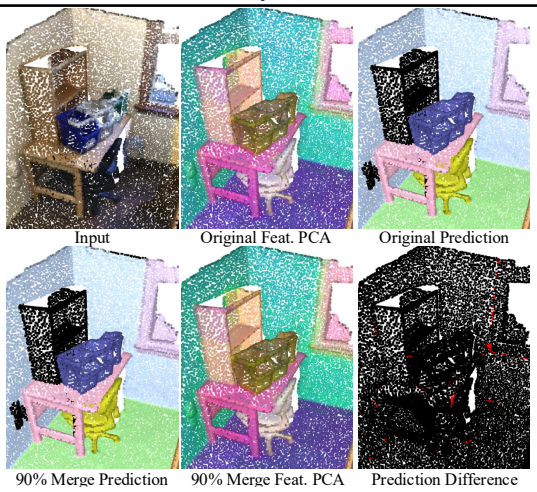

Figure 2. **Observation**: After merging 90% of the tokens in each attention layer, the change in principal component analysis (PCA) visualization of feature representation (2nd image, 2nd row) is minimal compared to the original feature (2nd image, 1st row). Most of the predictions remain unchanged after merging, with red indicating the areas where predictions differ (3rd image, 2nd row). This leads us to conclude that there is high redundancy in the point cloud processing model.

The attention for each point x_i is computed only over the points in its own partition $\mathcal{P}(i)$, formulated as:

$$\mathrm{Attn}(x_i) = \sum_{x_j \in \mathcal{P}(i)} \mathrm{softmax}_j\left(\frac{\mathbf{q}_i^\top \mathbf{k}_j}{\sqrt{d}}\right)\mathbf{v}_j,$$

where $\mathbf{q}_i = \mathbf{q}(x_i)$, $\mathbf{k}_j = \mathbf{k}(x_j)$, and $\mathbf{v}_j = \mathbf{v}(x_j)$ denote the query, key, and value projections of the respective points. The summation is restricted to $x_j \in \mathcal{P}(i)$, ensuring attention is confined to the local context defined by the partition. While this attention mechanism offers strong representational capacity, it becomes prohibitively slow and computationally expensive with $\mathcal{O}(N^2)$ complexity - especially when processing point clouds containing millions of points.

3.2. Token Merging Formulation

To address this limitation, *token merging (Bolya et al., 2023; Tran et al., 2024; Chen et al., 2023; Norouzi et al., 2024)* is introduced to reduce the number of tokens participating in the attention computation. Each original token is mapped to a merged representation via a function $f : x_i \mapsto \widetilde{x}_i$, inducing a transformation of the attention partition $\mathcal{P}(i) \to \widetilde{\mathcal{P}}(i)$, where $|\widetilde{\mathcal{P}}(i)| < |\mathcal{P}(i)|$. Attention is then computed over the merged tokens as:

$$\mathrm{Attn}(f(x_i)) = \sum_{\widetilde{x}_j \in \widetilde{\mathcal{P}}(i)} \mathrm{softmax}_j\left(\frac{f(\mathbf{q}_i)^\top f(\mathbf{k}_j)}{\sqrt{d}}\right) f(\mathbf{v}_j),$$

The function f merges token features according to a learned token-level mapping. Unlike existing token merging meth-

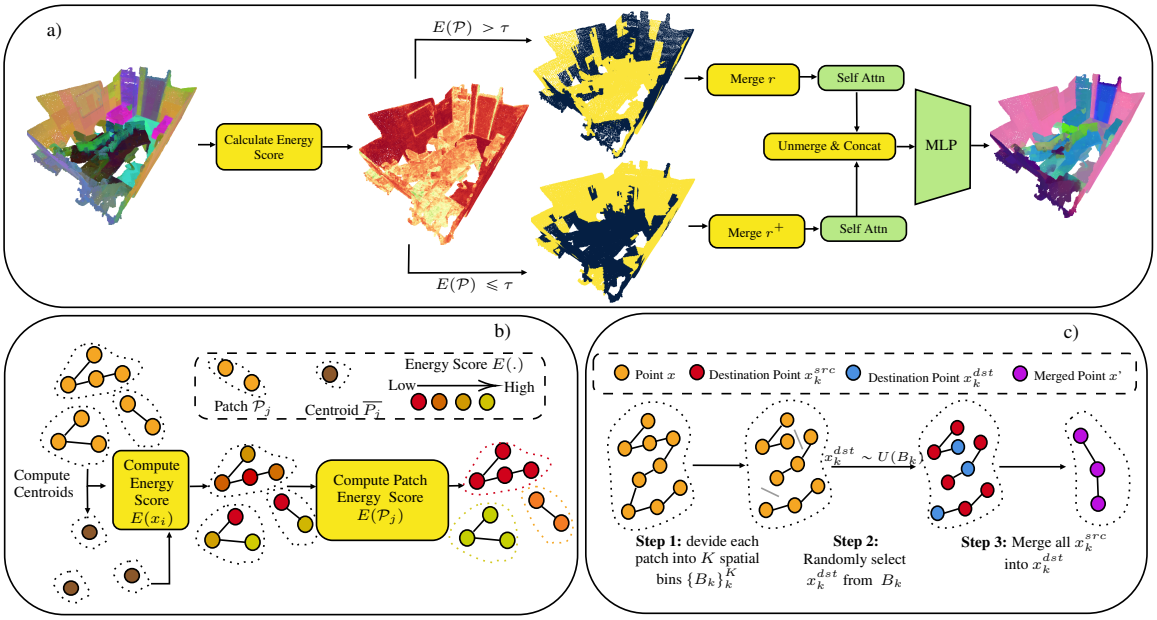

Figure 3. **b)** For each Point Transformer layer, we compute global-informed energy scores, which are later used to calculate patch-level energy scores. **a)**These patch-level scores guide adaptive merging, retaining more information for high-energy patches. **c)** Each patch is divided into evenly sized bins, and destination tokens are randomly selected within these bins to enable spatially aware merging.

ods (Bolya et al., 2023; Tran et al., 2024; Chen et al., 2023; Norouzi et al., 2024), which are designed for classification and operate on the merged tokens throughout subsequent layers, **our approach targets dense 3D point cloud representations**. This requires restoring the token features to their original resolution. To achieve this, an unmerging function f^{-1}, approximating original attention layer is required: $\text{Attn}(x_i) \approx f^{-1}\left(\text{Attn}(f(x_i))\right)$. In our method, f^{-1} simply duplicates the merged attention outputs back to their source tokens.

Token Merging Algorithms. Token Merging (ToMe) (Bolya et al., 2023) defines the merge function $f(\{x_i\}, r)$ by partitioning the set of tokens $\{x_i\}$ into source (src) and destination (dst) sets, and assigning the r most similar tokens from src to tokens in dst. The function f returns merged tokens, where each merged token is obtained by averaging the feature representation of a destination token with the features of its corresponding assigned source tokens.

As shown in Fig. 5, even with 50% token reduction, prediction outputs remain consistent with the original prediction. For instance, the ToMe results at 50% merging, achieving 77.6 mIoU which is similar original Ptv-3, but reducing GFLOPs from 107 down to 76. However, existing methods are primarily designed for image-based tasks and fail to account for 3D-specific characteristics such as spatial locality and density variation. They also typically lack mechanisms for feature recovery, which is essential for dense segmentation. This motivates us to propose a 3D-aware token merging strategy that (i) enables aggressive compression - merging up to 95-99% of tokens and (ii) preserves fine-grained structural details critical for accurate segmentation. We present it in the next section.

4. Adaptive Spatial-Preserving Token Merging

We begin by merging tokens inside equally divided patches inside each Partition $\mathcal{P}_i$. To preserve spatial information, we divide the 1D serialized tokens into evenly sized bins, randomly selecting one token per bin for the destination set. The bin size is determined by the token reduction rate, i.e., $\text{bin\_size} = \lceil 1/(1 - \text{merge\_rate}) \rceil$. Given this partition, we can merge up to 90% of the number of tokens. To better understand this behavior, we analyze the feature representations, as shown in Figure 2. Motivated by these insights and inspired by recent energy-based approaches (Tran et al., 2024; Bolya et al., 2023; Norouzi et al., 2024; Chen et al., 2023), we introduce a globally-informed energy score to guide adaptive, spatially-aware token merging. The overall architecture of our approach is illustrated in Figure 3.

We define a bipartite graph $G = (\mathcal{V}, \mathcal{E})$, where the vertex set is $\mathcal{V} = \{x_i\} \cup \{\bar{P}_j\}$. Here, $\bar{P}_j$ denotes the centroid of partition $\mathcal{P}_j$, computed as: $\bar{P}_j = \frac{1}{|\mathcal{P}_j|} \sum_{x_k \in \mathcal{P}_j} x_k$. The edge set is defined as $\mathcal{E} = \{(x_i, \bar{P}_j)\}$, forming a directed bipartite graph from each token x_i to all partition centroids $\bar{P}_j$. For each token x_i, we define its outgoing neighbors as $\mathcal{N}(x_i) = \{\bar{P}_j \mid (x_i, \bar{P}_j) \in E\}$.

The energy score $E(x_i)$ is then computed as the mean cosine

similarity between x_i and all connected centroids:

$$E(x_i) = -\frac{1}{|\mathcal{N}(x_i)|} \sum_{\bar{P}_j \in \mathcal{N}(x_i)} \cos(x_i, \bar{P}_j). \quad (1)$$

This score reflects how globally aligned a token is with all partition centroids. Tokens with lower energy (i.e., less aligned with global structure) are considered less informative and can be merged more aggressively, while high-energy tokens are preserved to retain critical information.

Adaptive Merging by Energy. Using the above formulation, we define the importance score of a partition $\mathcal{P}$ as the mean energy of its tokens:

$$E(\mathcal{P}) = \frac{1}{|\mathcal{P}|} \sum_{x \in \mathcal{P}} E(x). \quad (2)$$

If $E(\mathcal{P}) > \tau$, we apply moderate merging $f(\mathcal{P}, r)$; otherwise, we apply aggressive merging $f(\mathcal{P}, r^+)$, where $r^+ \gg r$. This branching mechanism enables us to aggressively reduce redundancy and significantly improve computational efficiency, while still supporting batch training and preserving performance on off-the-shelf evaluation. Here τ is a common threshold we effectively used for all datasets and tasks.

5. Experimental Results

5.1. Experiment Setup

We evaluate our method on two 3D tasks: **Indoor Semantic Segmentation** and **3D Reconstruction**. For the Indoor Semantic reconstruction task, we tested our method on Sonata (Wu et al., 2025) and PTv3 (Wu et al., 2024) on 3 different datasets: Scannet200 (Rozenberszki et al., 2022), Scannet (Dai et al., 2017), and S3DIS (Armeni et al., 2016). For the Reconstruction task, we evaluated our method on SplatFormer (Chen et al., 2025) on three different datasets: ShapeNet (Chang et al., 2015), ObjectVerse (Deitke et al., 2023), and GSO (Downs et al., 2022b). We present our method's performance on the evaluation sets of indoor semantic segmentation datasets. Unless stated otherwise, we use an energy threshold of $t = -0.2$ and merge down to 32 tokens via the lower-energy branch.

In addition to recent token merging methods (Tran et al., 2024; Bolya et al., 2023; Norouzi et al., 2024), we incorporate point cloud downsampling techniques to reduce input complexity. **Random Token Drop** (Hu et al., 2020) randomly discards a subset of points, offering fast but coarse reduction. **Farthest Point Sampling** (FPS) (Dovrat et al., 2019; Xu et al., 2020) selects points that are maximally distant from each other to preserve geometric coverage. **VoxelGrid Downsampling** (Que et al., 2021; Lyu et al., 2024) partitions space into voxels and retains one representative point per voxel, ensuring spatial regularity. Final predictions are upsampled in the last stage.

5.2. 3D Point Cloud Semantic Segmentation

Table 1. We compare our method with a merge rate of 0.8 in two settings - fine-tuned and off-the-shelf - against other segmentation and point cloud downsampling methods applied to PTv3, evaluating performance in terms of mIoU.

Methods	ScanNet Val	ScanNet200 Val	S3DIS Area5
MinkUNet (Choy et al., 2019)	72.2	25.0	65.4
ST (Lai et al., 2022)	74.3	-	72.0
OctFormer (Wang, 2023)	75.7	32.6	-
Swin3D (Yang et al., 2025)	76.4	-	72.5
PTv1 (Zhao et al., 2021)	70.6	27.8	70.4
PTv2 (Wu et al., 2022)	75.4	30.2	71.6
PTv3 (Wu et al., 2024)	77.6	35.2	74.7
- *Random Drop*	70.1	31.1	73.4
- *FPS*	71.2	32.4	70.9
- *VoxelGrid Down.*	72.1	32.2	69.1
- Ours	77.0	34.4	74.3
- Ours	77.4	35.2	72.3
PTv3-Sonata (Wu et al., 2025)	79.1	30.4	72.2
- *Random Drop*	72.2	25.2	68.5
- *FPS*	73.9	26.1	69.0
- *VoxelGrid Down.*	73.9	25.5	68.8
- Ours	77.5	28.8	72.8
- Ours	78.9	30.9	73.5

We evaluate our method on Sonata and PTv3 using GFLOPs and mIoU across multiple datasets (Fig. 5). We observe that without fine-tuning, segmentation quality drops minimally. With just 10% of the original training, efficiency improves significantly. At 80% merging for high-energy and 97% for low-energy branches, performance remains stable. Our method also outperforms traditional downsampling by preserving more latent information (Tab. 1).

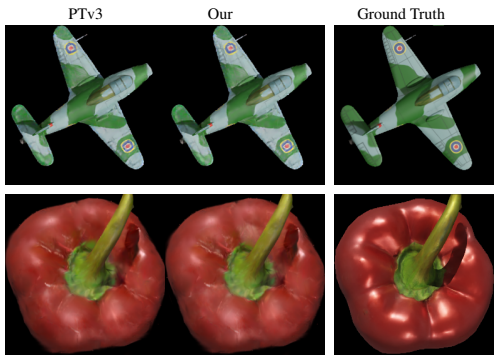

Figure 6. We visualize the output of various token compression techniques after removing 80% of the tokens, comparing their visual quality degradation on the 3D object reconstruction task.

5.3. 3D Object Reconstruction

We also conduct experiments to evaluate the performance of our method on the novel view synthesis task under out-of-distribution (OOD) test camera angles. For this task, we adopt SplatFormer (Chen et al., 2025) as the backbone, which also incorporates PTv3 as its core to refines flawed 3D Gaussian splats to mitigate artifacts in OOD views.

As shown in Table 3, Figure 7 and 6, our method archives high performance - with only about a 0.1% drop across all

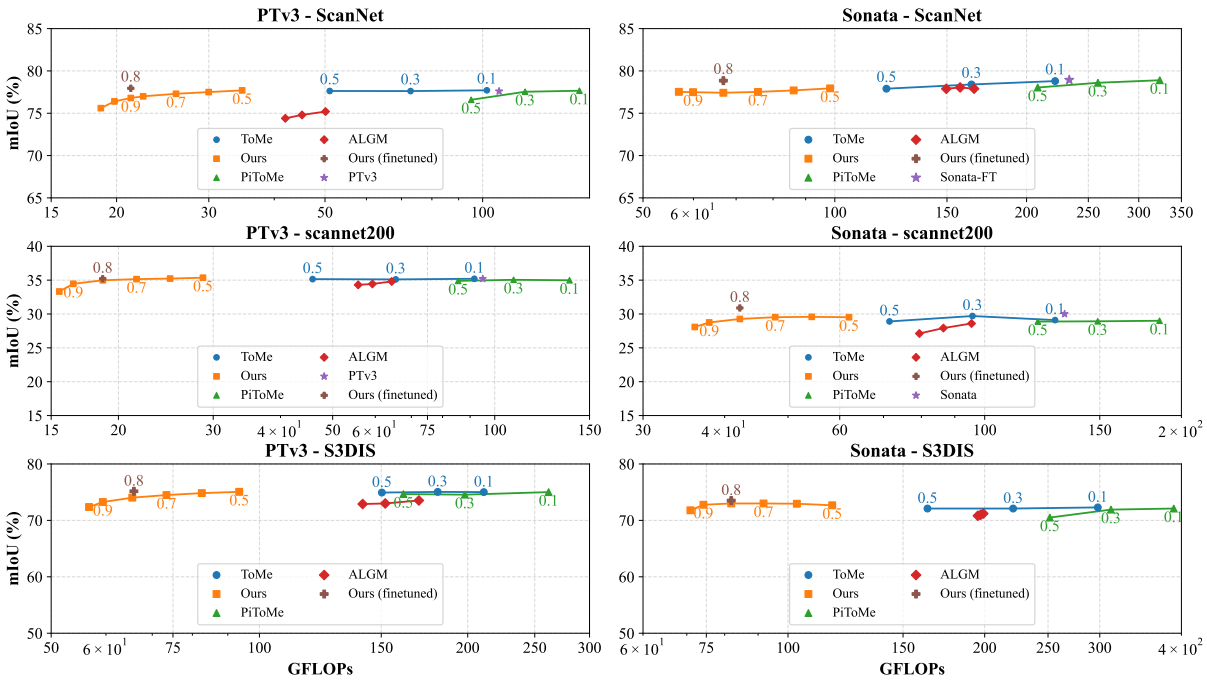

Figure 5. Semantic segmentation comparison of token merging methods. The x-axis shows GFLOPs, the y-axis shows mIoU, and the numbers indicate default merge rates. As shown, our method achieves better computational efficiency while maintaining competitive performance.

Table 2. ScanNet results at different merge ratios without and with adaptive merging.

Method	mIoU↑	mAcc↑	allAcc↑	GFLOPs↓	Peak Mem (GB)↓
PTv3	77.68	84.77	91.82	107.5	10.8
$+r = 0.3$	77.63	84.62	91.91	66.98	6.0
	77.60	84.40	91.79	41.37	4.1
$+r = 0.6$	77.51	84.55	91.79	37.80	2.8
	77.45	83.71	91.48	26.43	2.0
$+r = 0.8$	77.10	84.22	91.81	27.17	2.0
	76.98	83.41	91.34	21.10	1.6

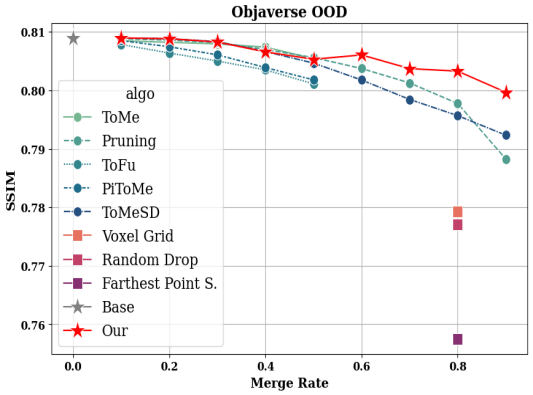

Figure 7. **3D Object reconstruction**: Off-the-shelf performance of our method on Objaverse (Deitke et al., 2023).

metrics - even after reducing up to 90% of the tokens processed by the model, while still outperforming other state-of-the-art methods such as MipNeRF360 (Barron et al., 2022), 3DGS (Kerbl et al., 2023), 2DGS (Huang et al., 2024), Nerfbusters (Warburg et al., 2023). In contrast, alternative token compression techniques such as Random Drop, Voxel Downsampling, and Furthest Point Sampling significantly degrade model performance after reducing 80% number of tokens.

Table 3. **OOD-NVS.** Comparisons on the GSO-OOD and RealWorld-OOD evaluation sets with off-the-shelf evaluation. The metric is evaluated on OOD test views with elevation $\phi_{\text{ood}} \geqslant 70°$.

Methods	GSO-OOD			RealWorld-OOD		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
MipNeRF360 (Barron et al., 2022)	22.90	0.824	0.192	21.99	0.878	0.127
3DGS (Kerbl et al., 2023)	21.78	0.746	0.25	23.83	0.877	0.109
2DGS (Huang et al., 2024)	23.29	0.816	0.204	23.64	0.891	0.104
Nerfbusters (Warburg et al., 2023)	15.95	0.678	0.300	23.93	0.893	0.114
SplatFormer (Chen et al., 2025)	24.71	0.857	0.152	24.33	0.900	0.100
Our	**24.56**	**0.852**	**0.157**	**24.06**	**0.899**	**0.101**

6. Ablation Study

Energy Threshold. We use a threshold τ to decide which patches $\mathcal{P}$ to aggressively merge. As shown in Table 4, when τ is close to 1, most patches are merged aggressively, resulting in unchanged GFLOPS and a 2% drop in MiOU. As τ decreases, fewer patches are merged aggressively, and GFLOPS increase until it reaches the non-adaptive merging baseline. In all experiments, we selected $\tau = -0.2$ as it offers the best trade-off.

Adaptive Merging (Eq.2). Tab.2 demonstrates the perfor-

mance–efficiency trade-off with and without our adaptive merging strategy. Notably, it achieves significant gains in efficiency with minimal performance loss. For instance, at a merge rate of 0.8 (high-energy branch), we observe a 20% reduction in both GFLOPs and peak memory usage, with only a 0.12 mIoU drop.

Table 4. Impact of different thresholds τ on performance.

τ	0.6	0.4	0.2	0.0	-0.2	-0.4	-0.6
GFLOPs↓	19.35	19.35	19.35	20.04	22.68	26.09	27.17
mIoU↑	75.10	75.10	75.10	76.26	76.98	77.09	77.11

7. Conclusion

In conclusion, we show that state-of-the-art 3D point cloud transformers are heavily over-tokenized, and their performance remains largely intact even after reducing 80–95% of tokens using an effective merging strategy. Our 3D-aware method leverages local geometry and attention saliency to estimate voxel importance, enabling aggressive compression with minimal accuracy loss. These findings reveal inefficiencies in current models and offer a practical path toward scalable, efficient 3D vision systems, opening a question - shifting focus from parameter scaling to smarter token utilization.

Acknowledgement

This work was supported by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2075 – 390740016, the DARPA ANSR program under award FA8750-23-2-0004, the DARPA CODORD program under award HR00112590089. The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting Duy M. H. Nguyen. Tuan Anh Tran, Duy M. H. Nguyen, Michael Barz and Daniel Sonntag are also supported by the No-IDLE project (BMBF, 01IW23002), the MASTER project (EU, 101093079), and the Endowed Chair of Applied Artificial Intelligence, Oldenburg University.

References

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Armeni, I., Sener, O., Zamir, A. R., Jiang, H., Brilakis, I., Fischer, M., and 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.

Barron, J. T., Mildenhall, B., Verbin, D., Srinivasan, P. P., and Hedman, P. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5470–5479, 2022.

Bianchi, F. M., Grattarola, D., and Alippi, C. Spectral clustering with graph neural networks for graph pooling. In *International conference on machine learning*, pp. 874–883. PMLR, 2020.

Bolya, D. and Hoffman, J. Token merging for fast stable diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4599–4603, 2023.

Bolya, D., Fu, C.-Y., Dai, X., Zhang, P., Feichtenhofer, C., and Hoffman, J. Token merging: Your vit but faster. *International Conference on Learning Representations (ICLR)*, 2023.

Bonnaerens, M. and Dambre, J. Learned thresholds token merging and pruning for vision transformers. *Transactions on Machine Learning Research*, 2023.

Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., and Beijbom, O. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11621–11631, 2020.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. End-to-end object detection with transformers. In *European conference on computer vision*, pp. 213–229. Springer, 2020.

Chang, A. X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

Chen, A., Xu, H., Esposito, S., Tang, S., and Geiger, A. Lara: Efficient large-baseline radiance fields. In *European Conference on Computer Vision*, pp. 338–355. Springer, 2024.

Chen, M., Shao, W., Xu, P., Lin, M., Zhang, K., Chao, F., Ji, R., Qiao, Y., and Luo, P. Diffrate: Differentiable compression rate for efficient vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 17164–17174, 2023.

Chen, X., Ma, H., Wan, J., Li, B., and Xia, T. Multi-view 3d object detection network for autonomous driving. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 1907–1915, 2017.

Chen, Y., Mihajlovic, M., Chen, X., Wang, Y., Prokudin, S., and Tang, S. Splatformer: Point transformer for robust 3d gaussian splatting. *International Conference on Learning Representations (ICLR)*, 2025.

Choy, C., Gwak, J., and Savarese, S. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3075–3084, 2019.

Dai, A., Chang, A. X., Savva, M., Halber, M., Funkhouser, T., and Nießner, M. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5828–5839, 2017.

Daras, G., Kitaev, N., Odena, A., and Dimakis, A. G. Smyrf-efficient attention using asymmetric clustering. *Advances in Neural Information Processing Systems*, 33: 6476–6489, 2020.

Deitke, M., Schwenk, D., Salvador, J., Weihs, L., Michel, O., VanderBilt, E., Schmidt, L., Ehsani, K., Kembhavi, A., and Farhadi, A. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13142–13153, 2023.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. *International Conference on Learning Representations (ICLR*, 2021.

Dovrat, O., Lang, I., and Avidan, S. Learning to sample. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2760–2769, 2019.

Downs, L., Francis, A., Koenig, N., Kinman, B., Hickman, R., Reymann, K., McHugh, T. B., and Vanhoucke, V. Google scanned objects: A high-quality dataset of 3d scanned household items. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 2553–2560. IEEE Press, 2022a. doi: 10.1109/ICRA46639.2022.9811809. URL https://doi.org/10.1109/ICRA46639.2022.9811809.

Downs, L., Francis, A., Koenig, N., Kinman, B., Hickman, R., Reymann, K., McHugh, T. B., and Vanhoucke, V. Google scanned objects: A high-quality dataset of 3d scanned household items. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 2553–2560. IEEE, 2022b.

Eldar, Y., Lindenbaum, M., Porat, M., and Zeevi, Y. Y. The farthest point strategy for progressive image sampling. *IEEE transactions on image processing*, 6(9):1305–1315, 1997.

Fan, X., Liu, Z., Lian, J., Zhao, W. X., Xie, X., and Wen, J.-R. Lighter and better: low-rank decomposed self-attention networks for next-item recommendation. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pp. 1733–1737, 2021.

Fan, Z., Zhang, J., Cong, W., Wang, P., Li, R., Wen, K., Zhou, S., Kadambi, A., Wang, Z., Xu, D., et al. Large spatial model: End-to-end unposed images to semantic 3d. *Advances in neural information processing systems*, 37:40212–40229, 2024.

Fang, Z., Li, X., Li, X., Buhmann, J. M., Loy, C. C., and Liu, M. Explore in-context learning for 3d point cloud understanding. *Advances in Neural Information Processing Systems*, 36:42382–42395, 2023.

Fayyaz, M., Koohpayegani, S. A., Jafari, F. R., Sengupta, S., Joze, H. R. V., Sommerlade, E., Pirsiavash, H., and Gall, J. Adaptive token sampling for efficient vision transformers. In *European Conference on Computer Vision*, pp. 396–414. Springer, 2022.

Feng, X., Du, H., Fan, H., Duan, Y., and Liu, Y. Seformer: Structure embedding transformer for 3d object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 632–640, 2023.

Graham, B., Engelcke, M., and Van Der Maaten, L. 3d semantic segmentation with submanifold sparse convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9224–9232, 2018.

Guo, M.-H., Cai, J.-X., Liu, Z.-N., Mu, T.-J., Martin, R. R., and Hu, S.-M. Pct: Point cloud transformer. *Computational visual media*, 7:187–199, 2021.

Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L., and Bennamoun, M. Deep learning for 3d point clouds: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 43(12):4338–4364, 2020.

Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., and Markham, A. Randla-net: Efficient semantic segmentation of large-scale point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11108–11117, 2020.

Huang, B., Yu, Z., Chen, A., Geiger, A., and Gao, S. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 conference papers*, pp. 1–11, 2024.

Huang, Z., Zhao, Z., Li, B., and Han, J. Lcpformer: Towards effective 3d point cloud analysis via local context propagation in transformers. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(9):4985–4996, 2023.

Jang, W., Park, M., and Kim, E. Real-time driving scene understanding via efficient 3-d lidar processing. *IEEE Transactions on Instrumentation and Measurement*, 71: 1–14, 2022.

Kerbl, B., Kopanas, G., Leimkühler, T., and Drettakis, G. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023.

Kim, M., Gao, S., Hsu, Y.-C., Shen, Y., and Jin, H. Token fusion: Bridging the gap between token pruning and token merging. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1383–1392, 2024.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4015–4026, 2023.

Kitaev, N., Kaiser, Ł., and Levskaya, A. Reformer: The efficient transformer. *International Conference on Learning Representations (ICLR)*, 2020.

Kong, X., Liu, S., Lyu, X., Taher, M., Qi, X., and Davison, A. J. Eschernet: A generative model for scalable view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9503–9513, 2024.

Lahoud, J., Cao, J., Khan, F. S., Cholakkal, H., Anwer, R. M., Khan, S., and Yang, M.-H. 3d vision with transformers: A survey. *arXiv preprint arXiv:2208.04309*, 2022.

Lai, X., Liu, J., Jiang, L., Wang, L., Zhao, H., Liu, S., Qi, X., and Jia, J. Stratified transformer for 3d point cloud segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8500–8509, 2022.

Lai, X., Chen, Y., Lu, F., Liu, J., and Jia, J. Spherical transformer for lidar-based 3d recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17545–17555, 2023.

Lang, A. H., Vora, S., Caesar, H., Zhou, L., Yang, J., and Beijbom, O. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12697–12705, 2019.

Lee, D. H. and Hong, S. Learning to merge tokens via decoupled embedding for efficient vision transformers. *Advances in Neural Information Processing Systems*, 2024.

Li, B., Zhang, T., and Xia, T. Vehicle detection from 3d lidar using fully convolutional network. *Robotics: Science and Systems*, 2016.

Li, J., Zhou, J., Xiong, Y., Chen, X., and Chakrabarti, C. An adjustable farthest point sampling method for approximately-sorted point cloud data. In *2022 IEEE workshop on signal processing systems (SiPS)*, pp. 1–6. IEEE, 2022.

Li, Y., Bu, R., Sun, M., Wu, W., Di, X., and Chen, B. Pointcnn: Convolution on x-transformed points. *Advances in neural information processing systems*, 31, 2018.

Likhosherstov, V., Choromanski, K. M., Davis, J. Q., Song, X., and Weller, A. Sub-linear memory: How to make performers slim. *Advances in Neural Information Processing Systems*, 34:6707–6719, 2021.

Lin, K., Wang, L., and Liu, Z. Mesh graphormer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 12939–12948, 2021.

Liu, Z., Tang, H., Lin, Y., and Han, S. Point-voxel cnn for efficient 3d deep learning. *Advances in neural information processing systems*, 32, 2019.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022, 2021.

Liu, Z., Ning, J., Cao, Y., Wei, Y., Zhang, Z., Lin, S., and Hu, H. Video swin transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3202–3211, 2022.

Lu, F., Chen, G., Liu, Y., Qu, Z., and Knoll, A. Rskdd-net: Random sample-based keypoint detector and descriptor. *Advances in Neural Information Processing Systems*, 33: 21297–21308, 2020.

Lyu, W., Ke, W., Sheng, H., Ma, X., and Zhang, H. Dynamic downsampling algorithm for 3d point cloud map based on voxel filtering. *Applied Sciences*, 14(8):3160, 2024.

Ma, X., Qin, C., You, H., Ran, H., and Fu, Y. Rethinking network design and local geometry in point cloud: A simple residual mlp framework. *International Conference on Learning Representations (ICLR)*, 2022.

Marin, D., Chang, J.-H. R., Ranjan, A., Prabhu, A., Rastegari, M., and Tuzel, O. Token pooling in vision transformers for image classification. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 12–21, 2023.

Maturana, D. and 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)*, pp. 922–928. IEEE, 2015.

Meng, L., Li, H., Chen, B.-C., Lan, S., Wu, Z., Jiang, Y.-G., and Lim, S.-N. Adavit: Adaptive vision transformers for efficient image recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12309–12318, 2022.

Niepert, M., Minervini, P., and Franceschi, L. Implicit mle: backpropagating through discrete exponential family distributions. *Advances in Neural Information Processing Systems*, 34:14567–14579, 2021.

Norouzi, N., Orlova, S., De Geus, D., and Dubbelman, G. Algm: Adaptive local-then-global token merging for efficient semantic segmentation with plain vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15773–15782, 2024.

Qi, C. R., Su, H., Mo, K., and 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*, pp. 652–660, 2017a.

Qi, C. R., Yi, L., Su, H., and Guibas, L. J. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017b.

Que, Z., Lu, G., and Xu, D. Voxelcontext-net: An octree based framework for point cloud compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6042–6051, 2021.

Ren, H., Dai, H., Dai, Z., Yang, M., Leskovec, J., Schuurmans, D., and Dai, B. Combiner: Full attention transformer with sparse computation cost. *Advances in Neural Information Processing Systems*, 34:22470–22482, 2021.

Rozenberszki, D., Litany, O., and Dai, A. Language-grounded indoor 3d semantic segmentation in the wild. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.

Sander, M. E., Puigcerver, J., Djolonga, J., Peyré, G., and Blondel, M. Fast, differentiable and sparse top-k: a convex analysis perspective. In *International Conference on Machine Learning*, pp. 29919–29936. PMLR, 2023.

Shen, Z., Zhang, M., Zhao, H., Yi, S., and Li, H. Efficient attention: Attention with linear complexities. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 3531–3539, 2021.

Siddiqui, Y., Alliegro, A., Artemov, A., Tommasi, T., Sirigatti, D., Rosov, V., Dai, A., and Nießner, M. Meshgpt: Generating triangle meshes with decoder-only transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 19615–19625, 2024.

Song, S., Yu, F., Zeng, A., Chang, A. X., Savva, M., and Funkhouser, T. Semantic scene completion from a single depth image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1746–1754, 2017.

Su, H., Maji, S., Kalogerakis, E., and Learned-Miller, E. Multi-view convolutional neural networks for 3d shape recognition. In *Proceedings of the IEEE international conference on computer vision*, pp. 945–953, 2015.

Sun, P., Tan, M., Wang, W., Liu, C., Xia, F., Leng, Z., and Anguelov, D. Swformer: Sparse window transformer for 3d object detection in point clouds. In *European Conference on Computer Vision*, pp. 426–442. Springer, 2022.

Tang, H., Liu, Z., Zhao, S., Lin, Y., Lin, J., Wang, H., and Han, S. Searching efficient 3d architectures with sparse point-voxel convolution. In *European conference on computer vision*, pp. 685–702. Springer, 2020.

Tang, J., Chen, X., Wang, J., and Zeng, G. Point scene understanding via disentangled instance mesh reconstruction. In *European conference on computer vision*, pp. 684–701. Springer, 2022.

Tang, J., Chen, Z., Chen, X., Wang, T., Zeng, G., and Liu, Z. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pp. 1–18. Springer, 2024.

Tran, C., MH Nguyen, D., Nguyen, M.-D., Nguyen, T., Le, N., Xie, P., Sonntag, D., Zou, J. Y., Nguyen, B., and Niepert, M. Accelerating transformers with spectrum-preserving token merging. *Advances in Neural Information Processing Systems*, 37:30772–30810, 2024.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Vizzo, I., Guadagnino, T., Mersch, B., Wiesmann, L., Behley, J., and Stachniss, C. Kiss-icp: In defense of

point-to-point icp–simple, accurate, and robust registration if done the right way. *IEEE Robotics and Automation Letters*, 8(2):1029–1036, 2023.

Wang, H., Shi, C., Shi, S., Lei, M., Wang, S., He, D., Schiele, B., and Wang, L. Dsvt: Dynamic sparse voxel transformer with rotated sets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13520–13529, 2023a.

Wang, H., Dedhia, B., and Jha, N. K. Zero-tprune: Zero-shot token pruning through leveraging of the attention graph in pre-trained transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16070–16079, 2024.

Wang, P.-S. Octformer: Octree-based transformers for 3d point clouds. *ACM Transactions on Graphics (TOG)*, 42 (4):1–11, 2023.

Wang, P.-S., Liu, Y., Guo, Y.-X., Sun, C.-Y., and Tong, X. O-cnn: Octree-based convolutional neural networks for 3d shape analysis. *ACM Transactions On Graphics (TOG)*, 36(4):1–11, 2017.

Wang, W., Dai, J., Chen, Z., Huang, Z., Li, Z., Zhu, X., Hu, X., Lu, T., Lu, L., Li, H., et al. Internimage: Exploring large-scale vision foundation models with deformable convolutions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14408–14419, 2023b.

Warburg, F., Weber, E., Tancik, M., Holynski, A., and Kanazawa, A. Nerfbusters: Removing ghostly artifacts from casually captured nerfs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18120–18130, 2023.

Wu, X., Lao, Y., Jiang, L., Liu, X., and Zhao, H. Point transformer v2: Grouped vector attention and partition-based pooling. 35:33330–33342, 2022.

Wu, X., Jiang, L., Wang, P.-S., Liu, Z., Liu, X., Qiao, Y., Ouyang, W., He, T., and Zhao, H. Point transformer v3: Simpler faster stronger. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4840–4851, 2024.

Wu, X., DeTone, D., Frost, D., Shen, T., Xie, C., Yang, N., Engel, J., Newcombe, R., Zhao, H., and Straub, J. Sonata: Self-supervised learning of reliable point representations. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Xie, Y., Dai, H., Chen, M., Dai, B., Zhao, T., Zha, H., Wei, W., and Pfister, T. Differentiable top-k with optimal transport. *Advances in neural information processing systems*, 33:20520–20531, 2020.

Xu, Q., Sun, X., Wu, C.-Y., Wang, P., and Neumann, U. Grid-gcn for fast and scalable point cloud learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5661–5670, 2020.

Xu, X., Wang, S., Chen, Y., Zheng, Y., Wei, Z., and Liu, J. Gtp-vit: efficient vision transformers via graph-based token propagation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 86–95, 2024.

Yang, Y.-Q., Guo, Y.-X., Xiong, J.-Y., Liu, Y., Pan, H., Wang, P.-S., Tong, X., and Guo, B. Swin3d: A pretrained transformer backbone for 3d indoor scene understanding. *Computational Visual Media*, 11(1):83–101, 2025.

Yeshwanth, C., Liu, Y.-C., Nießner, M., and Dai, A. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12–22, 2023.

Yin, H., Vahdat, A., Alvarez, J. M., Mallya, A., Kautz, J., and Molchanov, P. A-vit: Adaptive tokens for efficient vision transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10809–10818, 2022.

Zhang, R., Guo, Z., Gao, P., Fang, R., Zhao, B., Wang, D., Qiao, Y., and Li, H. Point-m2ae: multi-scale masked autoencoders for hierarchical point cloud pre-training. *Advances in neural information processing systems*, 35: 27061–27074, 2022a.

Zhang, R., Guo, Z., Zhang, W., Li, K., Miao, X., Cui, B., Qiao, Y., Gao, P., and Li, H. Pointclip: Point cloud understanding by clip. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8552–8562, 2022b.

Zhao, H., Jiang, L., Jia, J., Torr, P. H., and Koltun, V. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 16259–16268, 2021.

Zhou, Y., Xiang, W., Li, C., Wang, B., Wei, X., Zhang, L., Keuper, M., and Hua, X. Sp-vit: Learning 2d spatial priors for vision transformers. *The British Machine Vision Conference (BMVC)*, 2022.

Supplementary Materials for "How Many Tokens Do 3D Point Cloud Transformer Architectures Really Need?"

A. Limitation Discussion

Despite the promising results demonstrated by our method, several limitations remain that highlight important avenues for future research. One key open question is how to automatically determine the optimal merging rate

r for a given FLOP budget while preserving model performance (Chen et al., 2023; Lee & Hong, 2024). In its current form, our approach relies on a manually specified merging rate and merges only the most informative tokens. Automating this process would require an end-to-end training framework that can learn optimal merging schedules by backpropagating gradients from a FLOP-constrained objective. However, implementing such a system is technically challenging, as it involves non-differentiable operations like sorting and grouping during the merging process. Overcoming this hurdle would likely require gradient approximation techniques for discrete decision-making, such as those explored in (Sander et al., 2023; Xie et al., 2020; Niepert et al., 2021).

In addition, our current study is limited to 3D point cloud data. Extending our token reduction strategy to other 3D data modalities such as meshes (Lin et al., 2021; Siddiqui et al., 2024) and real-time video sensor streams (Jang et al., 2022; Tang et al., 2022) presents a compelling direction for future work. These modalities introduce distinct structural and temporal complexities, yet the fundamental insight underpinning our method, that many tokens in 3D data are redundant, may still hold. Successfully adapting token merging to these formats could yield further efficiency improvements and broaden the practical impact of our approach, particularly in applications such as robotics and autonomous driving where computational resources are limited.

B. Experiments Setup Details

B.1. Semantic Segmentation - Datasets and Metrics:

S3DIS (Armeni et al., 2016) is a large-scale indoor dataset composed of 3D scans from six areas in office buildings. It includes point-wise semantic annotations across 13 categories, making it a common benchmark for semantic segmentation in indoor environments.

ScanNet (Dai et al., 2017) is a richly annotated dataset of indoor scenes, consisting of RGB-D videos that are reconstructed into 3D meshes. It provides point-wise semantic labels over 20 object categories and is widely used for evaluating 3D semantic segmentation models.

NuScenes (Caesar et al., 2020) is an autonomous driving dataset that includes LiDAR point clouds, camera images, and radar data, collected in urban scenes. The 3D semantic segmentation task focuses on labeling LiDAR points across 32 object classes.

ScanNet200 (Yeshwanth et al., 2023) is an extended version of ScanNet with 200 fine-grained object categories. It introduces a more challenging segmentation task due to its larger label space and long-tail class distribution.

Metrics: We evaluate models using several standard metrics. **mIoU** (mean Intersection over Union) measures the average overlap between predicted and ground truth labels across all classes. **mAcc** (mean accuracy) computes the average of per-class accuracies, while **allAcc** (overall accuracy) reflects the proportion of correctly classified points over the entire dataset. In addition to accuracy metrics, we report **FLOPs** (Floating Point Operations) to quantify the computational cost of a model, and **PeakMem** (Peak Memory Usage), which indicates the maximum GPU memory required during inference. These efficiency metrics are critical for understanding model scalability and deployment feasibility.

B.2. Semantic Segmentation - Baselines:

ToMe (Bolya et al., 2023) (Token Merging) is a general framework that reduces token count by merging tokens based on feature similarity, originally proposed for vision transformers. **PiToMe** (Tran et al., 2024) extends ToMe to 3D point cloud processing by introducing point-wise importance scores to guide the merging process. Both ToMe and PiToMe are limited to merging up to 50% of the tokens.

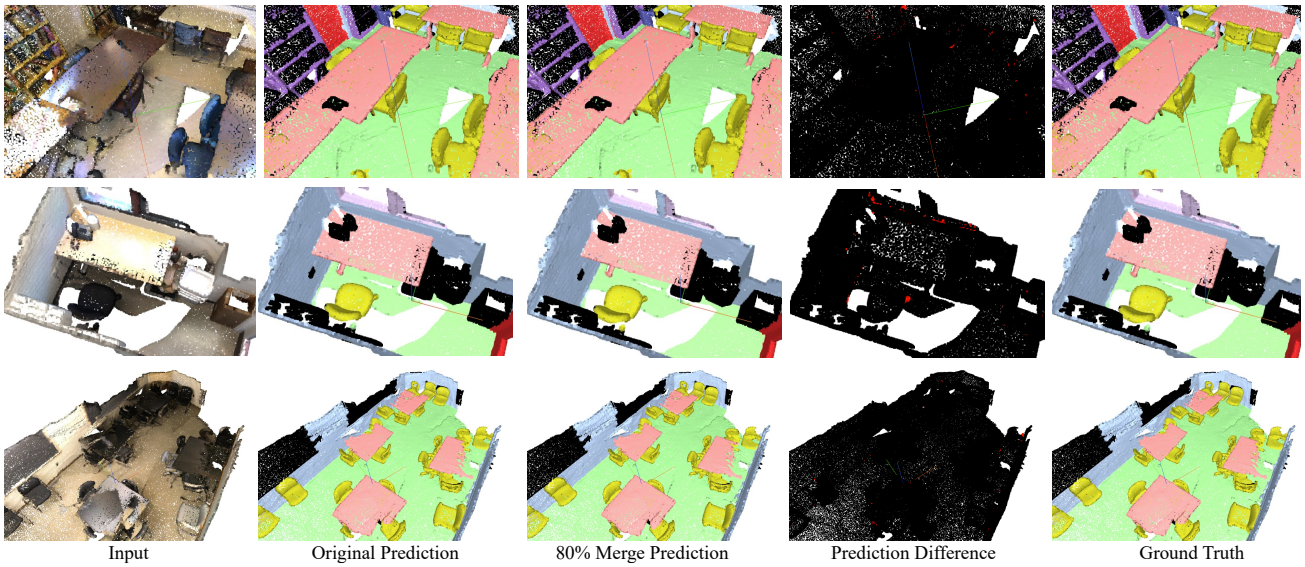

Figure 8. Illustration of ScanNet segmentation results with and without our merging method. As shown in the fourth column, the differences - highlighted in red - are limited to only a few points among hundreds of thousands.

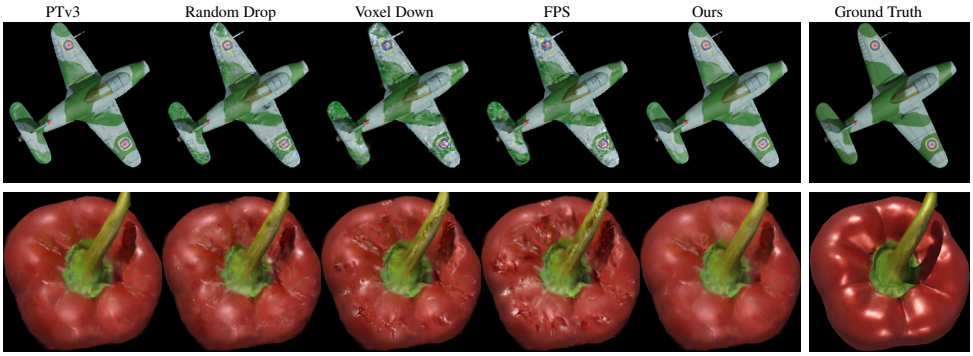

Figure 9. We visualize the output of various token compression techniques after removing 80% of the tokens, comparing their visual quality degradation (or preservation) on the 3D object reconstruction task.

ALGM (Norouzi et al., 2024) is a two-stage token merging approach involving global merging followed by local merging. In our adaptation, we use only the local merging stage, which evenly divides tokens into spatial bins and computes intra-bin similarity. Bins containing highly similar tokens are merged based on a similarity threshold. We evaluate three analytic thresholds for merging: μ, $\mu - \sigma^2$, and $\mu - 2\sigma^2$, where μ is the mean similarity of tokens within a bin and σ^2 is the variance.

Point Cloud Downsampling Techniques: We evaluate several common downsampling strategies for 3D point clouds. **Voxel Downsampling** partitions the 3D space into uniform voxels and retains one representative point per voxel. The feature of each representative point is computed as the mean of the features of all original points within the voxel. **Furthest Point Sampling (FPS)** iteratively selects points such that each newly selected point is as far as possible from previously selected ones, ensuring coverage of the spatial domain. **Random Sampling** simply selects a subset of points uniformly at random from the input set. For all methods, we adjust parameters to ensure that the resulting downsampled point cloud retains approximately 20% of the original points.

C. Detailed Results

C.1. Segmentation

We qualitatively compare our method—merging up to 80% of tokens (high-energy branch)—with the original model in Fig. 8. Our approach effectively removes redundant information while preserving the original predictions.

C.2. 3D Reconstruction

We additionally report detailed results for Objverse-OOD in Tab. 5 and provide qualitative comparisons with baselines in Fig. 9. Despite merging 80% of the tokens, the drop in reconstruction performance is negligible. As shown in Figure 8, our method achieves significantly better visual fidelity compared to existing token compression techniques such as PTv3, Random Drop, Voxel Down, and FPS. While other methods introduce noticeable distortions or surface degradation, our approach preserves fine geometric and textural details, producing reconstructions that closely match the ground truth. This highlights the effectiveness of our token merging strategy in maintaining high-quality 3D reconstruction under extreme compression.

Table 5. **OOD-NVS.** Comparisons on the ShapeNet-OOD and Objaverse-OOD evaluation sets with off-the-shelf evaluation. The metric is evaluated on OOD test views with elevation $\phi_{\text{ood}} \geqslant 70^{\circ}$.

Methods	GSO-OOD			Objverse-OOD			RealWorld-OOD		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR ↑	SSIM↑	LPIPS ↓
MipNeRF360 (Barron et al., 2022)	22.90	0.824	0.192	19.6	0.72	0.28	21.99	0.878	0.127
3DGS (Kerbl et al., 2023)	21.78	0.746	0.25	19.24	0.67	0.29	23.83	0.877	0.109
2DGS (Huang et al., 2024)	23.29	0.816	0.204	19.24	0.67	0.29	23.64	0.891	0.104
Nerfbusters (Warburg et al., 2023)	15.95	0.678	0.300	16.9	0.69	0.29	23.93	0.893	0.114
LaRa (Chen et al., 2024)	-	-	-	19.0	0.68	0.32	-	-	-
SplatFormer (Chen et al., 2025)	24.71	0.857	0.152	22.43	0.808	0.179	24.33	0.900	0.100
- Random Drop	23.77	0.821	0.19	21.80	0.777	0.208	24.02	0.889	0.105
- Farthest Point S.	23.29	0.817	0.194	21.13	0.757	0.223	23.91	0.889	0.107
- VoxelGrid Down.	23.74	0.827	0.18	21.47	0.756	0.224	23.88	0.887	0.108
- Our	**24.56**	**0.852**	**0.157**	**22.34**	**0.803**	**0.185**	**24.06**	**0.899**	**0.101**

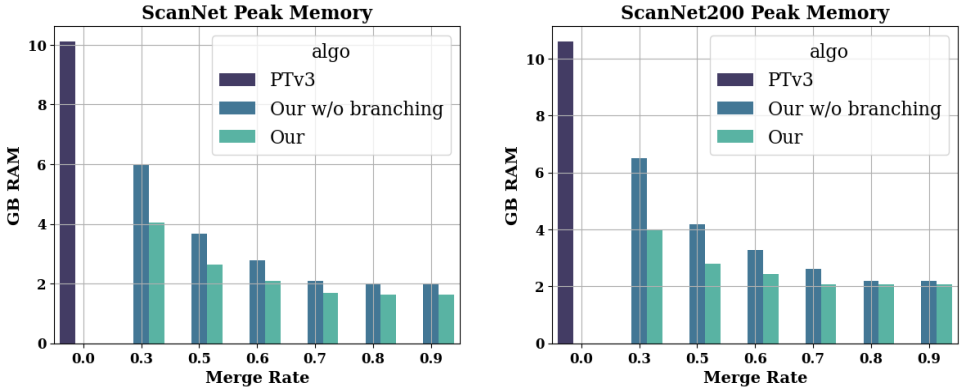

Figure 10. Peak memory

C.3. Further Ablation Study

Merging Metric. In Table 6, we evaluate the effect of using Q, K, or V features as the merging criterion. We also compare applying the merging function independently per head versus uniformly across all heads. Results show that using the value feature (V) and merging independently per head yields the best performance.

Table 6. Impact of metric and independent head during token matching.

Metric	Q	K	V
No Independent Heads	76.08	76.37	76.55
With Independent Heads	76.27	76.36	76.98

Adaptive Merging. We further conduct experiments to verify the impact of the adaptive merging strategy. As shown in Figure 10 and Table 7, adaptive merging effectively identifies which patches should be aggressively merged. This enables the model to achieve lower GFLOPs and reduced peak memory usage (up to 1.5 times compare to without adaptive merging) during inference, without compromising accuracy.

Table 7. Details results on the segmentation task

		ScanNet				ScanNet200			
		mIoU	mAcc	allAcc	GFLOPS	mIoU	mAcc	allAcc	GFLOPS
PTv3 (Wu et al., 2024)		77.68	84.77	91.82	107.5	34.57	45.58	82.79	104.99
$+r = 0.3$	w adaptive merge	77.60	84.40	91.79	41.37	35.10	45.03	83.20	36.40
	w/o adaptive merge	77.63	84.62	91.91	66.98	35.09	45.58	83.29	63.89
$+r = 0.5$	w adaptive merge	77.62	83.91	91.57	30.48	34.72	44.37	83.06	27.79
	w/o adaptive merge	77.69	84.59	91.80	45.73	34.76	44.92	83.16	42.65
$+r = 0.6$	w adaptive merge	77.45	83.71	91.48	26.43	34.48	44.07	82.96	24.62
	w/o adaptive merge	77.51	84.55	91.79	37.80	34.52	44.55	83.09	34.72
$+r = 0.7$	w adaptive merge	77.20	83.53	91.39	23.32	34.21	43.74	82.90	22.17
	w/o adaptive merge	77.31	84.60	91.81	31.63	34.29	44.25	83.02	28.54
$+r = 0.8$	w adaptive merge	76.98	83.41	91.34	21.10	34.20	43.70	82.98	20.42
	w/o adaptive merge	77.11	84.22	91.81	27.17	34.24	44.13	83.06	24.09
$+r = 0.9$	w adaptive merge	76.24	83.05	91.36	19.75	34.38	43.64	83.21	19.39
	w/o adaptive merge	76.40	84.17	91.80	27.14	34.54	44.13	83.28	21.44

D. Local vs Global energy score

To justify the motivation for our globally-informed energy score, we conduct a detailed analysis comparing the behavior of locally-informed energy scores used in PiToMe (Tran et al., 2024) and explain why it failed for the 3D Point Cloud models. As demonstrated in Figure 11, most points belonging to the same object exhibit similar features, as indicated by their shared color. This suggests that in the initial and final layers—where each patch's receptive field is still local and covers only a portion of a larger object—individual tokens lack sufficient contextual information. As a result, computing the energy score locally within each patch does not accurately reflect a token's alignment with the global feature space formed by all points in the input point cloud.

To mitigate this limitation, we introduce a globally informed energy score. This involves first computing centroids for each patch, followed by calculating each token's energy score as the average of its alignment with all patch centroids. As illustrated in Figure 12, the globally-informed energy score provides a clearer distinction between foreground and background regions. This enables more effective identification of patches that can be aggressively merged in the initial and final layers. Consequently, tokens representing the foreground are better preserved before entering the middle layers, where the token space is downsampled and each patch has a wider receptive field.

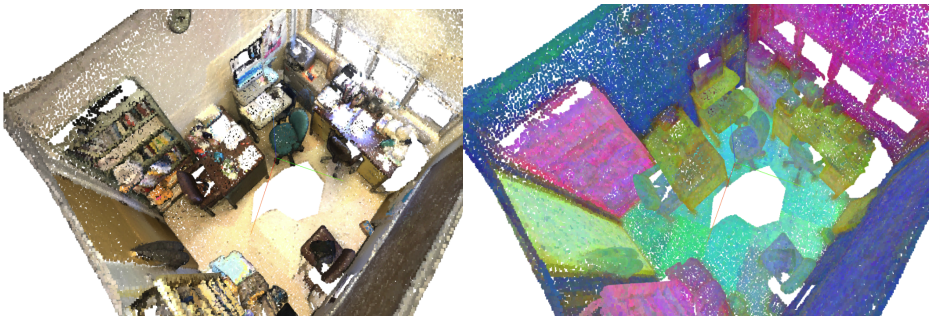

Figure 11. PCA-Based Color Mapping of all tokens in the last layer of PTv3 model.

E. Complexity Analysis

We provide our pseudo code for token merging in Algorithm 1. In our algorithm, the global graph is constructed via matrix multiplication between each point and the patch centroids, resulting in a complexity of $\mathcal{O}(Nkh)$, where h is the dimensionality of the input vectors, N is the number of points, and k is the number of patch centroids (with $k \ll N$). The resulting global energy scores are then used to determine which patches should be aggressively merged.

Let n denote the number of tokens in each aggressively merged patch, and T (where $n \ll T$) be the number of tokens in each patch. The time complexity of the attention operator can be approximated as $\mathcal{O}(k((rT)^2h + n^2h))$ (here r is the ratio of tokens that remain), capturing the dominant contributors to computational cost. However, actual performance may vary with PyTorch version and hardware, due to differences in optimization and parallelization.

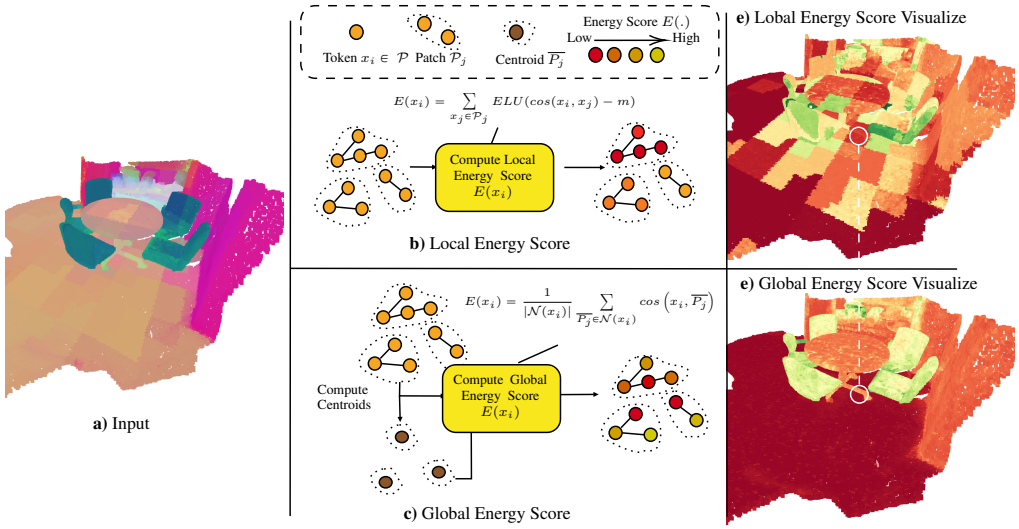

Figure 12. Visualizing Global (Ours) vs. Local (PiToMe(Tran et al., 2024)) Energy Score for each token

Globally Informed Token Merging

Input: Serialized 1D point cloud $\mathcal{P} \in \mathbb{R}^{N \times K \times C}$ (N patches, K points per patch, C features)
Output: Merged point cloud representation

Step 1: Construct Global Bipartite Graph

Compute patch centroids:

$$\bar{P}_j = \frac{1}{|\mathcal{P}_j|} \sum_{x_k \in \mathcal{P}_j} x_k \quad \text{for each patch } \mathcal{P}_j$$

Construct bipartite graph $G = (\mathcal{V}, \mathcal{E})$, where:

- $\mathcal{V} = \{x_i\} \cup \{\bar{P}_j\}$
- $\mathcal{E} = \{(x_i, \bar{P}_j)\}$ — directed edges from points to all patch centroids

Step 2: Compute Energy Scores

Define outgoing neighbors $\mathcal{N}(x_i) = \{\bar{P}_j \mid (x_i, \bar{P}_j) \in \mathcal{E}\}$
Compute point energy:

$$E(x_i) = -\frac{1}{|\mathcal{N}(x_i)|} \sum_{\bar{P}_j \in \mathcal{N}(x_i)} \cos(x_i, \bar{P}_j)$$

Compute patch energy:

$$E(\mathcal{P}_j) = \frac{1}{|\mathcal{P}_j|} \sum_{x \in \mathcal{P}_j} E(x)$$

Step 3: Adaptive Merging by Energy

foreach *patch* $\mathcal{P}_j$ **do**
  if $E(\mathcal{P}_j) > \tau$ **then**
    Apply moderate merging $f(\mathcal{P}_j, r)$
  else
    Apply aggressive merging $f(\mathcal{P}_j, r^+)$
  end
end
return *Merged point cloud*