Low-Resource White-Box Semantic Segmentation of Supporting Towers on 3D Point Clouds via Signature Shape Identification

Diogo Lavado[*], Cláudia Soares[*], Alessandra Micheletti[†], Giovanni Bocchi[†], Alex Coronati[‡], Manuel Pio[‡], and Patrizio Frosini[§]

Abstract. Research in 3D semantic segmentation has been increasing performance metrics, like the IoU, by scaling model complexity and computational resources, leaving behind researchers and practitioners that (1) cannot access the necessary resources and (2) need transparency on the model decision mechanisms. In this paper, we propose SCENE-Net, a low-resource white-box model for 3D point cloud semantic segmentation of pole-like objects. SCENE-Net identifies signature shapes on the point cloud via group equivariant non-expansive operators (GENEOs), providing intrinsic geometric interpretability. Our model serves as a proof-of-concept that GENEOs can be used to build learning agents that are interpretable, robust to noisy labeling, and resource-efficient. We apply SCENE-Net to the challenging detection of power line supporting towers in power grids—a key task in preventing forest fires and power outages. Our training time on a laptop is 85 min, and our inference time is 20 ms. SCENE-Net has 11 trainable geometrical parameters, like the radius of a ball, and a Precision gain of 38% against a comparable CNN with 2190 parameters. SCENE-Net requires less data to train and shows robustness to data imbalance. With this paper, we release our code implementation: https://github.com/dlavado/scene-net.

Key words. 3D Semantic Segmentation, Point Clouds, Group Equivariant Non-Expansive Operators, White-Box Models, Power Grids

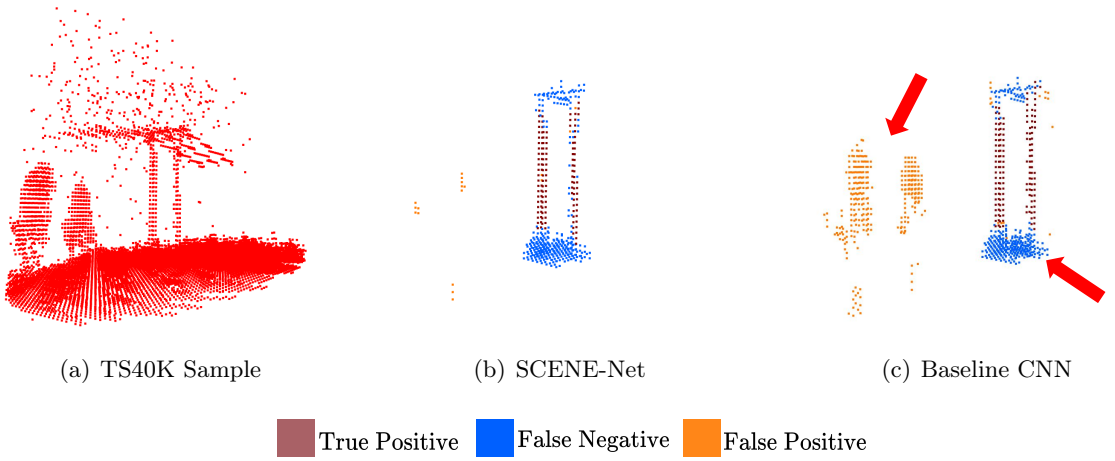
AMS subject classifications. 68T45, 46N10, 58K70

1. Introduction.

Powerful Machine Learning (ML) algorithms applied to critical applications, such as autonomous driving or environmental protection, highlight the importance of (1) ease of implementation for non-tech organization, entailing data efficiency and general-purpose hardware, and (2) transparent models regarding their decision-making process, thus ensuring a responsible deployment [24, 15, 12]. Most methods in Explainable AI (XAI) provide *post hoc* explanations to black-box models (i.e., algorithms unintelligible to humans). However, these are often limited in terms of their model fidelity [24, 33], that is, they provide explanations for the predictions of the underlying model (e.g., heatmaps [7, 42] and input masks [31, 13]), instead of providing a mechanistic understanding of its inner-workings. Conversely, intrinsic interpretability methods (i.e., white-box models) provide an understanding of their decisions through their architecture and parameters [33]. Transparency is achieved by enforcing constraints that reflect domain knowledge and simple designs [24, 8], which can result in a loss in performance when compared to complex black-boxes.

We propose a novel white-box model, **SCENE-Net**, that provides intrinsic geometric interpretability by leveraging on group equivariant non-expansive operators (GENEOs) [3, 6]. Unlike traditional interpretable models, GENEOs are complex observers parameterized with

[*]NOVA School of Science and Technology, Lisbon (d.lavado@campus.fct.unl.pt, claudia.soares@fct.unl.pt).
[†]University of Milan, Milan (alessandra.micheletti@unimi.it, giovanni.bocchi1@unimi.it).
[‡]EDP NEW, Lisbon (alex.coronati@edp.pt, manuelpio.silva@edp.pt).
[§]University of Bologna, Bologna (patrizio.frosini@unibo.it).



(a) TS40K Sample (b) SCENE-Net (c) Baseline CNN

Figure 1. *Signature shapes for power line supporting tower detection. For our TS40K sample shown in (a), SCENE-Net accurately detects the body of the tower (b), while a comparable CNN has a large false positive area in the vegetation (c). Our model is interpretable with 11 trainable geometric parameters whereas the CNN has a total of 2190 parameters. The ground and power lines are mislabeled in the ground truth*

meaningful geometric features. In our case, task dependency comes as a collaboration of Machine Learning and electrical utility teams to transparently segment power line supporting towers on 3D point clouds to inspect extensive power grids automatically. Electrical grid operators have the critical job of assessing the risk of contact between the power grid and its environment to prevent failures and forest fires. These grids spread over countries and even continents, thus making careful inspection an important and challenging problem. Often, this task is based on LiDAR large-scale point clouds with high-point density, no sparsity, and no object occlusion. However, the captured point clouds are quite extensive and mostly composed of rural areas. These data are different from large urban datasets for autonomous driving [18, 2] due to the point of view, point density, occlusion, and extension. To bootstrap this work, we created a labeled dataset of 40 000 Km of rural and forest terrain, and the **T**ransmission **S**ystem, named **TS40K**. These point clouds show noisy labels and class imbalance (see Appendix SM1 for details), and SCENE-Net is robust to labeling noise as it encodes the geometric properties we need to detect.

Moreover, practitioners in high-risk tasks, such as autonomous driving and power grid inspection, are often limited in terms of resources, namely computational power and available data, to train and deploy state-of-the-art models [26, 1]. This clashes with the current trend in DL to scale up models in both complexity and needed resources in order to maximize performance, for example, state-of-the-art 3D semantic segmentation models [41, 37, 45, 47, 48] follow this trend. Our model, SCENE-Net, maintains a simple design conventional to white-boxes (it is composed of 11 trainable parameters) that allows for resource-efficient training while taking advantage of powerful Deep Learning (DL) strategies, such as convolutions. By assessing our model on the SemanticKITTI benchmark [2], we show that SCENE-Net achieves performance on-par with state-of-the-art methods in pole segmentation.

Our main contributions are:

- SCENE-Net is the first white-box model for 3D semantic segmentation on large-scale landscapes, including non-urban environments (Section 4);
- The architecture of SCENE-Net has fewer trainable parameters than traditional methods and is resource-efficient in both data and computational requirements (Section 5.7);
- Empirically, SCENE-Net is intrinsically and posthoc interpretable and robust under noisy labels, with au par IoU (Section 5.3);
- We present TS40K, a new 3D point cloud dataset covering 40 000 Km of non-urban terrain, with more than 9000 million 3D points (details in Appendix SM1);

2. Related Work.

2.1. Point Cloud Semantic Segmentation. Processing point clouds is a challenging task due to their unstructured nature and invariance to permutations. Voxel-based strategies endow point clouds with structure in order to apply 3D convolutions [25, 29, 40]. However, memory footprint is too large for high-resolution voxel grids, while low resolution entails information loss. Subsequent methods try to answer these issues by employing sparse convolutions [14, 36] and octree-based CNNs [43, 22]. Point-based models take point clouds directly as input. The work of PointNet [27] and PointNet++ [28] inspired the use of point sub-sampling strategies with feature aggregation techniques to learn local features on each sub-point [20, 46]. Convolution-based methods [21, 23, 44, 41, 50] demonstrate good performance on 3D semantic segmentation benchmarks, such as *SemanticKITTI* [2] and SensatUrban [19]. Following this strategy, recent methods exploit multi-representation fusion, i.e, they combine different mediums (voxel grids, raw point clouds, and projection images) to boost feature retrieval [9, 37, 45, 48] and achieve top performance on the above benchmarks. While voxel-based methods are computationally expensive due to 3D convolutions on high-resolution voxel grids, point-based strategies have to use costly neighbor searching to extract local information. We propose a voxel-based architecture that is time-efficient with high-resolution voxel grids, with shapes of 64^3 and 128^3. Moreover, learning from imbalanced and noisy data is still a challenging task in point cloud segmentation [17], SCENE-Net is interpretable and robust to these conditions.

2.2. Power line segmentation from 3D point clouds.. Power line inspection is generally performed by on-site maintenance personnel and manned helicopters that examine the power grid with portable devices or the naked eye. These methods are costly, inefficient, and demanding for staff. Thus, process automation is crucial for operators. To this end, unmanned aerial vehicles (UAVs) carrying LiDAR sensors are deployed to scan the power grid and capture a 3D point cloud representation of the environment. In the work [11], the authors combine SLAM algorithms with multi-sensor data to patrol the electrical grid with UAVs. This method employs a multi-view-based approach to point cloud segmentation, so the 3D reconstructions from 2D raster maps usually introduce information loss and decrease in accuracy. Alternative methods project point clouds onto the xy-plane in order to cluster them [16] to segment power lines. This approach does not consider ground and irregular terrain and focuses on segmenting incomplete power lines. Other methods take advantage of fine-grained elevation statistics of the original point cloud and xy-plane projections [38]. The proposals above focus on the segmentation of high-voltage power lines and disregard their supporting towers. By incorporating prior knowledge, our proposal segments supporting towers of any voltage. Not

only are these structures also subject to inspections, but they serve as a point of reference for the location of power lines. In addition, by taking into account raw 3D scenes, other scene elements, such as vegetation, can be segmented to assess the risk of contact between the power grid and the environment.

2.3. Explainable Machine Learning. Explainability is a crucial aspect of ML methods in high-stakes tasks such as autonomous driving [24, 15, 12]. Two main approaches have been proposed in the literature: *post hoc* explainability, and intrinsic interpretability. *Post hoc* methods, such as LIME [31], meaningful perturbations [13], anchors [32], and ontologies [30], are applied to trained black-box models and provide instance-based explanations that correlate model predictions to the given input. These methods are model-agnostic, and thus more flexible, but they often lack mechanistic cause-effect relations and have a limited understanding of feature importance [33]. For instance, a dog image and random noise may generate similar importance heatmaps for the same class with the LIME method [33]. Moreover, they introduce computational overhead, which may limit their application in real-world scenarios with complex black-box models, such as in the 3D semantic segmentation task. In contrast, intrinsic interpretability methods provide an understanding of their decisions through their architecture and parameters [33]. Decision trees and linear regression are examples of white-box models. However, transparency is usually achieved by imposing domain constraints and simple designs, which implies limited performance compared to deep neural networks. Recent advances in interpretable techniques, such as concept whitening [8] and interpretable CNNs [49], have shown that interpretability does not have to imply performance loss. However, these methods provide evidential interpretability, that is, they offer intrinsic explanations to model predictions that are still linked to human interpretations and may imply an evidential correlation, but not causation.

We propose a white-box model, SCENE-Net, with intrinsic geometric interpretability that is not subject to human interpretation. SCENE-Net analyzes the input 3D space according to prior knowledge of the geometry of objects of interest, which is encoded in functional observers and whose parameters are fine-tuned during training. These observers encode high-level geometrical concepts. Thus, our predictions exhibit direct mechanistic cause-effect w.r.t. the learned observers. SCENE-Net maintains a simple model design in high-level mathematical operations while taking advantage of DL complex convolutional kernels.

3. Group Equivariant Non-Expansive Operators (GENEOs). GENEOs are the building blocks of a mathematical framework [3] that formally describes machine learning agents as a set of operators acting on the input data. These operators provide a measure of the world, just as CNN kernels learn essential features to, for instance, recognize objects. Such agents can be thought of as observers that analyze data. They transform it into higher-level representations while respecting a set of properties (i.e., a group of transformations). An appropriate observer transforms data in such a way that respects the right group of transformations, that is, it commutes with these transformations. Formally, we say that the observer is *equivariant* with respect to a group of transformations. The framework takes advantage of topological data analysis (TDA) to describe data as topological spaces. Specifically, a set of data X is represented by a topological space Φ with admissible functions $\varphi\colon X \to \mathbb{R}^3$. Φ can be thought of as a set of admissible measurements that we can perform on the measurement

space X. For example, images can be seen as functions assigning RGB values to pixels. This not only provides uniformity to the framework but also allows us to shift our attention from raw data to the space of measurements that characterizes it. Now that the input data is well represented, let us introduce how the framework defines prior knowledge. Data properties are defined through maps from X to X that are Φ-preserving homeomorphisms. That is, the composition of functions in Φ with such homeomorphisms produces functions that still belong to Φ. Therefore, we can define a group G of Φ-preserving homeomorphisms, representing a group of transformations on the input data for which we require equivariance to be respected. In other words, G is the group of properties that we chose to enforce equivariance w.r.t. the geometry in the original data. It is through G that we embed prior knowledge into a GENEO model. Following the previous example, planar translations can define a subgroup of G.

Let us consider the notion of a *perception pair* (Φ, G): it is composed of all admissible measurements Φ and a subgroup of Φ-preserving homeomorphisms G.

Definition 3.1 (Group Equivariant Non-Expansive Operator (GENEO)). *Consider two perception pairs (Φ, G) and (Ψ, H) and a homomorphism $T\colon G \to H$. A map $F\colon \Phi \to \Psi$ is a group equivariant non-expansive operator if it exhibits equivariance:*

$$\forall \varphi \in \Phi, \forall g \in G, F(\varphi \circ g) = F(\varphi) \circ T(g) \tag{3.1}$$

and is non-expansive:

$$\forall \varphi_1, \varphi_2 \in \Phi, \|F(\varphi_1) - F(\varphi_2)\|_\infty \leq \|\varphi_1 - \varphi_2\|_\infty \tag{3.2}$$

Non-expansivity and convexity are essential for the applicability of GENEOs in a machine-learning context. When the spaces Φ and Ψ are compact, non-expansivity guarantees that the space of all GENEOs $\mathcal{F}$ is compact as well. Compactness ensures that any operator can be approximated by a finite set of operators sampled in the same space. Moreover, by assuming that Ψ is convex, [3] proves that $\mathcal{F}$ is also convex. Convexity guarantees that the convex combination of GENEOs is also a GENEO. Therefore, these results prove that any GENEO can be efficiently approximated by a certain number of other GENEOs in the same space.

In addition to drastically reducing the number of parameters in the modeling of the considered problems and making their solution more transparent, we underline that the use of GENEOs makes available various theoretical results that allow us to take advantage of a new mathematical theory of knowledge engineering. We stress that, besides the cited compactness and convexity theorems, algebraic methods concerning the construction of GENEOs are already available [4, 10, 5]

4. SCENE-Net: Signature geometriC Equivariant Non-Expansive operator Network.

In this section, we introduce the overall architecture of **SCENE-Net**. Next, we define the geometrical properties that describe power line supporting towers. Lastly, we detail the loss function used to train the observer.

4.1. Overview.

3D Point clouds are generally denoted as $\mathcal{P} \in \mathbb{R}^{N \times (3+d)}$, where N is the number of points and $3+d$ is the cardinality of spatial coordinates plus any point-wise features, such as colors or normal vectors. The input point cloud is first transformed in accordance with a measurement function $\varphi\colon \mathbb{R}^3 \to \{0, 1\}$ that signals the presence of 3D points in a voxel

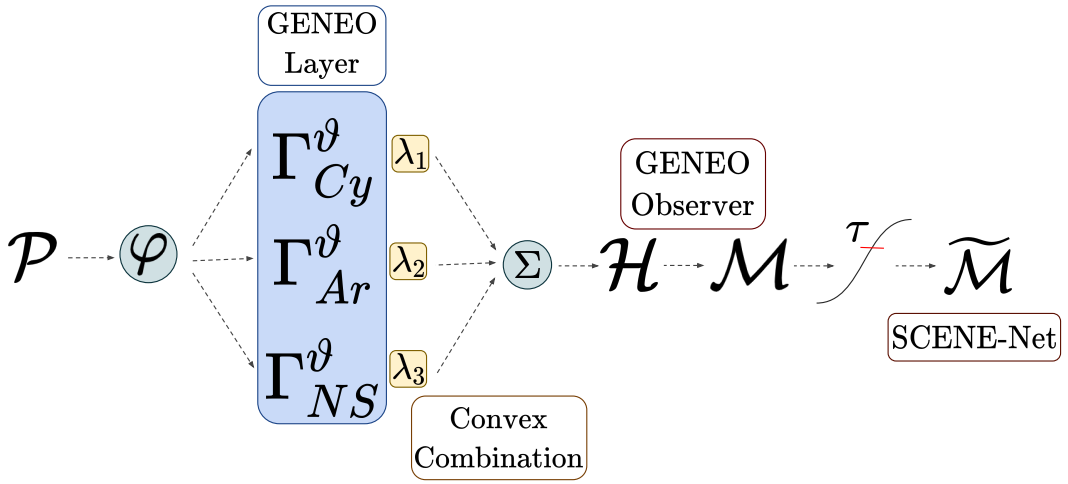


Figure 2. *Pipeline of SCENE-Net: an input point cloud $\mathcal{P}$ is measured according to function φ and voxelized. This representation then is fed to a GENEO-layer, where each operator $\Gamma_i^{\vartheta_i}$ separately convolves the input. A GENEO observer $\mathcal{H}$ is then achieved by a convex combination of the operators in the GENEO layer. $\mathcal{M}$ transforms the analysis of the observer into a probability of belonging to a tower. Lastly, a threshold operation is applied to classify the voxels. Note that this final step occurs after training is completed.*

discretization. Next, the transformed input is fed to a layer of multiple GENEOs (GENEO-layer), each chosen randomly from a parametric family of operators, and defined by a set of trainable shape parameters ϑ (Fig. 2). Such GENEOs are in the form of convolutional operators with carefully designed kernels as described later. Not only is convolution a well-studied operation, but it also offers equivariance w.r.t. translations by definition. During training, it is not the kernels themselves that are fine-tuned with back-propagation, since this would not preserve equivariance at each optimization step. Instead, the error is propagated to the shape parameters ϑ of each operator. Following the GENEO-layer, its set of operators $\Gamma = \{\Gamma_i^{\vartheta_i}\}_{i=1}^K$, with shape parameters $\vartheta = \vartheta_1, \ldots, \vartheta_k$, are combined through convex combination with weights $\lambda = (\lambda_1, \ldots, \lambda_k)^T$ by $\underset{\lambda,\vartheta}{\mathcal{H}} : \mathcal{P} \to \mathcal{P}$ such that

$$\underset{\lambda,\vartheta}{\mathcal{H}}(x) = \sum_{i=1}^{K} \lambda_i \Gamma_i^{\vartheta_i}(\varphi)(x) \tag{4.1}$$

Since the convex combination of GENEOs is also a GENEO [3], $\mathcal{H}$ preserves the equivariance of each operator $\Gamma^{\vartheta} \in \Gamma$. In fact, $\mathcal{H}$ defines a GENEO observer that analyzes the 3D input scenes looking for the geometrical properties encoded in Γ. The convex coefficients λ represent the overall contribution of each operator Γ_i^{ϑ} to $\mathcal{H}$ to the analysis. The parameters grant our model its intrinsic interpretability. They are learned during training and represent geometric properties and the importance of each Γ^{ϑ} in modeling the ground truth.

Next, we transform the observer's analysis into a probability of each 3D voxel belonging to

a supporting tower as a model $\underset{\lambda,\vartheta}{\mathcal{M}}\colon \mathcal{P} \to [0,1]^N$

$$\underset{\lambda,\vartheta}{\mathcal{M}}(x) = \left(\tanh\left(\underset{\lambda,\vartheta}{\mathcal{H}}(x) \right) \right)_+,$$

where $(t)_+ = \max\{0, t\}$ is the rectified linear unit (ReLU). Negative signals in $\mathcal{H}(x)$ represent patterns that do not exhibit the sought-out geometrical properties. Conversely, positive values quantify their presence. Therefore, tanh compresses the observer's value distribution into [-1, 1], and the ReLU is then applied to enforce a zero probability to negative signals. Lastly, a probability threshold $\tau \in [0,1]$ is defined through hyperparameter fine-tuning and applied to $\mathcal{M}$ resulting in a map $\underset{\lambda,\vartheta}{\widetilde{\mathcal{M}}}\colon \mathcal{P} \times \mathbb{R} \to \{0,1\}^N$

$$\underset{\lambda,\vartheta}{\widetilde{\mathcal{M}}}(x,\tau) = \left\{ \underset{\lambda,\vartheta}{\mathcal{M}}(x) \right\} \geq \tau,$$

where $\widetilde{\mathcal{M}}$ denotes the **SCENE-Net** model.

4.2. Knowledge Engineering via GENEOs. In this section, we formally define the knowledge embedded in the observer $\mathcal{H}$. The following GENEOs describe power line supporting towers in order to fully discriminate them from their environment.

4.2.1. Cylinder GENEO. The most striking characteristic of supporting towers against the rural environment is their long, vertical and narrow structure. As such, their identification is equivariant w.r.t. rotations along the *z-axis* and translations in the *xy* plane, which we encode by the means of a cylinder.

Definition 4.1. *In order to promote smooth patterns, a cylinder is defined by* $g_{Cy}\colon \mathbb{R}^3 \to [0,1]$*:*

$$g_{Cy}(x) = e^{-\frac{1}{2\sigma^2}(\|\pi_{-3}(x)-\pi_{-3}(c)\|^2 - r^2)^2}$$

where $\pi_{-3}(x) = (\pi_1(x), \pi_2(x), 0)$ *and* π_i *defines a projection function of the ith element of the input vector.*

Definition 4.1 is a smoothed characterization of a cylinder. The function g_{Cy} defines a smoothed cylinder centered in c by means of a Gaussian function, with the distance between x and the cylinder's radius (r) as its mean. The shape parameters are the Gaussian's standard deviation and r, defined as $\vartheta_{Cy} = [r, \sigma]$.

GENEOs act on functions, transforming them to remain equivariant to a specific group of transformations. Our GENEOs act upon Φ, the topological space representing $\mathcal{P}$ with admissible functions $\varphi\colon \mathbb{R}^3 \to \{0,1\}$. Specifically, we work with appropriate $\varphi \in \Phi$ functions that represent point clouds and preserve their geometry. For instance, φ can be a function that signals the presence of 3D points in a voxel grid. Therefore, the cylinder GENEO Γ^{ϑ}_{Cy} transforms φ into a new function that detects sections in the input point cloud that demonstrate the properties of g_{Cy} and, simultaneously, preserves the geometry of the 3D scene

$$\Gamma^{\vartheta}_{Cy}\colon \Phi \to \Psi, \qquad \psi_{Cy} = \Gamma^{\vartheta}_{Cy}(\varphi)$$

$$\psi_{Cy}(x) = \int_{\mathbb{R}^3} \tilde{g}_{Cy}(y)\varphi(x-y)dy$$

where Ψ is a new topological space that represents $\mathcal{P}$ with functions $\psi\colon \mathbb{R}^3 \to [0,1]$ and $\tilde{g}_{Cy}$ defines a normalized Cylinder. The kernel g_{Cy} is normalized to have a zero-sum to promote the stability of the observer. This way, we encourage the geometrical properties that exhibit the sought-out group of transformations and punish those which do not. Thus, $\psi_{Cy}(x)$ assumes positive values for 3D points near the radius, whereas negative values discourage shapes that do not fall under the g_{Cy} definition. This leads to a more precise detection of the encoded group of transformations. The cylinder kernel discretized in a voxel grid can be seen in Fig. 3(a).

4.2.2. Arrow GENEO. Towers are not the only element in rural environments characterized by a vertical and narrow structure. The identification of trees also shows equivariance w.r.t. rotations along the *z-axis*. Therefore, it is not enough to detect the body of towers, we also require the power lines that they support. To this end, we define a cylinder following the rationale behind the cylinder GENEO with a cone on top of it. This arrow defines equivariance w.r.t. the different angles at which power lines may find their supporting tower.

Definition 4.2. *The function describing the Arrow is defined as* $g_{Ar}\colon \mathbb{R}^3 \to [0,1]$*:*

$$g_{Ar}(x) = \begin{cases} e^{\frac{-1}{2\sigma^2}(\|\pi_{-3}(x)-\pi_{-3}(c)\|^2-r^2)^2} & \text{if } \pi_3(x) < h \\ e^{\frac{-1}{2\sigma^2}(\|\pi_{-3}(x)-\pi_{-3}(c)\|^2-(r_c\tan(\beta\pi))^2)^2} & \text{if } \pi_3(x) \geq h \end{cases}$$

with $\beta \in [0, 0.5)$ *defining the inclination of the cone.*

Definition 4.2 is a smoothed characterization of a cone on top of a cylinder. The radii of the cylinder and cone are defined by r and r_c, respectively, with c as their center. Lastly, h defines the height at which the cone is placed on top of the cylinder. Thus, the shape parameters of the Arrow are defined by the vector $\vartheta_{Ar} = [r, \sigma, h, r_c, \beta]$. Lastly, we are also interested that this kernel sums to zero, so we define

$$\Gamma^{\vartheta}_{Ar}\colon \Phi \to \Psi, \qquad \psi_{Ar} = \Gamma^{\vartheta}_{Ar}(\varphi)$$
$$\psi_{Ar}(x) = \int_{\mathbb{R}^3} \tilde{g}_{Ar}(y)\varphi(x-y)dy,$$

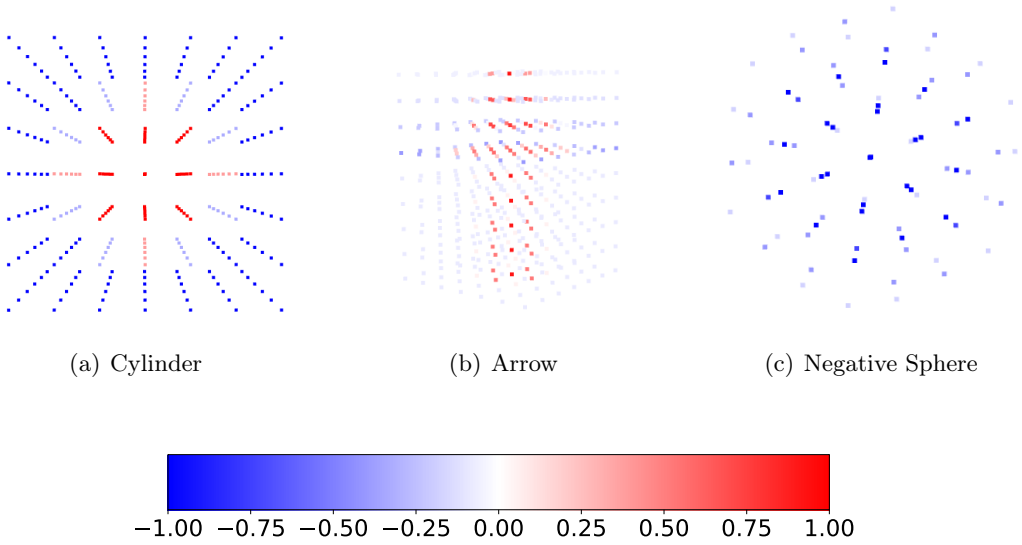
where $\tilde{g}_{Ar}(y)$ represents a normalized Arrow kernel. Its discretization is depicted in Fig. 3(b).

4.2.3. Negative Sphere GENEO. Detecting power lines does not exclude the remaining objects in the scene whose identification also demonstrates equivariance w.r.t. rotations along the *z-axis*. Tree elements, such as bushes, are especially frequent in the TS40K dataset. Thus, we designed a negative sphere to diminish their detection and simultaneously punish the geometry of trees.

Definition 4.3. *The Negative Sphere* $g_{NS}\colon \mathbb{R}^3 \to [-\omega, 1[$ *is defined as*

$$g_{NS}(x) = -\omega e^{\frac{-1}{2\sigma^2}(\|x-c\|^2-r^2)^2}.$$

with $\omega \in]0, 1]$ *defining a small negative weight that punishes the spherical shape.*



(a) Cylinder (b) Arrow (c) Negative Sphere

Figure 3. *GENEO kernels discretized in a voxel grid and colored according to weight distribution*

Definition 4.3 is a smoothed characterization of a geometric sphere. The shape parameters of this operator are $\vartheta_{NS} = [r, \sigma, \omega]$. Since we wish to discourage spherical patterns following the definition of g_{NS}, so we do not enforce that its space sums to zero, obtaining

$$\Gamma^{\vartheta}_{NS} \colon \Phi \to \Psi_{NS}, \qquad \psi_{NS} = \Gamma^{\vartheta}_{NS}(\varphi)$$
$$\psi_{NS}(x) = \int_{\mathbb{R}^3} g_{NS}(y)\varphi(x-y)dy.$$

where Ψ_{NS} is a topological space containing functions $\psi \colon \mathbb{R}^3 \to [-\omega, 1[$. Fig. 3(c) depicts the computation of this kernel in a voxel grid.

4.3. GENEO Loss. The use of GENEOs in knowledge embedding forces our model to uphold the convexity of the observer during training. Thus, our problem statement is represented by the following optimization problem

$$\begin{aligned} \underset{\lambda,\vartheta}{\text{minimize}} \quad & \mathop{\mathbb{E}}_{X,y,\alpha,\epsilon}\left\{\mathcal{L}_{seg}(\lambda,\vartheta)\right\} \\ \text{s.t} \quad & \vartheta \geq 0 \\ & \lambda^T \mathbf{1} = 1 \\ & \lambda \geq 0, \end{aligned}$$

where the segmentation loss $\mathcal{L}_{seg}$ is defined as

$$\mathcal{L}_{seg}(\lambda,\vartheta) = f_w(\alpha,\epsilon,y)\Big(\underset{\lambda,\vartheta}{\mathcal{M}}(X) - y\Big)^2.$$

The loss uses a weighted squared error following the weighting scheme f_w proposed in [35] to mitigate data imbalance. The hyperparameter α emphasizes the weighting scheme, whereas ϵ is a small positive number that ensures positive weights. Thus, $\mathbb{E}\{\cdot\}$ represents the expectation of the segmentation loss over the data distribution. The above constraints ensure that our model $\mathcal{M}$ maintains convexity throughout training, with $\mathbf{1}$ denoting a vector composed of entries one. The reparametrization of the hyperparameters λ allows us to obtain an equivalent optimization problem, considering $\lambda_k = 1 - \sum_{i=1}^{K-1} \lambda_i$, thus obtaining Problem (4.2),

$$\begin{aligned} \underset{\lambda,\vartheta}{\text{minimize}} \quad & \underset{X,y,\alpha,\epsilon}{\mathbb{E}} \Big\{ \mathcal{L}_{seg}(\lambda, \vartheta) \Big\} \\ \text{s.t} \quad & \vartheta \geq 0 \\ & \lambda \geq 0 \end{aligned} , \tag{4.2}$$

with one less constraint. Then, we ensure non-negativity of $\mathcal{M}$'s trainable parameters λ, ϑ by relaxing Problem (4.2) and introducing a penalty in the optimization cost definition as

$$\underset{\lambda,\vartheta}{\text{minimize}} \quad \underset{X,y,\alpha,\epsilon}{\mathbb{E}} \Big\{ \mathcal{L}_{seg}(\lambda, \vartheta) \Big\} + \rho_l \Big(\sum_{i}^{K} h(\lambda_i) \Big) + \rho_t \Big(\sum_{i}^{K} \sum_{j}^{T_i} h(\vartheta_{ij}) \Big), \tag{4.3}$$

where $h(x) = \big(-x\big)_+$, ρ_l and ρ_t are scaling factors of the negativity penalty h, K is the number of GENEO-kernels that composes $\mathcal{M}$ and T_i is the number of shape parameters in ϑ_i. GENEO final loss is formalized in optimization Problem (4.3). It consists of a data fidelity component (i.e., $\mathcal{L}_{seg}$) and two penalties on negative parameters.

5. Experiments.

In this Section, we assess properties of our model SCENE-Net that help electrical companies in the inspection of power lines: (1) TS40K dataset, (2) training protocol, (3) interpretability of the model, (4) accuracy, (5) robustness to noisy labels, (6) training and inference time, and (7) performance on the SemanticKITTI benchmark.

5.1. Dataset.

We evaluate the effectiveness of SCENE-Net on the TS40K dataset. To mitigate the severe data imbalance in our class of interest, we created an ancillary dataset focused on power line-supporting towers with 2823 samples. For each tower in the 3D data, we crop the ground around it with a radius equal to its height. This process introduces bias on classical machine learning agents, such as CNNs. Contrastingly, SCENE-Net is optimized, after incorporating the appropriate prior knowledge, to detect the chosen features that describe ground-truth. A biased training dataset results in an agent tailored to detect supporting towers' geometry. Elements in the 3D scene that do not align with these properties are not signaled by SCENE-Net. Furthermore, we discretize the cropped point clouds with a volume of 64^3 voxels. If a voxel contains any 3D tower points, it is labeled as a tower voxel, otherwise, it is labeled as a non-tower voxel. This emphasizes the geometry of supporting towers so that they can be better described by SCENE-Net. To represent the 3D input, all non-empty voxels are given a value of 1. This measurement function preserves the structure of raw point clouds and mitigates the difference in point density between supporting towers and other classes.

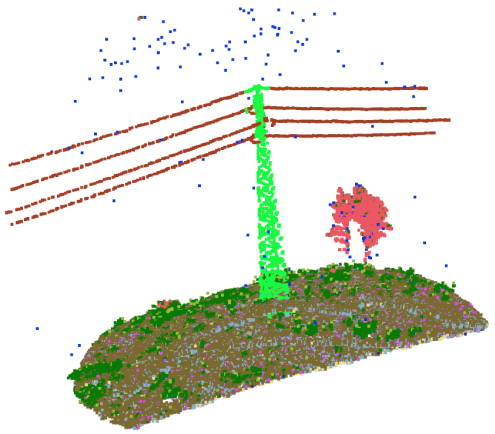


Figure 4. *Crop sample from the TS40K dataset.*

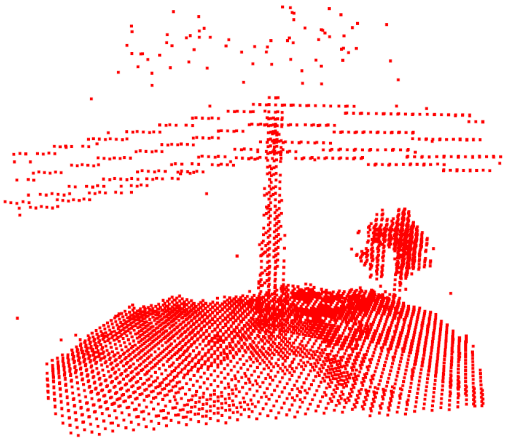


Figure 5. *Voxelization of sample in Figure 4.*

5.2. Training Protocol. During the end-to-end training process of SCENE-Net, we adopt the following settings: batch size is 8 for a total of 50 epochs. We employ the RMSProp optimizer with a learning rate of 0.001. The weighting scheme parameters α and ϵ are set to 5 and 0.1, respectively. While both scaling factors of the non-positive penalty ρ_l and ρ_t are set to 5. The kernel size used to discretize the GENEO operators is 9^3. The GENEOs parameters ϑ are randomly initialized under suitable and positive ranges. While the convex coefficients $\lambda_0, \ldots, \lambda_{n-1}$ are randomly initialized in the range $[0, \frac{2}{N-1}]$ to promote a valid convex space for $\mathcal{H}$. To demonstrate that SCENE-Net achieves good results even with fewer data, we use 20% of the dataset for training, 10% for validation, and 70% for testing. All experiments were conducted on an NVIDIA GeForce RTX 3070 GPU.

5.3. Interpretability of the trained SCENE-Net: The meaning of the 11 learned parameters. To understand if the model parameters are interpretable, we inspect SCENE-Net's 11 trainable parameters ϑ and λ after training. Each $\vartheta_i \in \vartheta$ holds the learned shape parameters of a geometrical operator Γ_i, such as their height or radius. The convex coefficients λ weigh each operator Γ_i in our model's analysis. For example, we can conclude that the instance ϑ_{NS} of the Negative Sphere GENEO (Γ_{NS}) holds a weight of 76.34% on SCENE-Net's output (Fig. 6). The geometric nature of the observer and combination parameters endow intrinsic **interpretability** to SCENE-Net.

5.4. Post-hoc interpretation for specific predictions. We can correlate the detection of scene elements, such as vegetation, to the contributions of each GENEO. This provides an extra layer of transparency to our model. The Arrow kernel is responsible for the detection of towers, the Cylinder aids this process and diminishes the detection of vegetation, and the Negative Sphere stabilizes the model by balancing contributions of the previous kernels (Fig 7).

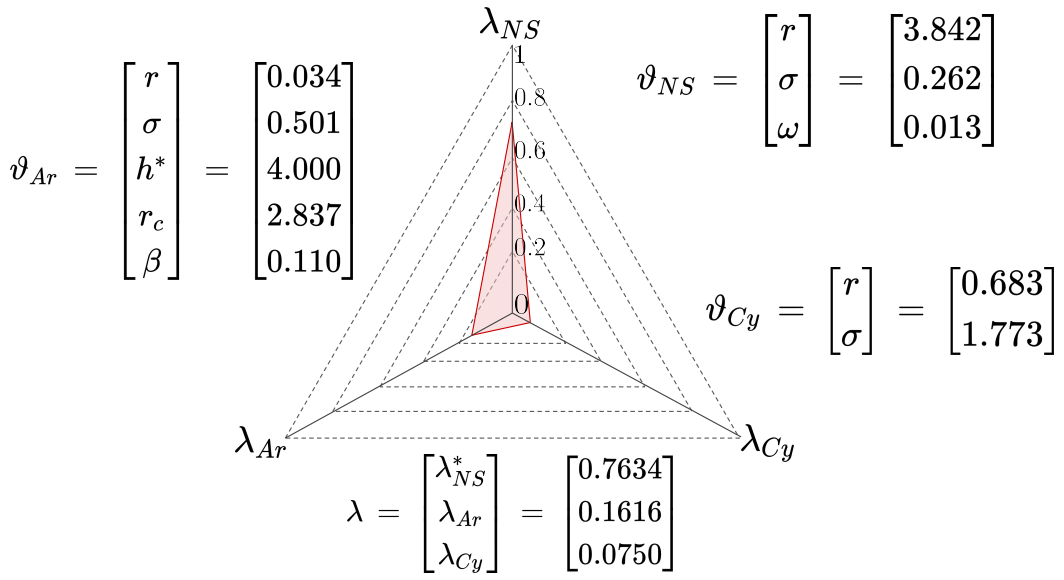


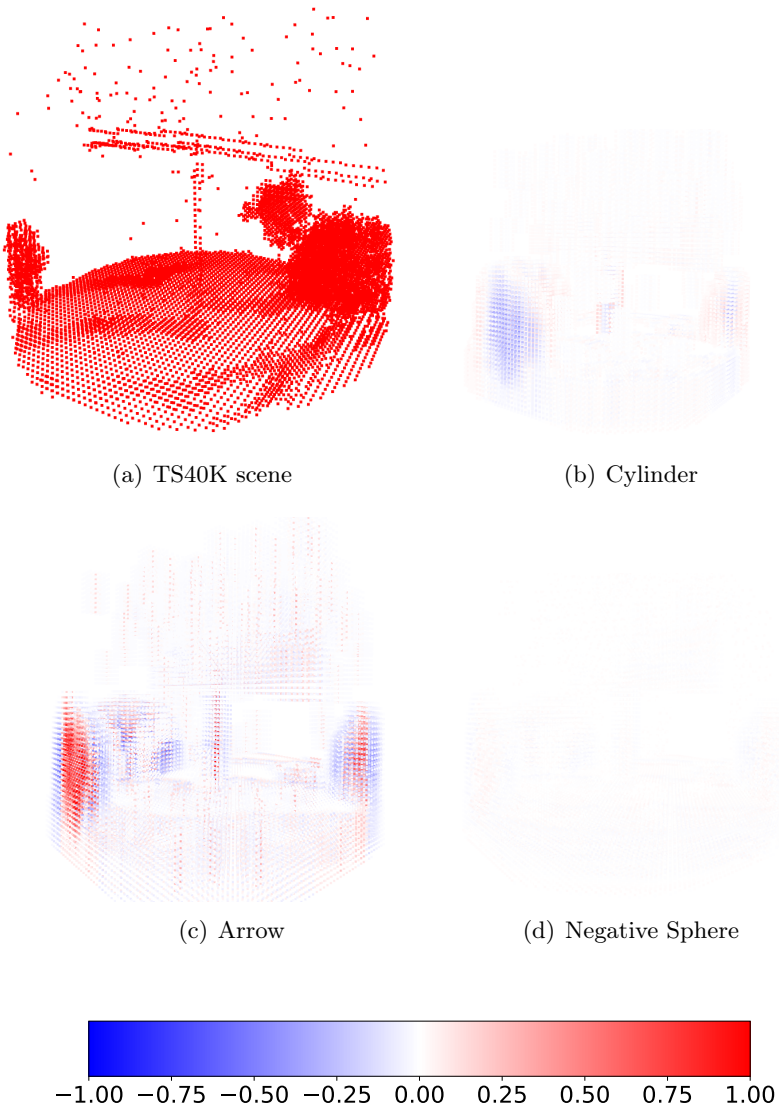
Figure 6. *The trainable parameters of SCENE-Net, ϑ and λ. Parameter h^* is not trainable, and λ^*_{NS} is defined as a function of the other mixing weights $\lambda^*_{NS} = 1 - \lambda_{Ar} - \lambda_{Cy}$*

5.5. Qualitative accuracy and quantitative metrics: SCENE-Net is more precise in detecting towers than a baseline CNN. To evaluate if SCENE-Net can correctly identify towers in landscapes of the noisy TS40K dataset, we chose the task of 3D semantic segmentation of power line towers. We trained SCENE-Net and a baseline CNN according to the protocol described in Section 5.2. We use a CNN with similar architecture and the same base operator (i.e., convolution) for feature retrieval. The main difference is their kernel initialization: SCENE-Net kernels are randomly initialized, but belong to a precise family of operators, while CNN kernels are completely random. Running models for 3D point cloud semantic segmentation [41, 9, 45, 48] was not done due to their computational requirements. The application penalizes the false positives more, thus we will emphasize Precision. Due to the imbalanced nature of the labels, we measured overall Precision, Recall, and Intersection over Union (IoU). Quantitatively, we observe a lift in Precision of 38%, and of 5% in IoU, and a drop of 13% in Recall (Table 1).

Table 1
3D semantic segmentation metrics on TS40K

Method	Precision	Recall	IoU
CNN	0.44 (± 0.07)	**0.26** (± 0.02)	0.53
SCENE-Net	**0.82** (± 0.08)	0.13 (± 0.05)	**0.58**

The lower Recall of SCENE-Net is due to mislabeled points (Figs. 1 and 9), and our choice to privilege Precision over Recall, in view of the fact that the Precision – Recall curve is slightly better for SCENE-Net (Fig 8).



(a) TS40K scene (b) Cylinder

(c) Arrow (d) Negative Sphere

Figure 7. *Post hoc analysis of SCENE-Net. We can examine the activation of each geometric operator and correlate it to the detection of certain elements in the scene. We see that the Arrow is responsible for the most activation, while the Negative Sphere has a smaller absolute value*

5.6. SCENE-Net is robust to noisy labels. It is important to assess the resilience to noisy labels in the ground-truth (GT) since 50% of 3D points labelled as supporting tower in the TS40K dataset are, in truth, patches of ground and road surface. These examples are abundant in the dataset and SCENE-Net is able to recover the body of the tower without detecting ground and power line patches that are mislabeled as tower (Fig. 9). Most noisy labels on this kind of dataset are due to annotation excess around the object of interest and are not randomly distributed. These consistently incorrect labels entail low Recall values (Tab. 1).

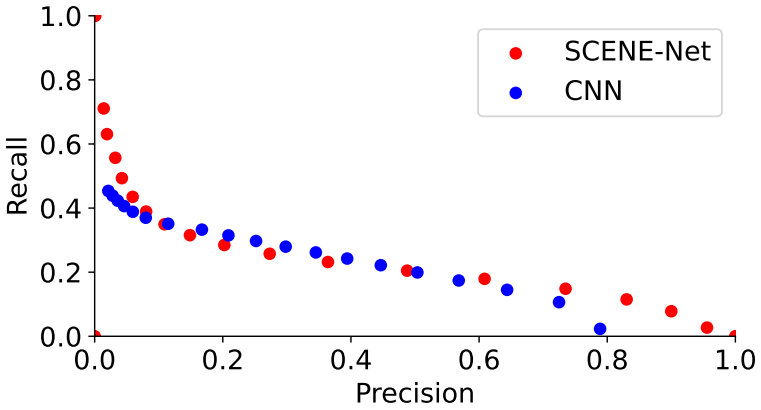


Figure 8. *Precision-Recall curve for SCENE-Net and the CNN benchmark, with changing detection threshold. Although our model SCENE-Net has two orders of magnitude fewer parameters than the CNN, it attains a comparable area under the P-R curve*

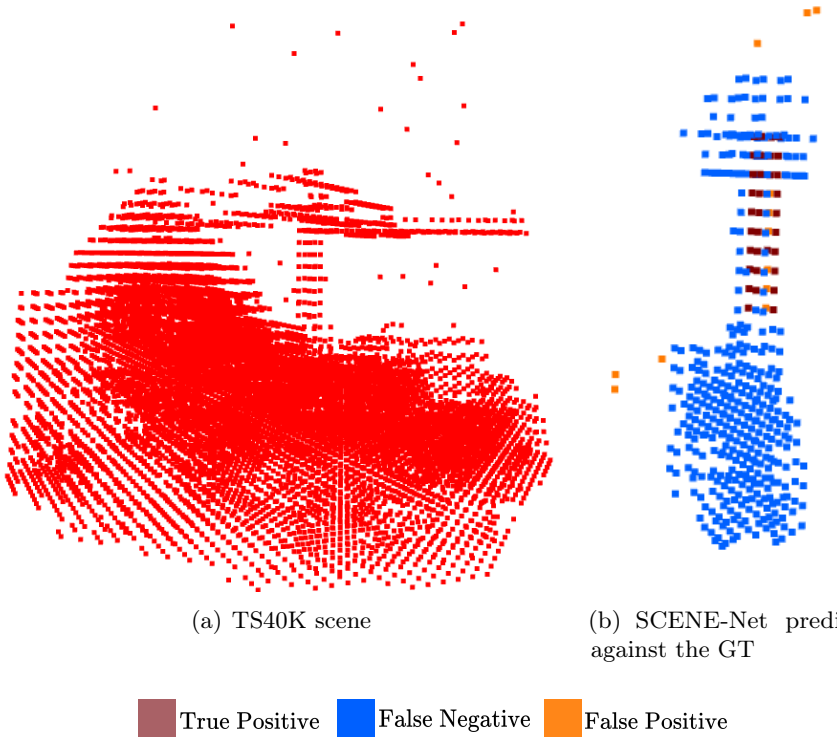
5.7. SCENE-Net has low data requirements and has modest training time in common hardware. The design of SCENE-Net embedded with GENEO observers culminates in a model with 11 meaningful trainable parameters. This enables the use of common hardware (see Section 5.2 for hardware specs) and a low data regime to train our model. The results reported in table 2 were achieved with 5% of the *SemanticKITTI* training set. Training SCENE-Net with 50% and 100% of the available training data leads to a variation of 0.5% in pole IoU performance. Moreover, the number of parameters of SCENE-Net remains unchanged regardless of kernel size, whereas in traditional models, such as the baseline CNN with 2190 parameters, the number of parameters grows exponentially with larger kernel sizes.

5.8. SCENE-Net on the SemanticKITTI: an efficient model for low-resource contexts. In Table 2, we present a comprehensive comparison of the performance of SCENE-Net against state-of-the-art models for the task of 3D semantic segmentation on the *SemanticKITTI* benchmark, specifically in terms of pole IoU, number of parameters, and the ratio of pole IoU to number of parameters. For this problem, we add to the GENEO loss in (4.3) the Tversky loss [34] to boost IoU performance of SCENE-Net:

$$\mathcal{L}_{Tversky}(y,\hat{y}) = 1 - \frac{y\hat{y} + \delta}{y\hat{y} + \alpha(\mathbf{1}-y)\hat{y} + \beta y(\mathbf{1}-\hat{y}) + \delta}$$

where y, $\hat{y}$ are the ground-truth and model prediction, $\alpha, \beta > 0$ are the penalty factors for false positives and false negatives respectively, and $\delta > 0$ is a smoothing term.

The comparison results demonstrate that SCENE-Net is a highly efficient model in terms of its parameter contribution. Our model has the lowest number of parameters, with at least a 5-order magnitude difference from the other models. SCENE-Net also has the highest ratio of pole IoU to a number of parameters, indicating that it can achieve a high level of performance with a minimal number of parameters. Although SCENE-Net does not achieve the highest pole IoU performance, it is somewhat on par with state-of-the-art models. Additionally, SCENE-Net



(a) TS40K scene

(b) SCENE-Net prediction against the GT

Figure 9. *SCENE-Net is robust against mislabeled data. Fig. 9(b) compares the prediction of SCENE-Net against the ground truth in Fig. 9(a). SCENE-Net detects the body of the tower, ignoring the patch of ground mislabeled as a tower*

provides intrinsic geometric interpretability and resource efficiency, which makes it a valuable model in high-risk tasks that require trustworthy predictions and good performance but have limited data and computing power.

5.9. Ablation Studies. In this section, we conduct ablation studies on SCENE-Net's architecture, specifically on the number of instances of each GENEO. All ablated models were tested on the TS40K validation set. Table 3 shows the following results: Models A and B are each equipped with a single GENEO, demonstrating an overall poor performance. The Negative Sphere (NS) GENEO proved essential for our observer to disregard arboreal elements in the scene. Models C and D study if employing the Cylinder or Arrow combined with NS is enough to analyze the TS40K scenes. However, SCENE-Net's architecture (model E) yields better results. Lastly, models F and G test the use of multiple instances of each GENEO, however, this proved to decrease performance when compared to model E.

5.10. SCENE-Net inference in high resolution, when trained with low-resolution kernel sizes. One of the issues of voxel-based models is the computational cost of 3D convolutions with large kernels and high-resolution voxel grids. Here, a CNN architecture leads to an exponential increase in training time. SCENE-Net has a continuous functional observer

Table 2

Semantic segmentation on SemanticKITTI, only methods that report the parameter count were included. Large models are included for comparison but cannot be used in low-resource contexts. Parameter efficiency is $\frac{Pole\ IoU}{log\ \#Parameters}$

Method	Pole IoU	#Parameters (M)	Parameter Efficiency
PointNet++ [28]	16.9	1.48	1.19
TangentConv [39]	35.8	0.4	2.77
KPConv [41]	56.4	14.9	3.41
RandLA-Net [20]	51.0	1.24	3.63
RPVNet [45]	**64.8**	24.8	3.80
SparseConv [14]	57.9	2.7	3.91
JS3C-Net [47]	60.7	2.7	4.09
SPVNAS [37]	64.3	12.5	4.62
SCENE-Net (Ours)	57.5	**1.1e-5**	**23.98**

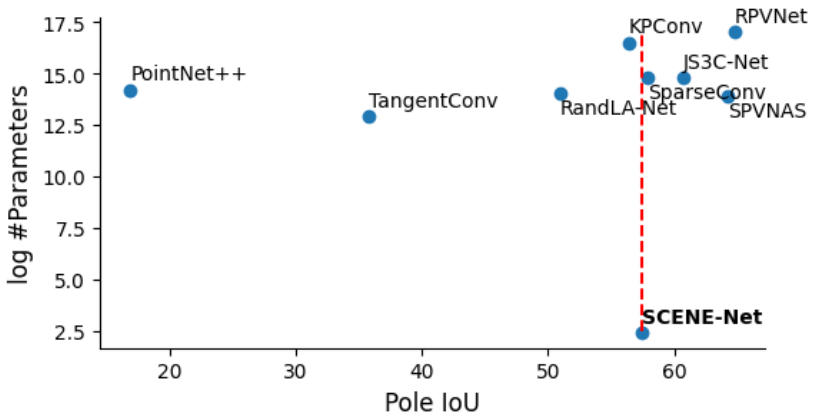
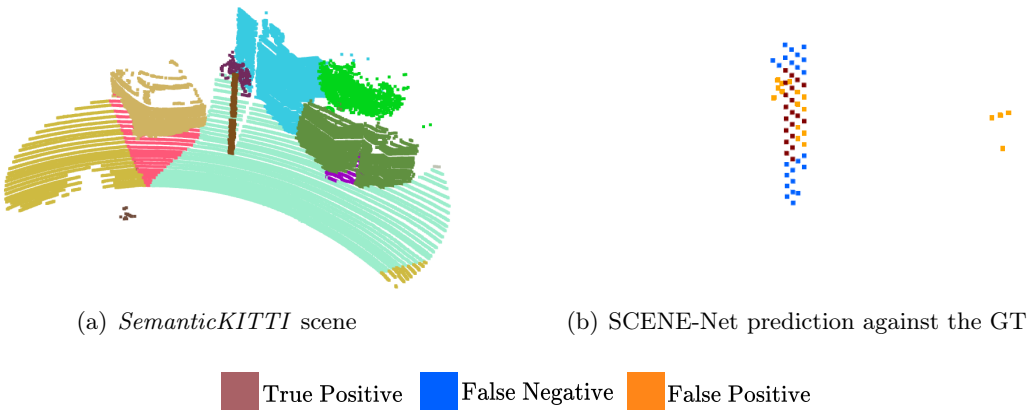


Figure 10. *Semantic Segmentation results on the SemanticKITTI benchmark. Log scale is used for more intelligible comparison*

of the raw input providing an analysis of its components. Unlike traditional models, this definition is **independent** from the input size as well as its own discretization (kernel size). In this experiment, we trained SCENE-Net with voxel grids of 64^3 and then applied to higher resolutions, such as 128^3, with good qualitative results (Fig. 12).

5.11. Template Matching Comparison. Template matching offers a direct approach to pattern detection by measuring the relation between the input data and a specific pattern of interest. Since we define the properties that constitute supporting towers, a comparison between SCENE-Net and this method is reasonable.

Our model combines GENEOs through convex combination and all shape parameters and convex coefficients are found through backpropagation, creating an aggregated GENEO operator - a data observer for composite shape signatures (e.g., cylinder+arrow+negative sphere). In



(a) *SemanticKITTI* scene (b) SCENE-Net prediction against the GT

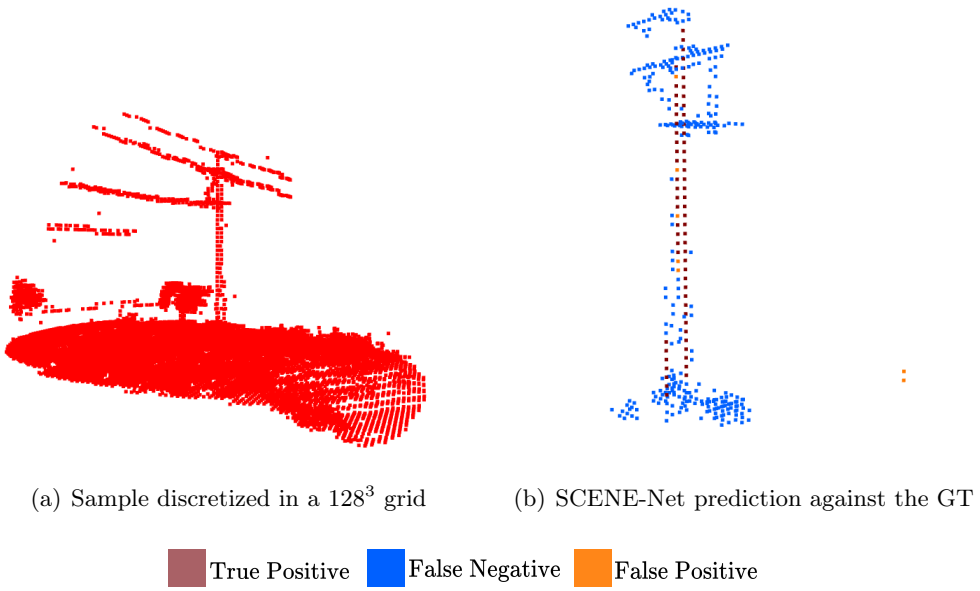
Figure 11. *Qualitative results of SCENE-Net on SemanticKITTI for pole detection. Fig. 11(b) compares the prediction of SCENE-Net against the ground truth in Fig. 11(a). SCENE-Net detects the body of the pole while disregarding the rest of the 3D scene*

Table 3
Ablation Study of SCENE-Net on TS40K validation set.

Model	Cylinder	Arrow	Neg. Sphere	Precision	Recall	IoU
A	1	0	0	0	0	0
B	0	1	0	0	0	0
C	1	0	1	0.34	0.01	0.12
D	0	1	1	0.13	0.01	0.08
E (Ours)	**1**	**1**	**1**	**0.82**	**0.13**	**0.58**
F	2	2	2	0.56	0.16	0.53
G	3	3	3	0.37	0.22	0.56

geometrical methods, such as template matching, this composition is not trivial. Employing template matching on 3D data is slow, and the parameter initialization (e.g., a Cylinder radius) is a crucial step. In addition, template matching is not directly applicable to some patterns, e.g., the Neg. Sphere, as it tries to diminish the activation of specific elements. We tested template matching of the TS40K with a Cylinder (the same definition used in SCENE-Net) with a radius equal to the average tower radius in the training set. It runs for 12 hours on the validation set of TS40K with an average precision (AP) of 0.001, while SCENE-Net's AP = 0.2.

6. Discussion. Traditional companies, like utilities, need a resource-efficient, responsible application of ML models for the segmentation of real-world point clouds, e.g., for inspecting thousands of kilometers of a power grid. In this paper, we present SCENE-Net, a low-resource white-box model for 3D semantic segmentation. Our approach offers a unique combination of intrinsic geometric interpretability, resource efficiency, and on-par state-of-the-art performance.



(a) Sample discretized in a 128^3 grid (b) SCENE-Net prediction against the GT

Figure 12. *SCENE-Net is independent of the input and kernel size. Our model was trained with voxel grids of shape* 64^3 *and kernel size* 9^3*. Fig. 12(b) shows SCENE-Net's prediction against the ground truth of the* 128^3 *input grid in Fig. 12(a) using a kernel-size of* $12 \times 5 \times 5$*.*

Limitations. Our model prioritizes transparency and performance over broader applicability while allowing a flexible extension. In this paper, we segmented pole-like structures. State-of-the-art methods often show similar trade-offs, for example, 3D semantic segmentation models are tailored for autonomous driving [47, 45]. SCENE-Net requires a knowledge engineering phase that is not necessary for black-box models. Despite these limitations, we believe that the transparency and efficiency of SCENE-Net make it a valuable tool for high-stakes applications. To detect other shapes, other geometrical observers have to be created. As the convex combination of GENEOs is a GENEO, this problem is mitigated by creating a library of primary shapes to be combined to form more complex geometrical structures. This is relevant follow-up work, but out of the scope of this paper. Multiclass and multilabel segmentation can be achieved by combining different binary class segmentation models.

Impact. From our experience deploying SCENE-Net within a utility company, low-resource transparent systems can critically help human decision-making—here, by facilitating fast and careful inspection of power lines with interpretable signals of observed geometrical properties. With only three observers and 11 meaningful trainable parameters, SCENE-Net can help reduce the risk of power outages and forest fires by learning from data.

REFERENCES

[1] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, *Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions*, Journal of big Data, 8 (2021), pp. 1–74.

[2] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall,

Semantickitti: A dataset for semantic scene understanding of lidar sequences, in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 9297–9307.

[3] M. G. Bergomi, P. Frosini, D. Giorgi, and N. Quercioli, *Towards a topological–geometrical theory of group equivariant non-expansive operators for data analysis and machine learning*, Nature Machine Intelligence, 1 (2019), pp. 423–433.

[4] G. Bocchi, P. Frosini, A. Micheletti, A. Pedretti, C. Gratteri, F. Lunghini, A. R. Beccari, and C. Talarico, *Geneonet: A new machine learning paradigm based on group equivariant non-expansive operators. an application to protein pocket detection*, arXiv preprint arXiv:2202.00451, (2022).

[5] S. Botteghi, M. Brasini, P. Frosini, and N. Quercioli, *On the finite representation of group equivariant operators via permutant measures*, arXiv preprint arXiv:2008.06340, (2020).

[6] P. Cascarano, P. Frosini, N. Quercioli, and A. Saki, *On the geometric and riemannian structure of the spaces of group equivariant non-expansive operators*, arXiv preprint arXiv:2103.02543, (2021).

[7] H. Chefer, S. Gur, and L. Wolf, *Transformer interpretability beyond attention visualization*, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2021, pp. 782–791.

[8] Z. Chen, Y. Bei, and C. Rudin, *Concept whitening for interpretable image recognition*, Nature Machine Intelligence, 2 (2020), pp. 772–782.

[9] R. Cheng, R. Razani, E. Taghavi, E. Li, and B. Liu, *2-s3net: Attentive feature fusion with adaptive feature selection for sparse semantic segmentation network*, in Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition, 2021, pp. 12547–12556.

[10] F. Conti, P. Frosini, and N. Quercioli, *On the construction of group equivariant non-expansive operators via permutants and symmetric functions*, Frontiers in Artificial Intelligence, 5 (2022), p. 16.

[11] L. Ding, J. Wang, and Y. Wu, *Electric power line patrol operation based on vision and laser slam fusion perception*, in 2021 IEEE 4th International Conference on Automation, Electronics and Electrical Engineering (AUTEEE), IEEE, 2021, pp. 125–129.

[12] F. Doshi-Velez and B. Kim, *Towards a rigorous science of interpretable machine learning*, arXiv preprint arXiv:1702.08608, (2017).

[13] R. C. Fong and A. Vedaldi, *Interpretable explanations of black boxes by meaningful perturbation*, in Proceedings of the IEEE international conference on computer vision, 2017, pp. 3429–3437.

[14] B. Graham, M. Engelcke, and L. Van Der Maaten, *3d semantic segmentation with submanifold sparse convolutional networks*, in Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2018, pp. 9224–9232.

[15] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, *A survey of methods for explaining black box models*, ACM computing surveys (CSUR), 51 (2018), pp. 1–42.

[16] T. Guo, L. Xu, Y. Chen, Y. Liu, J. Zhao, X. Luo, and S. Wu, *Research on point cloud power line segmentation and fitting algorithm*, in 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), vol. 1, IEEE, 2019, pp. 2404–2409.

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

[18] T. Hackel, N. Savinov, L. Ladicky, J. D. Wegner, K. Schindler, and M. Pollefeys, *Semantic3d. net: A new large-scale point cloud classification benchmark*, arXiv preprint arXiv:1704.03847, (2017).

[19] Q. Hu, B. Yang, S. Khalid, W. Xiao, N. Trigoni, and A. Markham, *Towards semantic segmentation of urban-scale 3d point clouds: A dataset, benchmarks and challenges*, in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 4977–4987.

[20] Q. Hu, B. Yang, L. Xie, S. Rosa, Y. Guo, Z. Wang, N. Trigoni, and A. Markham, *Randla-net: Efficient semantic segmentation of large-scale point clouds*, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 11108–11117.

[21] B.-S. Hua, M.-K. Tran, and S.-K. Yeung, *Pointwise convolutional neural networks*, in Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2018, pp. 984–993.

[22] T. Le and Y. Duan, *Pointgrid: A deep network for 3d shape understanding*, in Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2018, pp. 9204–9214.

[23] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, *PointCNN: Convolution on x-transformed points*, Advances in Neural Information Processing Systems, 31 (2018).

[24] Z. C. Lipton, *The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery.*, Queue, 16 (2018), pp. 31–57.
[25] J. Long, E. Shelhamer, and T. Darrell, *Fully convolutional networks for semantic segmentation*, in Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2015, pp. 3431–3440.
[26] K. Muhammad, A. Ullah, J. Lloret, J. Del Ser, and V. H. C. de Albuquerque, *Deep learning for safe autonomous driving: Current challenges and future directions*, IEEE Transactions on Intelligent Transportation Systems, 22 (2020), pp. 4316–4336.
[27] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, *Pointnet: Deep learning on point sets for 3d classification and segmentation*, in Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2017, pp. 652–660.
[28] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, *Pointnet++: Deep hierarchical feature learning on point sets in a metric space*, Advances in neural information processing systems, 30 (2017).
[29] D. Rethage, J. Wald, J. Sturm, N. Navab, and F. Tombari, *Fully-convolutional point networks for large-scale point clouds*, in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 596–611.
[30] M. Ribeiro and J. Leite, *Aligning artificial neural networks and ontologies towards explainable ai*, in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, 2021, pp. 4932–4940.
[31] M. T. Ribeiro, S. Singh, and C. Guestrin, *" why should i trust you?" explaining the predictions of any classifier*, in Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, 2016, pp. 1135–1144.
[32] M. T. Ribeiro, S. Singh, and C. Guestrin, *Anchors: High-precision model-agnostic explanations*, in Proceedings of the AAAI conference on artificial intelligence, vol. 32, 2018.
[33] C. Rudin, *Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead*, Nature Machine Intelligence, 1 (2019), pp. 206–215.
[34] S. S. M. Salehi, D. Erdogmus, and A. Gholipour, *Tversky loss function for image segmentation using 3d fully convolutional deep networks*, in Machine Learning in Medical Imaging: 8th International Workshop, MLMI 2017, Held in Conjunction with MICCAI 2017, Quebec City, QC, Canada, September 10, 2017, Proceedings 8, Springer, 2017, pp. 379–387.
[35] M. Steininger, K. Kobs, P. Davidson, A. Krause, and A. Hotho, *Density-based weighting for imbalanced regression*, Machine Learning, 110 (2021), pp. 2187–2211.
[36] H. Su, V. Jampani, D. Sun, S. Maji, E. Kalogerakis, M.-H. Yang, and J. Kautz, *Splatnet: Sparse lattice networks for point cloud processing*, in Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2018, pp. 2530–2539.
[37] H. Tang, Z. Liu, S. Zhao, Y. Lin, J. Lin, H. Wang, and S. Han, *Searching efficient 3d architectures with sparse point-voxel convolution*, in European conference on computer vision, Springer, 2020, pp. 685–702.
[38] G. Tao, X. Lianggang, Y. Heng, Y. Liugui, Z. Jian, W. Shaohua, and W. Di, *Study on segmentation algorithm with missing point cloud in power line*, in 2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), IEEE, 2019, pp. 1895–1899.
[39] M. Tatarchenko, J. Park, V. Koltun, and Q.-Y. Zhou, *Tangent convolutions for dense prediction in 3d*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 3887–3896.
[40] L. Tchapmi, C. Choy, I. Armeni, J. Gwak, and S. Savarese, *Segcloud: Semantic segmentation of 3d point clouds*, in 2017 International Conference on 3D vision (3DV), IEEE, 2017, pp. 537–547.
[41] H. Thomas, C. R. Qi, J.-E. Deschaud, B. Marcotegui, F. Goulette, and L. J. Guibas, *Kpconv: Flexible and deformable convolution for point clouds*, in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 6411–6420.
[42] E. Voita, D. Talbot, F. Moiseev, R. Sennrich, and I. Titov, *Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned*, arXiv preprint arXiv:1905.09418, (2019).
[43] P.-S. Wang, Y. Liu, Y.-X. Guo, C.-Y. Sun, and X. Tong, *O-CNN: Octree-based convolutional neural networks for 3d shape analysis*, ACM Transactions On Graphics (TOG), 36 (2017), pp. 1–11.
[44] W. Wu, Z. Qi, and L. Fuxin, *Pointconv: Deep convolutional networks on 3d point clouds*, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 9621–9630.

[45] J. Xu, R. Zhang, J. Dou, Y. Zhu, J. Sun, and S. Pu, *Rpvnet: A deep and efficient range-point-voxel fusion network for lidar point cloud segmentation*, in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 16024–16033.

[46] M. Xu, Z. Zhou, and Y. Qiao, *Geometry sharing network for 3d point cloud classification and segmentation*, in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, 2020, pp. 12500–12507.

[47] X. Yan, J. Gao, J. Li, R. Zhang, Z. Li, R. Huang, and S. Cui, *Sparse single sweep lidar point cloud segmentation via learning contextual shape priors from scene completion*, in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, 2021, pp. 3101–3109.

[48] X. Yan, J. Gao, C. Zheng, C. Zheng, R. Zhang, S. Cui, and Z. Li, *2DPASS: 2D Priors Assisted Semantic Segmentation on LiDAR Point Clouds*, arXiv e-prints, (2022), arXiv:2207.04397, p. arXiv:2207.04397, https://arxiv.org/abs/2207.04397.

[49] Q. Zhang, Y. N. Wu, and S.-C. Zhu, *Interpretable convolutional neural networks*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018.

[50] X. Zhu, H. Zhou, T. Wang, F. Hong, Y. Ma, W. Li, H. Li, and D. Lin, *Cylindrical and asymmetrical 3d convolution networks for lidar segmentation*, in Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition, 2021, pp. 9939–9948.