ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XI-2-2026
XXV ISPRS Congress 2026 “From Imagery to Understanding”, Commission II, 4–11 July 2026, Toronto, Canada

Hierarchical Gaussian Partitioning for Semantic Segmentation of Airborne LiDAR Scenes

Moussa Bendjilali[1,2], Nicola Luminari[1], Pierre Alliez[2]

[1]Alteia, France
[2]Inria Sophia-Antipolis, France

Keywords: Semantic Segmentation, Airborne LiDAR, Point Clouds, Gaussian Mixture Models, Tokenization

Abstract

In this paper, we present a novel approach for semantic segmentation of airborne LiDAR point clouds that integrates a hierarchical Gaussian Mixture Model (hGMM) within the Superpoint Transformer (SPT) framework. The hGMM constructs a coarse-to-fine representation of the scene by recursively fitting Gaussian components to spatially coherent subsets of the point cloud, resulting in a hierarchical and structured decomposition that serves as a structured token set for the segmentation objective. While Gaussian Mixture Models (GMMs) can virtually fit any distribution, we constrain their use to structured suburban scenes, where their parametric form is naturally suited to represent planar and ellipsoidal geometries, hence allowing parsimonious mixtures. Experimental results on the DALES benchmark demonstrate that our method achieves competitive performance with respect to state-of-the-art approaches, with notable improvements on classes such as ground and buildings. Results on indoor S3DIS confirm the method’s intended specificity to outdoor environments. These findings validate hGMM as a principled and effective alternative to heuristic partitioning techniques, integrating stochastic modelling with transformer-based semantic reasoning in large-scale 3D environments.

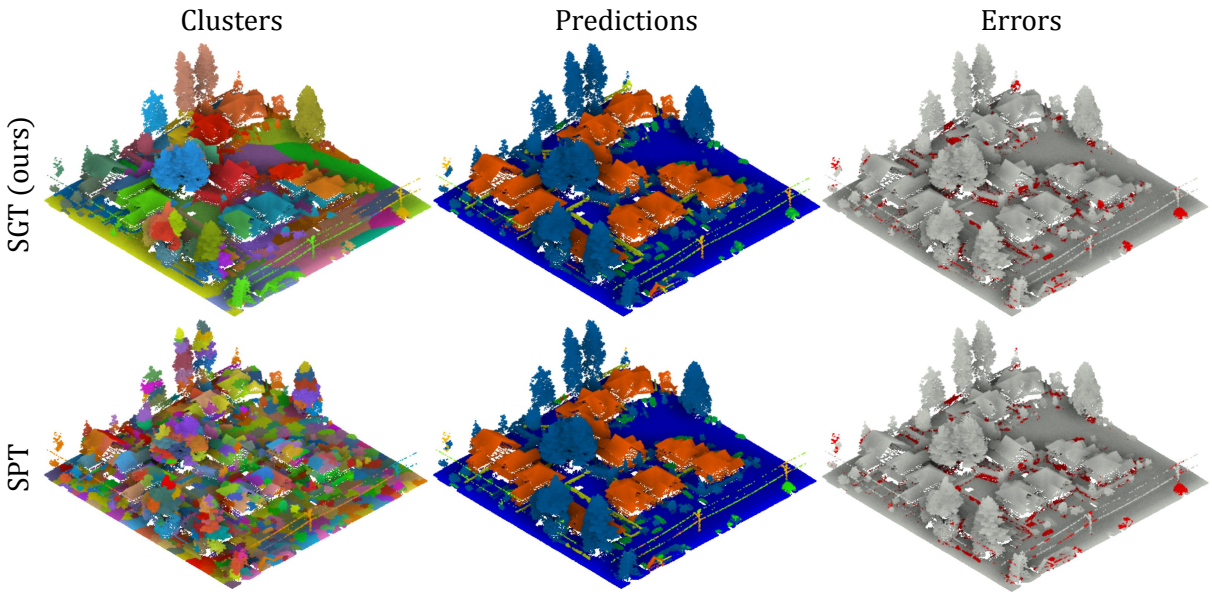

Figure 1. **Qualitative Results. Left:** Clusters color mapping. **Center:** ground, vegetation, car, building, fence, powerline, pole. **Right:** true positives, true negatives. Both Graph CutPursuit (Landrieu and Obozinski, 2017) (bottom) and hGMM (top) are partition-based methods which provide expressive partitioning of outdoor scenes, leading to near state-of-the-art performance (accuracy $\geq 95\%$). Gaussians closely fit planar and ellipsoidal structures such as ground, vegetation, roofs and lines.

1. Introduction

Semantic segmentation of 3D point clouds is a fundamental task in large-scale scene understanding, with applications ranging from urban mapping and autonomous navigation to infrastructure monitoring and environmental modelling (Figure 2). Airborne LiDAR sensors deliver accurate 3D representations over vast expanses, yet the resulting point clouds are often sparse, noisy, and unevenly sampled. Such conditions complicate structured reasoning. While significant progress has been made in adapting techniques from 2D image understanding to the 3D domain (Boulch et al., 2017), these efforts are presently limited by the unordered, irregular nature of point sets. These challenges have motivated the development of novel paradigms and specialized deep learning architectures tailored to the geometry of 3D point clouds, including point-based (Qi et al., 2017), convolution-based (Thomas et al., 2019, Tatarchenko et al., 2018), graph-based (Landrieu and Simonovsky, 2018), and attention-based (Zhao et al., 2021, Guo et al., 2021) frameworks.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-69-2026 | © Author(s) 2026. CC BY 4.0 License.

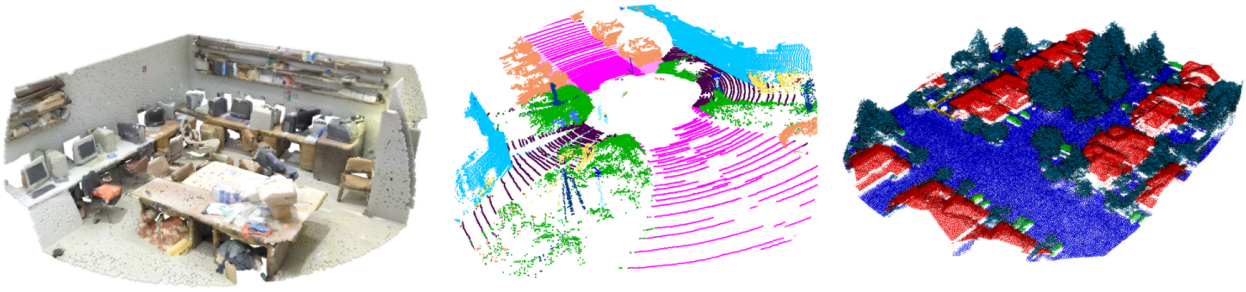

Figure 2. **Domain shift.** 3D point clouds serve a wide variety of applications, ranging from robotics to autonomous driving. However, the significant domain shifts between these modalities present a major obstacle to establishing a unified framework for 3D scene understanding. From left to right, S3DIS (Armeni et al., 2016), Semantic KITTI (Behley et al., 2019), DALES (Varney et al., 2020).

A central challenge in achieving robust scene understanding lies in effectively modelling long-range dependencies. Recent approaches have reintroduced the concept of *tokens* in the context of 3D scenes, which abstract local geometric structures into compact representations suitable for transformer-based architectures. For instance, POINT-BERT (Yu et al., 2021) learns discrete auto-encodings of local patches to perform masked point prediction, while EPCL (Huang et al., 2022) encodes local geometry into CLIP-compatible tokens to facilitate cross-modal alignment. PTV3 (Wu et al., 2024) focuses on simplicity and computational efficiency to enlarge the receptive field of the attention mechanism, replacing k-nearest neighbours calls by a Hilbert curve encoding of the point clouds to achieve state-of-the-art performance. While the application of self-attention to large-scale LiDAR point clouds has led to improved overall performance, computational costs have also grown significantly (Zhao et al., 2021, Guo et al., 2021), which results in long training times and costly GPU usage. Additionally, the extreme topological differences between indoor, automotive and airborne LiDAR scenes, such as density variations, occlusion, sensors biases, class imbalance and diversity, further exacerbate the difficulty to obtain a globally efficient and effective tokenization strategy.

Partition-based methods have been extensively studied for their ability to simplify data analysis in domains with high information redundancy, such as 2D images (Achanta et al., 2012, Tu et al., 2018). By reducing the number of entities to consider while preserving essential information for learning, partitioning serves as an effective tokenization strategy.
Although these methods offer efficient training, they often require computationally expensive preprocessing steps and may suffer from limited expressivity. Hierarchical partitions have demonstrated improved modelling of inter-regional interactions in both 2D (Zhang et al., 2022) and 3D analysis (Liang et al., 2021). SUPERPOINT TRANSFORMER (SPT) (Robert et al., 2023) computes a multi-scale hierarchical structure adapted to the local geometry of the data to produce semantically coherent partitions. These parametric partitions are enough to provide a lightweight semantic segmentation framework enabling fast and near state-of-the-art performances on various benchmarks.

We replace SPT’s original partitioning method based on the approximation of hand-picked and handcrafted features, with a hierarchical Gaussian Mixture Model (hGMM) that produces geometrically expressive Gaussian primitives, hence forming a SUPERGAUSSIAN TRANSFORMER (SGT). Probabilistic representations such as Gaussian Mixture Models (GMMs) have long been employed in 3D processing tasks such as registration (Jian and Vemuri, 2005), generative modelling (Hertz et al., 2020), and hierarchical surface fitting (Eckart et al., 2018) thanks to their ability to approximate geometric primitives and their robustness to noise. However, standard GMMs are often ill-suited for capturing non-parametric or highly curved structures (Lawrence, 2005), and standard Gaussian Expectation-Maximization (GEM) algorithms do not scale well for large-scale scenes (Hirschberger et al., 2022). We propose a hierarchical partition method that addresses these limitations by recursively partitioning the point cloud in a coarse-to-fine manner using parallelized and spatially constrained Expectation-Maximization (EM), yielding an efficient multi-scale decomposition aligned with the underlying geometry. Each Gaussian component acts as a geometrically informative token passed to the segmentation network.

We evaluate our method on the DALES dataset (Varney et al., 2020), a large-scale benchmark for airborne LiDAR segmentation, and on the S3DIS dataset (Armeni et al., 2016), an indoor benchmark. Our results on DALES (Varney et al., 2020) demonstrate that SGT matches or outperforms existing baselines, with improvements in classes that benefit from geometric regularity, such as *buildings* and *ground*, while maintaining computational efficiency. These findings provide strong support for geometry-aware tokenization in transformer-based segmentation of large-scale 3D environments.

Contributions. The main contributions of this work are:

- We present a fast and scalable partitioning algorithm based on spatially constrained Hierarchical Gaussian Expectation-Maximization, enabling robust and efficient hierarchical fitting on large-scale scenes.
- We propose the SuperGaussian Transformer (SGT), which replaces the heuristic, handcrafted partitioning of SPT with our hGMM, hence forming a lightweight and geometrically-interpretable architecture tailored for airborne LiDAR segmentation.
- We validate our hGMM-based tokenization on the DALES dataset (Varney et al., 2020), demonstrating that principled domain-aligned approaches achieve competitive performance against baseline methods.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-69-2026 | © Author(s) 2026. CC BY 4.0 License.

2. Related Work

Tokenization. Building upon the intuition of tokenization from natural language processing, recent works have applied *tokens* or super-structures to efficiently represent 3D point clouds for Transformer-based processing. POINT-BERT (Yu et al., 2021) introduced a discrete variational auto-encoder (dVAE) to convert local point patches into learned tokens, enabling BERT-style masked point modelling for pre-training. 3DLST (Lu et al., 2024) extends this paradigm by dynamically generating *learnable super-tokens* via an optimization module, without requiring pre-segmented super-points and achieving efficient segmentation of large-scale aerial and terrestrial LiDAR data. Similarly, EPCL (Huang et al., 2022) uses a learned tokenizer to embed local point cloud patches into token embeddings compatible with a frozen CLIP transformer, bridging 2D-3D modalities. Other approaches, such as POINTCAT (Yang et al., 2023), use a global class token and cross-attention mechanisms to perform 3D classification and segmentation. These methods highlight how semantic and spatial structures can be abstracted as discrete entities to facilitate learning, minimize redundancy, and improve generalization.

Within this landscape, SUPERPOINT TRANSFORMER (SPT) (Robert et al., 2023) builds upon the idea of geometric tokenization by aggregating points into super-points as a preprocessing step. These super-points serve as input tokens for a lightweight transformer that models contextual relationships between regions rather than individual points. While this design enables efficient and expressive learning for 3D semantic segmentation in most environments, it also inherits limitations from the static, heuristic-based nature of the super-point generation step. In this work, we address these limitations by replacing handcrafted super-points with learned Gaussian components, obtained via a differentiable and task-driven partitioning of the input point cloud, effectively producing super-points that closely fit outdoor topology, as shown in Figure 1.

Gaussian distribution. In the context of novel-view synthesis, Gaussian distributions have been shown effective to achieve state-of-the-art representational power while being affordable to optimize (Kerbl et al., 2023). The Gaussian Expectation-Maximization (GEM) algorithm is a classical method for fitting Gaussian Mixture Models (GMMs), widely used in unsupervised learning and point-set registration. GEM alternates between estimating posterior responsibilities (E-step) and updating mixture parameters to maximize log-likelihood (M-step) (Dempster et al., 1977). Variational Bayesian EM (VBEM) extends GEM by placing full posterior distributions over parameters, enabling automatic model complexity control and reducing over-fitting (Attias, 2000). Classification EM (CEM) introduces hard assignments during the E-step to accelerate convergence to the cost of a higher sensitivity to initialization (Celeux and Govaert, 1992).

GMMs have been effectively adapted for 3D point cloud modelling particularly in registration and segmentation tasks due to their probabilistic representation of surfaces and robustness to noisy scans (Jian and Vemuri, 2005). (Eckart et al., 2016) overcomes the typical slowness and lack of generalization of GMM-based statistical registration algorithms by adopting an efficient multi-scale registration using trees of mixtures (Eckart et al., 2016, Eckart et al., 2018). More recently, POINTGMM has used neural networks to learn hierarchical GMMs, improving both registration and generative modelling of 3D shapes (Hertz et al., 2020).

Gaussian mixtures applicability faces limitations : their smooth, elliptical densities are ill-suited for capturing sharp edges, fine structures, or highly curved surfaces commonly found in complex 3D environments (Lawrence, 2005). This trade-off constrains the use of Gaussian primitives to outdoor airborne scenes that can be characterized by simple geometric structures (lines, planes or ellipsoids), sparse sampling density and noise : these are topological properties well approximated by Gaussian distributions. Our objective is therefore to leverage these strong geometric priors to improve upon more generic methods performances over outdoor benchmarks. The S3DIS (Armeni et al., 2016) evaluation serves as a controlled experiment to validate this design choice, demonstrating that methods can be tailored to their intended application domains before pursuing universal applicability.

3. Method

We propose a hierarchical Gaussian Mixture Model (hGMM) to replace handcrafted super-points in the Superpoint Transformer, forming the SUPERGAUSSIAN TRANSFORMER (SGT). By recursively fitting Gaussian components to the point cloud, our method yields a structured and geometry-aware tokenization. This section details the core components of our approach: Gaussian mixture modelling, hierarchical and parallelized partitioning strategy, and integration of Gaussian tokens into the segmentation network.

Figure 3. **hGMM.** Our method takes as input a point cloud and generates an initial segmentation (level 1) by fitting a coarse Gaussian Mixture Model (GMM) over the entire scene. Each resulting cluster is then refined through the estimation of a local GMM, leading to a hierarchical decomposition of the scene into increasingly fine-grained clusters (level 2 and level 3). The value K denotes the total number of Gaussian components at each level across all branches. Some components may remain empty when a region contains insufficient points, which accounts for the non-round progression ($K = 3 \rightarrow K = 18 \rightarrow K = 50$).

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-69-2026 | © Author(s) 2026. CC BY 4.0 License.

3.1 Gaussian Expectation-Maximization

Gaussian Mixture Models (GMMs) approximate a data distribution as a weighted sum of K multivariate Gaussian components. Each component is defined by a mean $\boldsymbol{\mu}_k \in \mathbb{R}^d$, a covariance matrix $\boldsymbol{\Sigma}_k \in \mathbb{R}^{d\times d}$, and a mixing weight π_k, with $\sum_{k=1}^{K} \pi_k = 1$. The probability density at a point $\mathbf{x}_n$ is:

$$p(\mathbf{x}_n) = \sum_{k=1}^{K} \pi_k \, \mathcal{N}(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{1}$$

The standard Gaussian Expectation-Maximization (GEM) algorithm (Dempster et al., 1977) approximates the mixture by alternating between computing soft responsibilities:

$$\gamma_{nk} = \frac{\pi_k \, \mathcal{N}(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \, \mathcal{N}(\mathbf{x}_n \mid \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} \tag{2}$$

and updating parameters to maximize the expected log-likelihood. While GEM offers robust density estimation, its memory cost scales with $\mathcal{O}(NK)$, making it inefficient for large-scale point clouds.

CEM (Classification EM). The Classification EM algorithm (Celeux and Govaert, 1992) simplifies inference by replacing soft assignments with hard labels:

$$z_n = \arg\max_k \, \gamma_{nk} \tag{3}$$

Although CEM has the same numerical complexity as GEM, it tends to converge in fewer iterations. However, its discrete nature increases chances to get stuck in local minima, sensitivity to initialization and may degrade reconstruction quality in complex or noisy regions.

VBEM (Variational Bayesian EM). VBEM (Attias, 2000) adopts a fully Bayesian approach by placing distributions over parameters and optimizing a variational lower bound. It enables automatic selection of the number of components and mitigates over-fitting. Despite its theoretical appeal, VBEM remains computationally intensive and is less suited for time-critical or high-resolution scenarios. Therefore, we chose not to adopt it in an effort to balance partitioning precision with processing speed.

3.2 Parallelized Refinement and Stability

To enable scalable Gaussian mixture modelling on large 3D scenes, we adopt a recursive hierarchical strategy where each parent Gaussian is subdivided and refined independently, as shown in Figure 3. This coarse-to-fine decomposition not only reflects the geometric hierarchy of the scene but also enables full parallelization at each level. Our implementation leverages parallelism, inspired by multi-scale registration schemes in hierarchical GMMs (Eckart et al., 2018), assigning each local refinement task to a thread for efficient batch processing.

Each cluster is initialized using the result of its parent and refined using CEM or GEM depending on the application. The following pseudocode illustrates our recursive partitioning:

Algorithm 1 hGMM with Parallel Refinement

Require: Point cloud X, depth L, components per level $\{K_1, \ldots, K_L\}$, EM variant
Initialize: $\mathcal{C}_0 \leftarrow \text{GMM}(X, K_1)$
for $\ell = 1$ to L **do**
  $\mathcal{C}_\ell \leftarrow \emptyset$
  for all cluster $C \in \mathcal{C}_{\ell-1}$ **in parallel do**
    Fit GMM(C, K_ℓ) using selected EM variant
    Add resulting sub-clusters to $\mathcal{C}_\ell$ if non-empty
  end for
end for
return All components $\{\mathcal{C}_0, \ldots, \mathcal{C}_L\}$

Computational Efficiency. Let N denote the number of points in the input cloud, D the dimensionality, and $\{K_1, \ldots, K_L\}$ the number of Gaussians at each of the L hierarchical levels. The total number of final components is $K = \prod_{\ell=1}^{L} K_\ell$. At each level ℓ, the algorithm fits K_ℓ components within each of the $\prod_{j=1}^{\ell-1} K_j$ clusters. Assuming the number of points is evenly distributed and that EM variants perform a bounded number of iterations, the cost of fitting each sub-GMM at level ℓ is $\mathcal{O}(\frac{N}{\prod_{j=1}^{\ell-1} K_j} \cdot K_\ell \cdot D^2)$. Summing over all levels, the total cost becomes:

$$\mathcal{O}\left(ND^2 \sum_{\ell=1}^{L} K_\ell\right) \tag{4}$$

This hierarchical design scales linearly with the input size and sub-linearly with the total number of Gaussians, as shown in Figure 4. Furthermore, since refinement tasks at each level are independent, the method benefits from parallelization with negligible communication overhead. In fact, if we choose $K_\ell = K^{1/L}$ for all ℓ, the number of components is uniformly distributed across levels, yielding the total complexity:

$$\mathcal{O}\left(ND^2 L \cdot K^{1/L}\right) \tag{5}$$

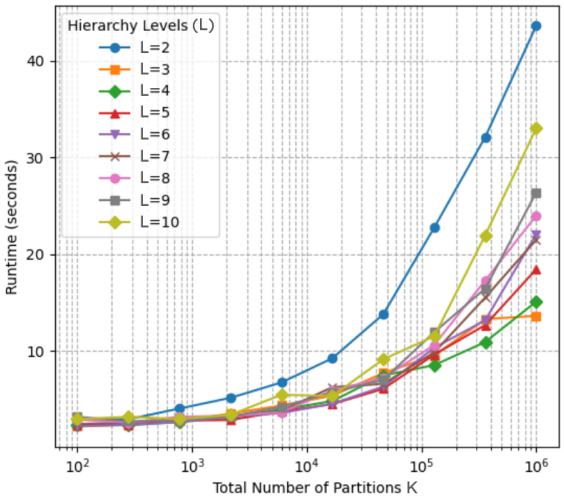

Figure 4. **Runtime analysis of hGMM.** $D = 3$. Each curve corresponds to a fixed depth L, showing how runtime varies with K. Moderate depths ($L = 3$ or 4) yield lower runtimes, balancing and refinement granularity. Smaller depths are also preferable to preserve geometric coherence at coarser levels.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-69-2026 | © Author(s) 2026. CC BY 4.0 License.

Numerical Stability. Maintaining well-conditioned covariance matrices is crucial for EM convergence. Following standard practice in statistical modelling (Jian and Vemuri, 2005), each covariance matrix is symmetrized and regularized as:

$$\boldsymbol{\Sigma}_k \leftarrow \frac{1}{2}\left(\boldsymbol{\Sigma}_k + \boldsymbol{\Sigma}_k^\top\right) + \lambda \mathbf{I} \tag{6}$$

We verify the smallest eigenvalue $\lambda_{\min}$ and ensure positive definiteness by adaptively increasing λ if necessary. Cholesky decomposition is used for efficient and stable evaluation of the log-likelihood and Mahalanobis distance:

$$\delta^2 = (\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \tag{7}$$

This combination of adaptive regularization and numerically stable matrix operations, also adopted in robust point-set alignment methods (Eckart et al., 2018, Jian and Vemuri, 2005), ensures that hGMM performs reliably, even in sparse or noisy scenes.

3.3 Leveraging Gaussians' Expressiveness

The use of Gaussian components in our hierarchical partitioning provides a structured and interpretable representation of the input point cloud. Compared to heuristic partitions such as those produced by Cut Pursuit, the Gaussian representation offers several practical advantages. The mean $\boldsymbol{\mu}_k$ and covariance $\boldsymbol{\Sigma}_k$ jointly encode both position and anisotropy, which can be used to inform downstream attention mechanisms or local descriptors. The associated mixing weights π_k provide a normalized measure of component size, which can be interpreted as a form of spatial salience or confidence.

The resulting representation preserves geometric regularity in structured environments such as suburban or man-made areas. Empirically, we find that components closely fit planar or ellipsoidal structures, which supports semantic segmentation tasks relying on geometric cues. In our implementation, π_k, $\boldsymbol{\mu}_k$, and $\boldsymbol{\Sigma}_k$ are passed as node features to the segmentation network, allowing the model to make use of both semantic and geometric information in a unified framework.

4. Experiments

We assess SGT using two datasets to evaluate both its effectiveness in the target domain and its limitations outside of it. We focus our primary evaluation on the DALES dataset, as our hGMM's geometric priors are explicitly designed for the top-down, sparse, and large-scale planar and ellipsoidal structures characteristic of Airborne LiDAR Scanner (ALS) data. The primary evaluation uses **DALES** (Varney et al., 2020), an outdoor dataset covering a 10km² area captured by aerial LiDAR with 500 million points across 40 urban and rural scenes (12 designated for evaluation). To demonstrate the method's domain specificity, we also evaluate on **S3DIS** (Armeni et al., 2016), an indoor dataset focusing on office buildings with over 274 million points across 6 building floors, where we expect reduced performance due to the incompatibility between Gaussian primitives and complex indoor geometries. This dataset is structured by individual rooms but can also be analysed by examining entire areas at once. We focus our experiments on Area 5 only. Additionally, we apply a hierarchical voxel-based partitioning method to evaluate the effectiveness of intricate partitioning strategies. We conducted training and evaluation of our method on a single-GPU workstation (24 cores, 128Gb RAM, 11 GB NVIDIA RTX 2080 Ti) and compared it to several state-of-the-art methods' official performances.

4.1 Model

We have adopted the same model architecture as the Superpoint Transformer (SPT) (Robert et al., 2023) and made some adaptations tailored to our hierarchical Gaussian-based tokenization. Unlike SPT which relies on geometric and handcrafted features for super-point generation, our hGMM-based partitioning is driven solely by spatial coordinates, requiring no auxiliary features during preprocessing.

During training, each Gaussian component is enriched with its associated parameters (mean vector μ_k, covariance matrix Σ_k, and mixing weight π_k), which are passed to the segmentation network as node features. In addition, we extract four shape descriptors (linearity, planarity, scattering (Demantké et al., 2011), and verticality (Guinard and Landrieu, 2017)) computed per component.

The hierarchical partitioning is defined using fixed numbers of Gaussian components per level with the constraint that the finest components must be comprised of at least 30 points, to be consistent with SPT's calculation of geometric features. Although a balanced configuration would theoretically achieve the lowest preprocessing time for hGMM, we empirically found that increasing the numbers of gaussians in the coarsest levels favored the purity of the initial partitioning. The number of partitions, iterations and hGMM's variants used for the present experiments are all reported in Table 3. This represents a deliberate trade-off between computational efficiency and the semantic relevance of the coarse-level partitioning, as shown in Figure 5.

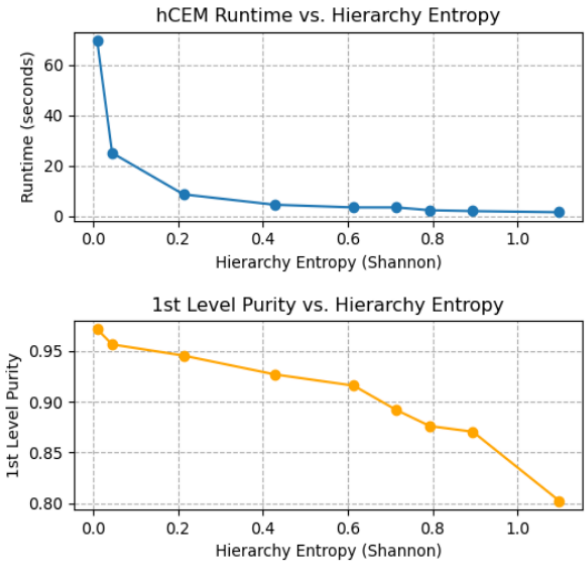

Figure 5. **Tree structure.** Higher entropy values correspond to more balanced tree structures. While these configurations are more efficient, we chose to prioritize geometric coherence at early stages to enhance precision.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-XI-2-2026-69-2026 | © Author(s) 2026. CC BY 4.0 License.

Method	Ground	Vegetation	Car	Truck	Powerline	Fence	Pole	Building	**mIoU**
ConvPoint (Boulch, 2020)	96.9	91.9	75.5	21.7	40.3	29.6	**86.7**	91.9	67.4
PointNet++ (Qi et al., 2017)	94.1	89.1	75.4	30.3	40.0	46.2	79.9	91.2	68.3
PTv3 (Wu et al., 2024)	-	-	-	-	-	-	-	-	77.4
3D-UMamba (Lu et al., 2025a)	-	-	-	-	-	-	-	-	78.1
RandLA (Hu et al., 2020)	97.0	93.2	82.2	38.6	95.0	**73.6**	58.0	96.6	79.3
3DLST (Lu et al., 2025b)	**97.6**	**95.4**	80.2	41.3	83.1	66.7	83.2	94.3	80.2
KPConv (Thomas et al., 2019)	97.1	94.1	85.3	41.9	95.5	63.5	75	96.6	**81.1**
SPT (Robert et al., 2023)	96.7	93.1	**86.1**	**52.4**	94.0	52.7	65.3	96.7	79.6
SPT + Voxel	95.3	91.4	72.1	44.6	**96.0**	49.8	67.8	96.4	76.7
SGT with GEM (ours)	97.0	93.0	83.7	41.8	95.0	50.0	64.1	96.8	77.7
SGT with CEM (ours)	97.2	93.4	85.0	43.0	95.1	52.0	62.7	**97.1**	78.2

Table 1. **Quantitative results on DALES benchmark.** Values show per-class and mean IoU scores. SGT achieves competitive performance (5th overall), excelling on prevalent classes and elongated structures, and shows degradation on rare object categories.

Method	ceiling	floor	wall	column	window	door	chair	table	bookcase	sofa	board	clutter	**mIoU**
SPG (Landrieu and Simonovsky, 2018)	89.4	96.9	78.1	**42.8**	48.9	61.6	84.7	75.4	69.8	52.6	2.1	52.2	58.4
MinkowskiNet (Choy et al., 2019)	91.8	**98.7**	86.2	34.1	48.9	62.4	81.6	**89.8**	47.2	74.9	74.4	58.6	65.4
KPConv (Thomas et al., 2019)	**92.8**	97.3	82.4	23.9	58.0	69.0	**91.0**	81.5	75.3	75.4	66.7	58.9	67.1
Point Trans (Zhao et al., 2021)	94.0	98.5	**86.3**	38.0	63.4	74.3	89.1	82.4	**74.3**	80.2	**76.0**	59.3	70.4
PTv3 (Wu et al., 2024)	-	-	-	-	-	-	-	-	-	-	-	-	**74.7**
DeepViewAgg (Robert et al., 2022)	87.2	97.3	84.3	23.4	**67.6**	72.6	87.8	81.0	**76.4**	54.9	82.4	58.7	67.2
SPT (Robert et al., 2023)	92.6	97.7	83.5	42.0	60.6	67.1	88.8	81.0	73.2	**86.0**	63.1	**60.0**	68.9
SGT with CEM (ours)	91.0	95.8	76.2	29.7	45.3	51.7	75.6	68.3	63.5	41.6	49.3	45.6	56.4
SGT with GEM (ours)	89.7	95.6	76.5	33.1	52.7	53.6	75.1	66.5	63.8	59.6	50.0	46.4	58.7

Table 2. **Quantitative results on S3DIS Area 5 benchmark.** Values show per-class and mean IoU scores. SGT demonstrates reduced performance in these complex indoor environments, only overtaking SPG by a slight margin. We also note that CEM's results are even more degraded than GEM's, as it is theoretically more likely to meet local minima in complex or noisy regions.

4.2 Quantitative Evaluation

Performance Evaluation. Table 1 reports the quantitative results found in publications on the DALES benchmark. The SGT method achieves competitive performance, ranking fifth overall in a tightly packed leaderboard. The SGT model demonstrates particularly strong performance in the prediction of vegetation and ground classes, and it also surpasses the current state-of-the-art for the building class by a margin of +0.4. Furthermore, SGT improves upon the SPT baseline for *powerlines* (+1.1) and *poles* (+2.4), indicating better modelling of elongated and sparse structures. On the other hand, performance drops are observed on less represented categories such as *cars* (-1.1) and *trucks* (-9.6) compared to the Graph CutPursuit (Landrieu and Obozinski, 2017) baseline, suggesting that hGMM may struggle to distinguish fine-grained geometric variations in rare classes.

Dataset	Structure	No. of iterations	Variant
DALES	[96, 32, 12]	20	CEM
S3DIS Area 5	[75, 8, 6]	50	GEM

Table 3. **Configuration.** We report the different structures, number of iterations and variants used to achieve our best results on DALES and S3DIS Area 5.

We also evaluate SGT on the S3DIS Area 5 (Armeni et al., 2016) benchmark to validate our hypothesis about the method's domain limitations (see Table 2). As expected, SGT demonstrates significantly reduced performance compared to outdoor scenes, ranking second-to-last and only outperforming SPG (Landrieu and Simonovsky, 2018). These results confirm our design assumptions: SGT's reliance on Gaussian primitives makes it unsuitable for indoor environments characterized by rapidly changing geometries, irregular surfaces, and complex object arrangements that cannot be effectively approximated by ellipsoidal distributions. This controlled failure validates the method's specificity to structured outdoor environments.

Preprocessing Speed. A core motivation for SGT is to replace SPT's heuristic partitioning with a more principled model, without sacrificing the computational efficiency of the baseline architecture. To ensure a direct and fair comparison, we reproduced the SPT experiment on our own hardware and benchmarked its performance against SGT on the S3DIS Area 5 and DALES datasets. As shown in Table 4, the hGMM partitioning introduces a slight computational overhead during preprocessing compared to SPT's heuristic Cut Pursuit.

Method	Dataset	Preprocessing in min	Training in min	mIoU
Cut Pursuit	S3DIS	73	184	60.2
hGMM (CEM)	S3DIS	105	159	51.8
Cut Pursuit	DALES	148	481	79.6
hGMM (CEM)	DALES	244	384	78.2

Table 4. **Efficiency Comparison on S3DIS Area 5 and DALES.** Preprocessing time and training time are reported in minutes.

4.3 Ablation Study

a) Partitioning. To assess the influence of the partitioning strategy, we compared two variants of the hGMM's partitioning, namely hGMM with GEM and hGMM with CEM, to a naive voxel-based decomposition consistent with SPT's calculation of geometric features (30 points per voxels). As seen in Table 1, voxelization led to a moderate drop in overall mIoU (-2.9), while performance remained competitive and even improved on the *powerline* class. These results suggest that the segmentation backbone is largely responsible for final performance, as also observed in SPT's ablation study. CEM increases

GEM’s results by 0.5 points as it most likely benefits from the simple topology of outdoor scenes, allowing it to achieve lower minima than GEM. The network appears robust to reduced partition quality, possibly due to its strong inductive bias or a tendency to over-fit on simpler datasets. This highlights both the strength and limitations of relying on architectural capacity over structured preprocessing.

b) Tree structure. We bias the hierarchy structure towards the coarsest levels to increase their partition purity, which improves overall results. This approach proves particularly effective on S3DIS Area 5 (Armeni et al., 2016), where objects are more complex and densely packed in the scenes, requiring additional partitions from the beginning, and fewer partitions overall, to reduce lossy tokenization and noisy Gaussian features, as shown in Table 5. For CEM, biasing the structure towards the first level results in a 0.8 point increase in mIoU. For GEM, reducing the total number of partitions by 6% leads to a 2.2 point increase in mIoU.

Variant	Tree structure	Number of clusters	mIoU
CEM	[64, 10, 6]	3840	55.4
	[80, 8, 6]	3840	56.2
GEM	[80, 8, 6]	3840	56.5
	[75, 8, 6]	3600	58.7

Table 5. **Tree structure.** We report the tree structure’s influence on SGT’s performance for S3DIS Area 5.

Limitations. The proposed hGMM relies on a Gaussian geometric prior that is well-suited to the planar and ellipsoidal structures characteristic of airborne outdoor scenes, but degrades for structures with sharp edges or right angles — such as urban furniture or road infrastructure — where smooth ellipsoidal primitives fail to capture fine geometric discontinuities. This is an intended trade-off rather than an oversight, as validated by the controlled degradation observed on S3DIS. Additionally, the purely top-down nature of the hierarchical partitioning may introduce boundary artifacts analogous to those observed in quadtree or octree decompositions. A bottom-up merging step could mitigate such effects by consolidating oversegmented boundary components in a post-processing pass; we leave this as a natural direction for future work, noting that the SPT attention mechanism partially compensates for this limitation by reasoning over neighbouring tokens globally.

5. Conclusion

We presented a fast hierarchical Gaussian mixture model (hGMM) and integrated it to the SuperGaussian Transformer (SGT) to form a lightweight and geometrically-interpretable segmentation framework. By replacing SPT’s heuristic, feature-driven partitioning with geometrically expressive Gaussian tokens, SGT offers a robust, domain-specific alternative for airborne LiDAR 3D semantic segmentation. Results on the DALES benchmark highlight the benefits of geometry-aware tokenization in structured erroneous outdoor scenes, while results on S3DIS confirm the method’s intended domain specificity, demonstrating clear performance degradation in dense indoor environments as expected given our geometric assumptions. Our work bridges statistical modelling and transformer-based reasoning, supporting partition-based methods for efficient segmentation architectures.

References

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S., 2012. SLIC Superpixels Compared to State-of-the-Art Superpixel Methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11), 2274-2282.

Armeni, I., Sener, O., Zamir, A. R., Jiang, H., Brilakis, I., Fischer, M., Savarese, S., 2016. 3d semantic parsing of large-scale indoor spaces. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1534–1543.

Attias, H., 2000. A variational bayesian framework for graphical models. *NeurIPS*, 12.

Behley, J., Garbade, M., Milioto, A., Quenzel, J., Behnke, S., Stachniss, C., Gall, J., 2019. SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences. *Proc. of the IEEE/CVF International Conf. on Computer Vision (ICCV)*.

Boulch, A., 2020. Convpoint: Continuous convolutions for point cloud processing.

Boulch, A., Guerry, J., Le Saux, B., Audebert, N., 2017. SnapNet: 3D point cloud semantic labeling with 2D deep segmentation networks. *Computers and Graphics*, 71, 189-198. https://hal.science/hal-02467932.

Celeux, G., Govaert, G., 1992. A Classification EM Algorithm for Clustering and Two Stochastic Versions. *Computational Statistics & Data Analysis*, 14(3), 315–332.

Choy, C., Gwak, J., Savarese, S., 2019. 4d spatio-temporal convnets: Minkowski convolutional neural networks.

Demantké, J., Mallet, C., David, N., Vallet, B., 2011. Dimensionality based scale selection in 3D LiDAR Point Clouds. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXVIII-5/W12, 97–102. https://isprs-archives.copernicus.org/articles/XXXVIII-5-W12/97/2011/.

Dempster, A. P., Laird, N. M., Rubin, D. B., 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society: series B (methodological)*, 39(1), 1–22.

Eckart, B., Kim, K., Kautz, J., 2018. Fast and Accurate Point Cloud Registration Using Trees of Gaussian Mixtures. *arXiv preprint arXiv:1807.02587*.

Eckart, B., Kim, K., Troccoli, A., Kelly, A., Kautz, J., 2016. Accelerated generative models for 3d point cloud data. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5497–5505.

Guinard, S., Landrieu, L., 2017. Weakly superversied segmentation-aided classification of urban scenes from 3D LiDAR point clouds. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-1/W1, 151–157. https://isprs-archives.copernicus.org/articles/XLII-1-W1/151/2017/.

Guo, M.-H., Cai, Z.-N., Liu, Z.-X., Mu, T.-X., Martin, R. R., Hu, S.-M., 2021. Pct: Point cloud transformer. *Computer Vision–ECCV 2020 Workshops*, Springer, 280–296.

Hertz, A., Hanocka, R., Giryes, R., Cohen-Or, D., 2020. Pointgmm: A neural gmm network for point clouds. *CVPR*.

Hirschberger, F., Forster, D., Lücke, J., 2022. A Variational EM Acceleration for Efficient Clustering at Very Large Scales. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12), 9787-9801.

Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., Markham, A., 2020. Randla-net: Efficient semantic segmentation of large-scale point clouds.

Huang, X., Huang, Z., Li, S., Qu, W., He, T., Hou, Y., Zuo, Y., Ouyang, W., 2022. EPCL: Frozen CLIP Transformer is An Efficient Point Cloud Encoder. *arXiv preprint arXiv:2212.04098*.

Jian, B., Vemuri, B., 2005. A Robust Algorithm for Point Set Registration Using Mixture of Gaussians. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. As summarized in Wikipedia’s point-set registration.

Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G., 2023. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics (SIGGRAPH Conference Proceedings)*, 42(4). http://www-sop.inria.fr/reves/Basilic/2023/KKLD23.

Landrieu, L., Obozinski, G., 2017. Cut Pursuit: fast algorithms to learn piecewise constant functions on general weighted graphs. *SIAM Journal on Imaging Sciences*, Vol. 10(No. 4), pp. 1724-1766. https://hal.science/hal-01306779.

Landrieu, L., Simonovsky, M., 2018. Large-scale point cloud semantic segmentation with superpoint graphs. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4558–4567.

Lawrence, N., 2005. Probabilistic Non-linear Principal Component Analysis with Gaussian Process Latent Variable Models. *J. Mach. Learn. Res.*, 6, 1783–1816.

Liang, Z., Li, Z., Xu, S., Tan, M., Jia, K., 2021. Instance segmentation in 3d scenes using semantic superpoint tree networks.

Lu, D., Xu, L., Zhou, J., Gao, K., Gong, Z., Zhang, D., 2025a. 3D-UMamba: 3D U-Net with state space model for semantic segmentation of multi-source LiDAR point clouds. *International Journal of Applied Earth Observation and Geoinformation*, 136, 104401. https://www.sciencedirect.com/science/article/pii/S1569843225000482.

Lu, D., Xu, L., Zhou, J., Gao, K. Y., Li, J., 2025b. 3DLST: 3D Learnable Supertoken Transformer for LiDAR point cloud scene segmentation. *International Journal of Applied Earth Observation and Geoinformation*, 140, 104572. https://www.sciencedirect.com/science/article/pii/S1569843225002195.

Lu, D., Zhou, J., Gao, K., Xu, L., Li, J., 2024. 3D Learnable Supertoken Transformer for LiDAR Point Cloud Scene Segmentation. *arXiv preprint arXiv:2405.15826*.

Qi, C. R., Yi, L., Su, H., Guibas, L. J., 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in Neural Information Processing Systems*, 30.

Robert, D., Raguet, H., Landrieu, L., 2023. Efficient 3d semantic segmentation with superpoint transformer. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 10852–10861.

Robert, D., Vallet, B., Landrieu, L., 2022. Learning multi-view aggregation in the wild for large-scale 3d semantic segmentation.

Tatarchenko, M., Park, J., Koltun, V., Brox, T., 2018. Tangent convolutions for dense prediction in 3d. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3887–3896.

Thomas, H., Qi, C. R., Deschaud, J.-E., Marcotegui, B., Goulette, F., Guibas, L. J., 2019. Kpconv: Flexible and deformable convolution for point clouds.

Tu, W.-C., Liu, M.-Y., Jampani, V., Sun, D., Chien, S.-Y., Yang, M.-H., Kautz, J., 2018. Learning superpixels with segmentation-aware affinity loss. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 568–576.

Varney, N., Asari, V. K., Graehling, Q., 2020. DALES: A Large-Scale Aerial LiDAR Dataset for Semantic Segmentation. *IEEE Access*, 8, 134488–134504.

Wu, X., Jiang, L., Wang, P.-S., Liu, Z., Liu, X., Qiao, Y., Ouyang, W., He, T., Zhao, H., 2024. Point transformer v3: Simpler, faster, stronger.

Yang, X., Jin, M., He, W., Chen, Q., 2023. PointCAT: Cross-Attention Transformer for Point Cloud. *arXiv preprint arXiv:2304.03012*.

Yu, X., Tang, L., Rao, Y., Huang, T., Zhou, J., Lu, J., 2021. Point-BERT: Pre-Training 3D Point Cloud Transformers with Masked Point Modelling. *arXiv preprint arXiv:2111.14819*.

Zhang, Z., Zhang, H., Zhao, L., Chen, T., Arik, S. Ö., Pfister, T., 2022. Nested hierarchical transformer: Towards accurate, data-efficient and interpretable visual understanding. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36number 3, 3417–3425.

Zhao, H., Jiang, L., Fu, C.-W., Jia, J., Koltun, V., 2021. Point transformer. *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 16259–16268.