AgriEngineering

Article

Benchmark Study of Point Cloud Semantic Segmentation Architectures on Strawberry Organs

Rundong Xu, Hiroki Naito * and Fumiki Hosoi

Graduate School of Agricultural and Life Sciences, The University of Tokyo, 1-1-1 Yayoi, Bunkyo, Tokyo 113-8657, Japan; xurundong234@g.ecc.u-tokyo.ac.jp (R.X.); ahosoi@g.ecc.u-tokyo.ac.jp (F.H.)
* Correspondence: naito@g.ecc.u-tokyo.ac.jp

Abstract: With the increasing consumer demand for healthy and natural foods, strawberries have emerged as one of the most popular small berries globally. Consequently, careful investigation of the relationship between leaf photosynthetic activity (source strength) and fruit development (sink strength) during strawberry growth provides important insights for maximizing the production potential of this crop. This objective necessitates accurate strawberry organ segmentation. Recently, advancements in deep learning (DL) have driven the development of numerous semantic segmentation models that have performed effectively on benchmark datasets. Nevertheless, their small-organ plant segmentation efficacy remains insufficiently explored. Consequently, this study evaluates eight representative point-based semantic segmentation models for the strawberry organ segmentation task: PointNet++, PointMetaBase, Point Transformer V2, Swin3D, KPConv, RandLA-Net, PointCNN, and Sparse UNet. The employed dataset comprises two components: the open-source LAST-Straw strawberry dataset and a custom Japanese strawberry dataset. Strawberry point cloud organs were categorized into four classes: leaf, stem, flower, and berry. The sparse convolution-based Sparse UNet achieved the highest mean intersection over union of 81.3, followed by the PointMetaBase model at 80.7. This study provides insights into the strengths and limitations of existing architectures, assisting researchers and practitioners in selecting appropriate models for strawberry organ segmentation tasks.

Keywords: deep learning; point cloud; semantic segmentation; strawberry phenotyping

check for updates

Academic Editors: Yee Siang Gan and Sze-Teng Liong

Received: 18 April 2025
Revised: 26 May 2025
Accepted: 4 June 2025
Published: 9 June 2025

Citation: Xu, R.; Naito, H.; Hosoi, F. Benchmark Study of Point Cloud Semantic Segmentation Architectures on Strawberry Organs. *AgriEngineering* **2025**, *7*, 181. https://doi.org/10.3390/agriengineering7060181

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Strawberry (*Fragaria* × *ananassa* Duch.) is a globally significant economic crop. Its yield and quality are markedly influenced by the dynamic relationship between the source (photosynthetic leaves) and the sink (developing fruits) [1]. In plant physiology, the terms 'source' and 'sink' are used to describe organs that produce and store assimilates, respectively. Leaves predominantly act as a source due to photosynthesis, producing carbohydrates that are translocated to developing fruits, the primary sink organs [2–4]. However, it is acknowledged that leaves themselves can temporarily act as sinks, especially during early development or under stress conditions [5]. In our study, leaves are specifically considered as sources in the context of mature strawberry plants actively transferring assimilates to developing fruits. The strength of the sink–source relationship determines the fruit's ability to compete for carbon, a critical factor affecting fruit growth and quality. Studies have demonstrated that sink–source strength is determined by both the size of the sink–source (e.g., fruit weight and volume) and the activity of the sink–source (e.g., metabolic efficiency) [6]. During the early growth stages of strawberries, organ size is the

primary factor influencing sink–source strength [7]. To measure the size of each organ during these growth stages, strawberry point cloud data must first be acquired and then segmented into individual organ point clouds.

By segmenting the three-dimensional structure of strawberry organs from their point cloud representations, we can precisely quantify key morphological traits—volume, surface area, angles, and counts—for each organ. If we can accurately quantify these morphological traits of each organ, it will be possible to acquire and analyze a time-series of whole-plant point cloud models from the seedling to maturity. This will enable detailed analysis of source–sink dynamics during strawberry development in future works. Moreover, comparative analyses across different cultivars afford critical insights into varietal differences in both morphology and assimilate partitioning, thereby informing breeding programs. Analogous approaches have been demonstrated in other crops: Daviet et al. reconstructed maize plants at multiple growth stages to track organ-level volumetric changes and quantify source–sink fluxes (e.g., leaf photosynthate allocation versus ear storage) [8]. Liu et al. developed an automated point cloud segmentation pipeline for cotton that isolates stems, leaves, and bolls and computes their volume, surface area, and count for high-throughput phenotyping [9].

In recent years, with the advancement of DL technology, numerous models have achieved excellent results in semantic segmentation tasks on public datasets such as S3DIS and ScanNet [10,11]. In the context of strawberry organ segmentation, many researchers have utilized 2D DL models with RGB images to achieve outstanding performance. For example, Fujinaga employed the DeepLabV3+ model for semantic segmentation of strawberries via RGB images in the development of a strawberry-picking robot [12]. However, RGB-based semantic segmentation faces challenges such as data loss owing to leaf occlusion. Other researchers have adopted 3D point clouds for strawberry semantic segmentation. For example, Le Louëdec and Cielniak [13] evaluated the performance of stereo and time-of-flight cameras in perceiving and segmenting strawberry shapes and proposed a convolutional network that integrates point clouds and normal information. They revealed the potential and limitations of 3D perception technology in such scenarios. Although satisfactory segmentation results can be achieved, there remains room for improvement in segmenting strawberry organs.

In addition to strawberry segmentation, other studies have applied point cloud semantic segmentation to different plants. Patel et al. [14] used LiDAR point clouds and DL models (e.g., PointNet++) to achieve high-precision segmentation (91.5%) of sorghum plant organs (stems, leaves, and panicles) and extract phenotypic features, with results highly correlated with manual measurements (R^2 up to 0.97). This provided an efficient automated solution for high-throughput plant phenotyping. Yang et al. [15] addressed the issue of insufficient training data for semantic segmentation of corn organs by developing an automatic labeling tool, employing PointNet++ for semantic segmentation and HAIS. Moreover, when DL models are selected for semantic segmentation, there is no systematic investigation into whether the neural network structure is suitable for plant organ segmentation. When faced with numerous state-of-the-art (SOTA) models, determining the neural network structure most beneficial for strawberry organ segmentation is challenging.

To address these challenges, this study selected eight DL models—PointNet++ [16], Point-MetaBase [17], Point Transformer V2 [18], Swin3D [19], KPConv [20], Rand-LA-Net [21], PointCNN [22], and Sparse Unet [23]—to systematically evaluate and compare the performance of existing DL models for strawberry organ semantic segmentation tasks, aiming to identify architectures suitable for precision farming applications. For the dataset, we utilized the open-source LAST-Straw strawberry dataset, which was captured via a high-precision handheld scanner that accurately reproduced the true colors and structures

of strawberries, laying the groundwork for correctly learning structural features for organ segmentation [24]. Additionally, another portion of the dataset, consisting of an in-house Japanese strawberry point cloud collection, was collected by the authors to enhance model robustness. This part of the dataset was acquired using a 3D Gaussian splatting method, which generates high-precision point cloud models by capturing a series of images or videos and employing DL neural networks. This method faithfully reproduces the true structure and color of strawberries [25]. Our research contributions can be summarized as follows:

1. Given the limited existing studies on this topic, this work evaluates the performance of eight neural network models for strawberry organ semantic segmentation.
2. Considering the complexity of the scenes and the characteristics of each point, we provide specific recommendations for selecting DL architectures for plant organ segmentation. Furthermore, we reveal the specific limitations of existing DL architectures when segmenting rare classes in strawberry organs. These findings could inform future selection or adaptation of existing architectures for improved performance in similar tasks.

2. Materials and Methods

2.1. Dataset Composition and Acquisition

To demonstrate the effectiveness and robustness of DL semantic segmentation for strawberry organ segmentation, we utilized the annotated LAST-Straw dataset and captured a portion of the strawberry dataset from Japanese strawberries. This dataset consists of 13 annotated LAST-Straw strawberry point cloud files and 11 strawberry point cloud files from the Japanese cultivar dataset captured by the authors. Out of the 24 pots of strawberries in the overall dataset, 15 pots were used as the training set, 6 pots were used as the validation set, and 3 pots were used as the test set. The number of point clouds for each category in the dataset, along with the division into training, validation, and test sets, is shown in Table 1. The overall proportions of the training, validation, and test sets are 53%, 27%, and 20%, respectively. According to the data in Table 2, the leaf category has the highest overall proportion, reaching about 85%, followed by the stem category with approximately 9%. The berry and flower categories account for only about 3% and 4%, respectively. This data reflects the extreme imbalance of point clouds across categories, posing a challenge for neural network models to perform effective semantic segmentation tasks.

Table 1. Number of point clouds contained in the training set, validation set, and test set across different categories.

	Total Points Numbers	Train Dataset	Val Dataset	Test Dataset
Leaf	5,859,645	3,067,178	1,597,567	1,194,900
Stem	642,909	352,095	168,428	122,386
Berry	171790	94,099	41,737	35,954
Flower	217,482	110,484	44,672	62,326
Total points	6,891,826	3,623,856	1,852,404	1,415,566

Table 2. Percentage of point clouds contained in the training set, validation set, and test set across different categories.

	Total Points Percentage	Train Dataset	Val Dataset	Test Dataset
Leaf	85.02%	84.64%	86.24%	84.41%
Stem	9.33%	9.72%	9.09%	8.65%
Berry	2.49%	2.60%	2.25%	2.54%
Flower	3.16%	3.05%	2.41%	4.40%

2.1.1. LAST-Straw Dataset

The dataset includes Driscoll's Katrina (variety A) and Driscoll's Zara (variety B) strawberry varieties. The Last-Straw dataset was captured by the University of Lincoln, UK, using the EinScan Pro 2X Plus handheld 3D scanner (Hangzhou, China). The scanner is equipped with an LED light source that projects specific structured light patterns (typically a series of stripes or grids) onto the surface of the object being scanned. When the structured light is projected onto the object's surface, the light patterns deform because of the varying shapes of the object. Using the principle of triangulation, the scanner calculates the three-dimensional coordinates (X, Y, Z) of each point on the object's surface on the basis of the known angle between the light source and the camera, as well as the degree of pattern deformation. The dataset provides high-precision, high-quality strawberry point cloud data with color information. Using EXScanPro (version 3.7.4) provided by Shining 3D, the data can be directly exported as OBJ models and then converted to XYZ-format point cloud files in CloudCompare (version 2.11.alpha).

2.1.2. Japanese Cultivar Dataset

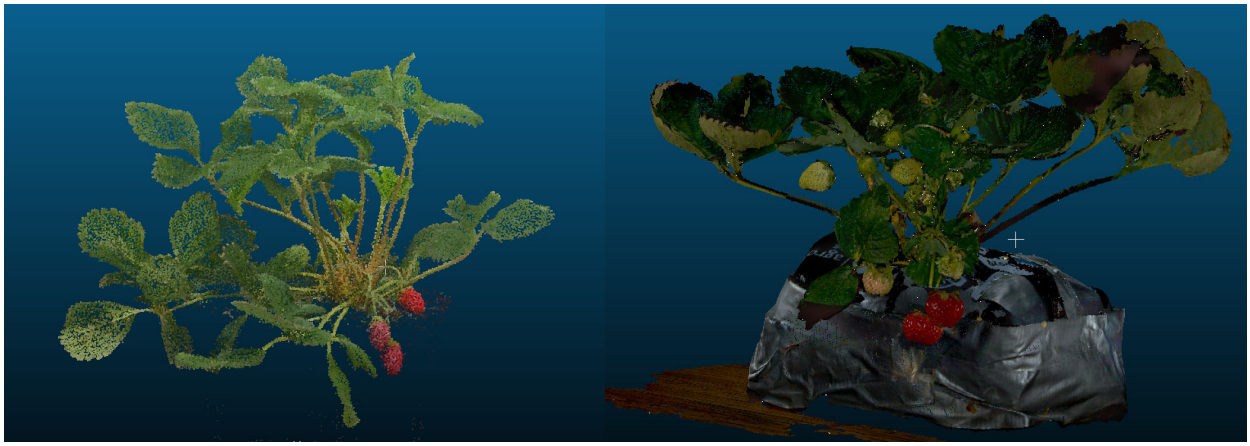
The dataset captured in this study consists of eleven plants from three Japanese strawberry varieties: Sachinoka, Toyonoka, and Fukuba. The three types of strawberry plants were cultivated in an artificial climate chamber at the Faculty of Agriculture and Life Sciences, University of Tokyo, as shown in Figure 1, with strawberry point cloud data captured at regular intervals of nine days. The Japanese cultivar dataset was captured using the Scaniverse (version 4.0.7) software installed on an iPhone 15 with iOS 18.3.2, and the 3DGS (3D Gaussian Splatting) mode of the software was employed to generate the point cloud models of the scanned objects. The working principle of 3DGS is based on extracting spatial data points from video or image inputs and initializing these points as a splat cloud defined by Gaussian functions. Each Gaussian splat has a position (X, Y, Z), color, and a covariance matrix, which describes its shape and orientation. The scanning process generates a three-dimensional radiance field representation by analyzing the perspective changes of the object in the video frames, combined with light field information. The files output by Scaniverse undergo various preprocessing steps in CloudCompare.



Figure 1. The strawberries included in the Japanese cultivar dataset were grown in an environment equipped with an automatic irrigation system and devices for monitoring temperature and light.

2.2. Data Preprocessing

Owing to the complex nature of point cloud data, such as the raw point cloud shown in Figure 2, which inherently contains a significant amount of noise, data preprocessing is essential before training commences. We applied different preprocessing techniques to the two distinct datasets. For the LAST-Straw dataset preprocessing, the original labels included eleven categories—such as soil, buds, and point clouds of neighboring strawberry plants—not belonging to the target plant. To align label categories with our in-house dataset and minimize ambiguity, we removed all of these extra labels, retaining only four classes: leaf, stem, berry, and flower. We also uniformly scaled the point clouds to ensure smooth data ingestion by the models. Since the Shining 3D software applies point cloud denoising during export, no further filtering was performed on the LAST-Straw data.



(**a**) (**b**)

Figure 2. Original point cloud files of the dataset: (**a**) raw point cloud of the Japanese cultivar dataset; (**b**) raw point cloud of the LAST-Straw dataset.

For the Japanese cultivar dataset preprocessing, we carried out the following preprocessing steps. First, for label annotation, we manually segmented the strawberry point clouds in CloudCompare and assigned the corresponding labels, ensuring that the point cloud file formats and label categories matched those of the LAST-Straw dataset. Next, for point cloud data processing, to remove noise generated by 3DGS, we applied radius filtering to select and eliminate outliers. Radius filtering removes isolated points with insufficient neighboring points by setting a radius threshold, thereby reducing noise while preserving the main structure. Finally, we performed coordinate transformations and scale adjustments on this dataset's point clouds to ensure their scale was consistent with that of the LAST-Straw dataset.

2.3. Model Selection

With the advent of DL, numerous point cloud semantic segmentation models have emerged. Early voxelization methods discretize point clouds into regular or adaptive 3D grids and apply 3D convolutional neural networks for classification and segmentation. However, these approaches suffer from high memory demands and loss of geometric information. Subsequently, projection methods project point clouds into 2D representations (such as multiview images or depth maps) and leverage mature 2D convolutional networks for processing. Although these methods offer strong real-time performance, they sacrifice 3D spatial relationships. It was not until the introduction of PointNet, which directly operates on sparse raw point clouds, that the limitations of voxelization and projection were fully overcome. Although PointNet addresses the issue of local feature extraction, it

overlooks global features, leading to the development of PointNet++, which introduced hierarchical multiscale feature extraction. This was followed by RandLA-Net, which improved efficiency and local geometric modeling through random sampling and adaptive feature aggregation. Convolution-based methods, such as PointCNN, adapt standard convolution with X-Conv, whereas KPConv uses kernel point convolution to handle complex geometries. Sparse UNet will first voxelize the point cloud and then perform convolution operations on the voxels containing only the point cloud. Transformer-based methods, such as Point Transformer, employ local self-attention mechanisms, with PTv2 and PTv3 enhancing robustness through grouping and patch attention and Swin3D using sliding window attention for multiscale feature extraction to model local and global contexts in complex scenes.

Table 3 shows the basic principles, publication dates, and mIoU performance on the public ScanNet dataset for eight DL models. ScanNet is an open-source point cloud dataset focused primarily on indoor furniture and objects, where the data consists mainly of a room containing various furniture and items, each with corresponding labels. The point cloud distribution in the ScanNet dataset differs significantly from that in the strawberry dataset. Therefore, some models that perform well on ScanNet may not necessarily excel at the task of strawberry organ segmentation. On the basis of these principles, we selected eight representative models: PointNet++, PointMetaBase, Point Transformer V2, Swin3D, KPConv, RandLA-Net, PointCNN, and Sparse UNet.

Table 3. Eight selected model types, publication dates, and intersection of union (IoU) scores on the ScanNet public dataset.

Model Name	Reference	Model Type	Publish Date	Performance on ScanNet
PointNet++	[16]	Point-based	2017	53.5%
PointMetaBase	[17]	Point-based	2023	71.4%
Point Transformer V2	[18]	Attention-based	2022	75.2%
Swin3D	[19]	Convolution with attention	2023	75.5%
KPConv	[20]	Convolution-based	2019	69.2%
RandLA-Net	[21]	Attention-based	2020	64.5%
PointCNN	[22]	Convolution-based	2018	45.8%
Sparse UNet	[23]	Convolution-based	2017	72.5%

2.3.1. PointNet++

PointNet++ [16] is a point-based point cloud semantic segmentation model that inherits the advantages of PointNet [26] in handling unordered point cloud data while enhancing the capture of local geometric features through hierarchical feature extraction. It employs set abstraction modules utilizing K-nearest neighbors (KNNs) to progressively extract point cloud features from local to global scales and uses the farthest point sampling and grouping mechanisms to improve the coverage and accuracy of feature extraction.

2.3.2. PointMetaBase

PointMetaBase [17] provides a detailed analysis of mainstream architectures in terms of neighbor update, neighbor aggregation, point update, and positional embedding. It adjusts the order of neighbor aggregation and MLP linear transformations to save computation time and improve performance. Before the neighbor aggregation stage, the model integrates explicit position embedding to directly embed geometric positional information into features, enhancing the perception of spatial information [18]. The entire structure adopts hierarchical abstraction modules (PointMetaSA) and uses max pooling to progressively extract local and global features. Each stage achieves efficient feature extraction through the interaction of these core components and has low computational complexity.

2.3.3. Point Transformer V2

The workflow of this model includes grouped vector attention (GVA) [18], improved positional encoding, and a partition-based pooling method. The GVA groups vector attention and shares weights, reducing the number of parameters while preserving the advantages of multihead attention [27]. By integrating multiplicative and additive positional encoding mechanisms, geometric positional information is embedded into the features, enhancing spatial relationship modeling capabilities. In the pooling stage, a partition-based pooling approach is adopted, improving spatial alignment and efficiency through local aggregation via grid partitioning. Ultimately, a U-Net-style encoder–decoder structure is utilized to progressively extract and fuse multiscale features, achieving efficient point cloud understanding.

2.3.4. Swin3D

Swin3D [19] is based on multilevel sparse voxel grids, employs sparse convolution to perform initial feature embedding, and leverages multistage Swin3D blocks to achieve multiscale feature extraction [23]. Each block uses self-attention mechanisms [27] with regular and shifted windows (W-MSA3D and SW-MSA3D) combined with context-relative signal embedding (cRSE) to capture the geometric and attribute information of the point cloud. Downsampling is performed via KNN and pooling, progressively extracting local and global features and supporting flexible integration with task-specific decoders.

2.3.5. KPConv

KPConv [20] is a DL model designed specifically for point cloud analysis, with its architecture centered around the concept of kernel points. Kernel points are a set of predefined points in 3D space that specify positions relative to each point in the point cloud, allowing the convolution operation to collect feature contributions. During the convolution process, the model examines the local neighborhood of each point in the point cloud and computes a weighted sum of the features of neighboring points on the basis of the positions of the kernel points and a correlation function (such as Gaussian or distance-based linear decay). The positions of these kernel points are learned and continuously adjusted during training, enabling the model to adapt to the specific geometric structure of the point cloud data.

2.3.6. RandLA-Net

The network structure of RandLA-Net [21] is based on efficient random sampling and local feature aggregation (LFA), designed as an encoder–decoder architecture. The encoder progressively reduces the scale of the point cloud through random sampling while employing LFA modules to extract local geometric and feature relationships. The LFA module comprises local spatial encoding and attentive pooling, which expand the receptive field to preserve complex geometric details [28]. The decoder performs upsampling via nearest-neighbor interpolation and fuses intermediate features from the encoder via skip connections, ultimately predicting the semantic label for each point through fully connected layers.

2.3.7. PointCNN

PointCNN [22] is a convolution-based neural network whose architecture is centered on an innovative X-transformation mechanism. X-transformation is a learnable function that maps the unordered points of a point cloud into a latent space, where they can be organized in a canonical order suitable for convolution. During the convolution process, for each point in the point cloud, the model leverages this transformation to align the

local features, enabling standard convolution operations—originally designed for grid-structured data—to be applied effectively on irregular point clouds. The X-transformation is optimized during training, allowing the model to adapt to the unique geometric properties of the input point cloud data. The PointCNN architecture typically comprises multiple layers that progressively extract and refine features through these transformed convolutions, making it a robust solution for tasks such as point cloud classification and segmentation by directly processing raw, unstructured data with convolutional feature extraction.

2.3.8. Sparse UNet

Sparse Unet [23,29], a convolution-based neural network model, effectively mitigates the inherent limitations of traditional convolution—specifically, sparsity loss and expansion inefficiencies—by restricting convolutional operations to occupied regions within a point cloud. The methodology commences with the voxelization of the point cloud into a discrete grid, wherein only voxels containing points are preserved as input. Sparse UNet subsequently adopts an architecture analogous to conventional convolutional neural networks, integrating downsampling, upsampling, and residual modules to facilitate robust feature extraction. By judiciously selecting an appropriate voxel size, the model ensures that its receptive field adequately encompasses local regions, thereby enabling effective feature capture even for sparse point clouds and underrepresented classes. Crucially, the sparse convolution mechanism substantially reduces computational complexity, rendering Sparse UNet highly efficient for processing large-scale point clouds.

2.4. Implementation Details

For the selection of code to reproduce the eight open-source models, most of the chosen implementations were compiled using PyTorch (version 1.12.1). For PointNet++, we selected the version compiled by Xu via PyTorch, which has undergone extensive testing and demonstrated performance equivalent to that of the original TensorFlow version [30]. Point Transformer V2 and PointMetaBase utilized the open-source code provided by the paper authors. KPConv was originally compiled in TensorFlow, so we opted for the PyTorch-compiled version by Hugues in this study [31]. Notably, for the reproduction of Swin3D and Sparse UNet, we chose the compiled versions from the Pointcept project [32]. The Pointcept is an open-source point cloud DL framework that includes numerous models. Although the PyTorch-compiled versions of PointCNN and RandLA-Net are available, because of insufficient project maintenance, the PyTorch versions exhibit inconsistent performance compared with the original code in certain task scenarios. Therefore, we chose to reproduce these two models via their original TensorFlow-based code.

3. Results

3.1. Experiment Setup

To mitigate the impact of overfitting on the training results, experiments were conducted separately on the validation and test sets. Training was performed using a GeForce RTX 2060 (6 GB); the PyTorch implementations were reproduced in an Ubuntu 20.04 environment, while the TensorFlow implementations were reproduced in an Ubuntu 18.04 environment. The batch size was set to 2, and the number of epochs was determined based on each model's convergence behavior.

3.2. Evaluation Metrics

We used overall accuracy (OA), IoU, and mIoU to evaluate the segmentation performance of the models. OA represents the proportion of correctly segmented points to the total number of points in the dataset [33]. However, when the class distribution in the

dataset is imbalanced, leading to excessively low or high segmentation precision, OA fails to accurately reflect the impact of the accuracy of these classes on the overall accuracy. To address these issues, the IoU (intersection over union) was employed [34]. IoU compensates for differences in class frequencies and indicates the overlap between the predicted point cloud and the ground truth point cloud for a specific class. mIoU (mean intersection over union) is calculated by averaging the IoU across all classes. These metrics are defined as follows:

$$OA = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

$$IoU_i = \frac{TP_i}{TP_i + FP_i + FN_i} \quad (2)$$

$$mIoU = \sum_{i=1}^{n} \left(\frac{TP_i}{TP_i + FP_i + FN_i}\right)/n \quad (3)$$

where TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively; i represents a specific class; and n is the number of classes, which in our case is four.

3.3. Semantic Segmentation Results of the Models

Based on the results in Tables 4 and 5, all the models achieved relatively high IoU scores in the semantic segmentation of the "leaf" class. Most models also obtained acceptable IoU scores for the "stem" class. However, only Sparse UNet and PointMetaBase performed well in segmenting the "flower" and "berry" classes. From an intuitive perspective, this can be attributed to the fact that point cloud data labeled "leaf" is the most abundant in the dataset, accounting for approximately 85% of the total point cloud. Consequently, models tend to prioritize learning the features of the "leaf" class. The "stem" class constitutes 9% of the point cloud, whereas the "flower" and "berry" classes account for only 4% and 3%, respectively. Owing to the extremely small proportion of point cloud data for these classes, models are prone to overlooking them. Therefore, in the strawberry semantic segmentation task, the ability to detect and extract features of these rare classes significantly impacts the final semantic segmentation results. The segmentation results are shown in Figure 3.

Table 4. OA, mIoU, and the IoU for each organ of the eight selected models on the validation dataset are expressed as percentages. The numbers in bold indicate that the item is the highest in the corresponding metric column.

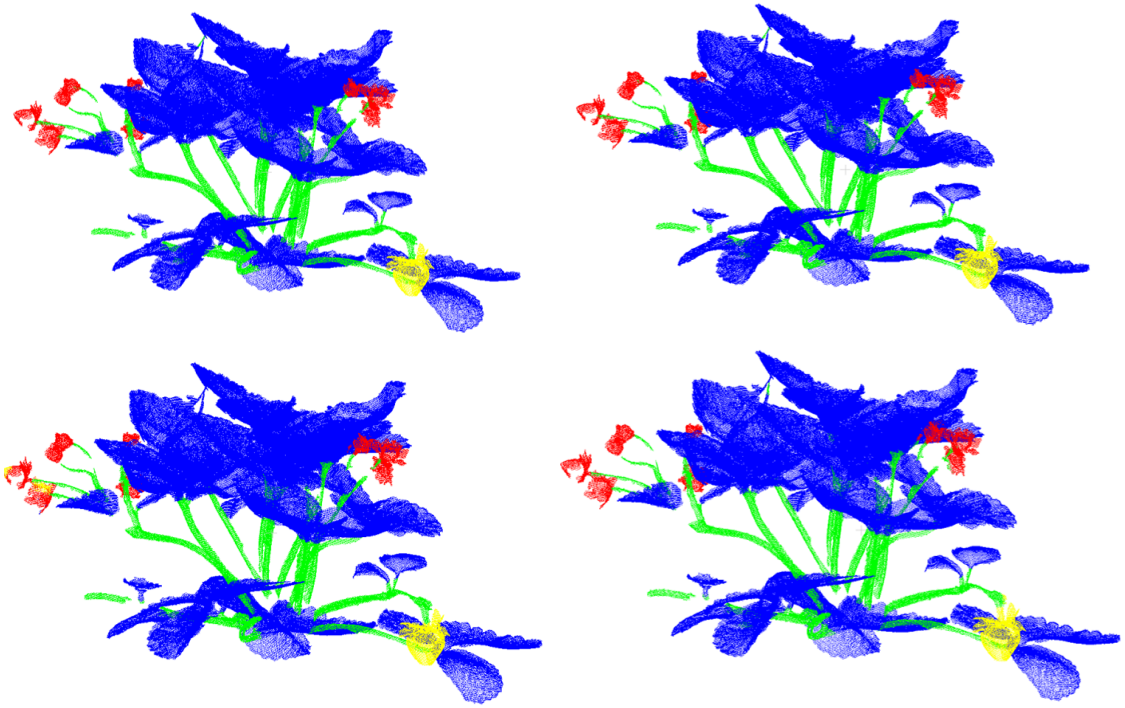
Model Name	OA (%)	mIoU (%)	Leaf IoU (%)	Stem IoU (%)	Flower IoU (%)	Berry IoU (%)
PointNet++	94.03	57.43	94.27	71.33	37.36	26.74
PointMetaBase	**98.85**	80.73	**98.72**	**90.90**	65.04	**68.26**
Point Transformer V2	96.32	68.05	96.46	85.06	58.54	32.13
Swin3D	97.41	71.46	98.51	89.42	59.03	38.87
KPConv	96.33	71.68	97.41	79.34	51.72	58.24
RandLA-Net	89.94	46.64	94.88	61.89	24.98	4.82
PointCNN	95.21	46.45	95.96	68.53	12.02	9.27
Sparse UNet	98.04	**81.31**	98.24	87.14	**78.06**	61.81

Table 5. OA, mIoU, and the IoU for each organ of the eight selected models on the test dataset are expressed as percentages. The numbers in bold indicate that the item is the highest in the corresponding metric column.

Model Name	OA (%)	mIoU (%)	Leaf IoU (%)	Stem IoU (%)	Flower IoU (%)	Berry IoU (%)
PointNet++	94.64	68.26	95.81	66.83	64.67	45.72
PointMetaBase	**98.48**	**83.62**	98.34	**91.08**	**73.12**	71.93

Table 5. *Cont.*

Model Name	OA (%)	mIoU (%)	Leaf IoU (%)	Stem IoU (%)	Flower IoU (%)	Berry IoU (%)
Point Transformer V2	95.41	62.74	96.87	86.34	45.57	22.18
Swin3D	95.12	76.60	95.23	76.86	60.73	73.59
KPConv	94.28	68.04	96.38	74.93	44.62	56.23
RandLA-Net	89.04	43.08	95.01	56.91	20.18	0.23
PointCNN	92.56	54.42	92.58	79.38	8.10	37.62
Sparse UNet	97.49	82.84	**98.63**	87.79	72.15	**72.78**



(**a**) (**b**)

Figure 3. Segmentation visualization results of Sparse UNet and PointMetaBase. Red indicates the "flower" label; yellow indicates the "berry" label; green indicates the "stem" label; and blue indicates the "leaf" label. (**a**) At the top is the ground truth result, and below is the prediction result of Sparse UNet; (**b**) at the top is the ground truth result, and below is the prediction result of PointMetaBase.

4. Discussion

4.1. Semantic Segmentation Results of the Models

4.1.1. PointNet++

As pioneering work among point-based methods, the PointNet++ [16] approach of directly extracting features from points has influenced numerous DL methods. However, in the current task, its farthest point sampling (FPS) combined with the KNN feature aggregation method performs poorly in extracting features from rare classes. Owing to the KNN computation, features of points in rare classes may be overwhelmed by features of points in more dominant classes. FPS (and its uniform sampling variants) ignores semantic distribution, leading to insufficient sampling in boundary or rare-class regions—a behavior also confirmed by other studies [35]. Moreover, PointNet++ does not incorporate an additional positional encoding embedding module for the point cloud, causing the XYZ coordinates of the point cloud to be aggregated together with the point cloud features. This results in the positional information of rare classes being difficult to preserve during the feature aggregation process.

4.1.2. PointMetaBase

PointMetaBase [17] achieved the second-best performance in this experiment, with IoU results of 65 and 68 for the "flower" and "berry" classes in validation dataset, respectively. As a point-based method, the most significant distinction between PointMetaBase and PointNet++ lies in its incorporation of an explicit positional embedding module. Through a trainable MLP network, positional information is more effectively embedded into the aggregated features, which are subsequently fed into a max pooling module. Additionally, PointMetaBase employs an "MLP-before-Group" strategy, which reduces computational complexity and accelerates training speed. Moreover, PointMetaBase replaces the vector attention mechanism with max pooling, significantly reducing the number of parameters without compromising training effectiveness. This mechanism collaborates with the neighbor update function, providing efficient aggregation capabilities. The selection and application of PointMetaBase's structural modules represent the optimal combination determined by the original authors after multiple ablation experiments, and its performance convincingly demonstrates the effectiveness of its modular design [17].

4.1.3. Point Transformer V2

Point Transformer V2 (PTV2) [18], as a transformer-based model, employs a vector attention mechanism to extract point cloud features. Building on PTV1, PTV2 enhances the standard global vector attention mechanism with a GVA approach and incorporates explicit positional encoding to strengthen the perception of local structures. However, despite the strength of attention mechanisms in capturing long-range dependencies within point clouds, they exhibit significant limitations in extracting features from rare classes [36–38]. This limitation stems from PTV2's lack of mechanisms to prioritize or amplify the fine-grained patterns of rare classes. Without such localized feature extraction capabilities, the self-attention mechanism tends to homogenize feature representations, favoring the dominant patterns of majority classes while overlooking the sparse local structures of rare classes.

4.1.4. Swin3D

The Swin3D model [19], currently a SOTA performer on certain public datasets, did not achieve outstanding results in this task. It combines sparse convolutional networks with self-attention, first voxelizing the point cloud and then performing attention operations on the sparse voxel features. Although this hybrid design partially mitigates sparsity issues by operating on voxelized regions, its self-attention component struggles to adequately emphasize the local geometric characteristics of rare classes [36–38]. Even when constrained within sparse voxels, the attention mechanism distributes focus across the entire voxelized space, failing to sufficiently highlight the subtle features of rare classes overshadowed by the dominant majority class points.

4.1.5. KPConv

KPConv [20] operates directly on raw point clouds, relying on kernel points and local neighborhoods. Although this method excels at capturing complex local geometric structures, its performance is challenged when the number of points in rare classes is extremely low. Additionally, the number and selection of kernel points influence the feature extraction performance for rare classes. In such cases, local neighborhoods may be dominated by points from more common classes, thereby compromising the accuracy of rare class feature extraction [39,40].

4.1.6. RandLA-Net

The RandLA-Net [21] method performed poorly in this experiment. Originally designed for rapid semantic segmentation tasks without sacrificing accuracy, its random sampling approach clearly fails to meet the requirements for extracting features from rare classes. Although it incorporates explicit positional encoding to enhance contextual relationships in point clouds, the random sampling strategy significantly reduces the likelihood of rare class points being selected, resulting in their features being underrepresented in subsequent feature extraction modules. The original RandLA-Net paper likewise notes that its random downsampling strategy may directly discard sparse or minority-class points during processing, thereby preventing those rare class features from being fully utilized by the subsequent network [21].

4.1.7. PointCNN

The X-Conv operation in PointCNN [22] transforms unordered point clouds into a convolution-friendly structure via FPS and X-transformation. However, when processing data with sparse and unevenly distributed rare classes, such as strawberry point clouds, it may fail because of FPS's tendency to select uniformly distributed points, potentially overlooking rare class points, or because of X-transformation's weighting and reordering process (which generates a transformation matrix via MLP from local coordinates), assigning lower weights to rare class features and possibly suppressing them, causing these features to be "overwhelmed" or blurred by more common classes [35]. Although PointCNN employs an encoder–decoder structure to integrate global information, if rare class features are weakened in early layers, subsequent layers may struggle to recover them, thus diminishing the model's ability to extract features from rare classes [41]. This issue may be exacerbated when the geometric characteristics of the dataset deviate from the model's assumption, highlighting a lack of sufficient robustness.

4.1.8. Sparse UNet

Sparse UNet [23] achieved the best results in this experiment, with IoU scores for the "flower" and "berry" classes exceeding 78 and 61. Despite its simple structure, Sparse UNet effectively retains all possible features by performing robust convolutional feature extraction on voxel blocks containing existing point clouds. Combined with downsampling, upsampling, and residual modules and aided by appropriately sized voxels, Sparse UNet ensures that its receptive field covers sufficient local regions, thereby effectively capturing features from sparse point clouds and rare classes [42].

4.2. Recommendations on Suitable Architectures and Future Research Directions

Based on the benchmarking results presented in this study, we identified certain characteristics of neural network architectures that are particularly beneficial for semantic segmentation tasks involving strawberry organs, especially considering the extreme imbalance of classes such as flowers (4%) and berries (3%). Among the evaluated models, Sparse UNet and PointMetaBase demonstrated superior capabilities in capturing fine-grained features from rare classes, owing to their efficient convolutional operations, explicit positional encoding modules, and robust feature aggregation strategies. In contrast, models utilizing traditional uniform sampling methods, such as PointCNN and RandLA-Net, exhibited limitations in effectively addressing class imbalance, leading to suboptimal segmentation performance for minority classes.

Our experimental findings suggest that architectures incorporating the following components tend to perform better in scenarios with imbalanced point cloud data such as the strawberry organ semantic segmentation task:

- Adaptive sampling methods, which ensure adequate representation of minority classes during feature extraction.
- Explicit positional encoding, as demonstrated in PointMetaBase, enhancing the model's sensitivity to subtle spatial structures.
- Sparse convolutional operations, as shown in Sparse UNet, which effectively handle sparse data distributions by preserving critical local geometric information.

It is important to note, however, that these recommendations arise strictly from comparative experimental observations rather than from direct structural modifications or ablation tests within this study.

In our future research, we intend to explicitly address the challenges identified here by developing novel architectures specifically optimized for class-imbalanced scenarios commonly encountered in plant organ segmentation tasks. These future developments will incorporate comprehensive ablation studies to rigorously validate the contribution of each proposed module or strategy. The insights and comparative baseline results provided by the current benchmark study will serve as a solid foundation for these subsequent explorations, guiding systematic architectural improvements and methodological innovations.

Author Contributions: Conceptualization and methodology: R.X. and H.N.; analysis, validation, and visualization: R.X., H.N. and F.H.; writing—original draft preparation: R.X.; review and editing: R.X., H.N. and F.H.; supervision: H.N. and F.H.; funding acquisition: H.N. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by JSPS (Japan Society for the Promotion of Science) KAKENHI Grant Number JP24K09159.

Data Availability Statement: The LAST-Straw dataset presented in this study is openly available in [https://lcas.github.io/LAST-Straw/] (accessed on 19 October 2024); the Japanese cultivar dataset presented in this study are available on request from the corresponding author due to the limited amount of labeled data, which restricts its readiness for open sharing.

Acknowledgments: The variety Fukuba was provided by the National Agriculture and Food Research Organization (NARO) Gene Bank Project. We would like to express our gratitude to Masahiro Misumi of the Kyushu Okinawa Agricultural Research Center, NARO, for his advice on selecting the variety.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hernández-Martínez, N.R.; Blanchard, C.; Wells, D.; Salazar-Gutiérrez, M.R. Current State and Future Perspectives of Commercial Strawberry Production: A Review. *Sci. Hortic.* **2023**, *312*, 111893. [CrossRef]
2. Rosado-Souza, L.; Yokoyama, R.; Sonnewald, U.; Fernie, A.R. Understanding Source–Sink Interactions: Progress in Model Plants and Translational Research to Crops. *Mol. Plant* **2023**, *16*, 96–121. [CrossRef] [PubMed]
3. Hidaka, K.; Miyoshi, Y.; Ishii, S.; Suzui, N.; Yin, Y.-G.; Kurita, K.; Nagao, K.; Araki, T.; Yasutake, D.; Kitano, M.; et al. Dynamic Analysis of Photosynthate Translocation into Strawberry Fruits Using Non-Invasive 11C-Labeling Supported with Conventional Destructive Measurements Using 13C-Labeling. *Front. Plant Sci.* **2019**, *9*, 1946. [CrossRef] [PubMed]
4. Frontiers | Source-Sink Relationships in Crop Plants and Their Influence on Yield Development and Nutritional Quality. Available online: https://www.frontiersin.org/journals/plant-science/articles/10.3389/fpls.2018.01889/full?utm_source=chatgpt.com (accessed on 14 May 2025).
5. Rafiqui, A.R.; Ayoub, O.B. Elucidation of Source-Sink Relationship in Mulberry (*Morus* spp.). *Indian J. Pure Appl. Biosci.* **2024**, *12*, 9–19. [CrossRef]
6. Hansen, P. Source-Sink Relations in Fruits Iv. Fruit Number and Fruit Growth in Strawberries. *Acta Hortic.* **1989**, *265*, 377–382. [CrossRef]
7. Nakai, H.; Yasutake, D.; Hidaka, K.; Miyoshi, Y.; Eguchi, T.; Yokoyama, G.; Hirota, T. Sink Strength Dynamics Based on Potential Growth and Carbohydrate Accumulation in Strawberry Fruit. *HortScience* **2024**, *59*, 1505–1510. [CrossRef]
8. Daviet, B.; Fernandez, R.; Cabrera-Bosquet, L.; Pradal, C.; Fournier, C. PhenoTrack3D: An Automatic High-Throughput Phenotyping Pipeline to Track Maize Organs over Time. *Plant Methods* **2022**, *18*, 130. [CrossRef]

9. Liu, F.-Y.; Geng, H.; Shang, L.-Y.; Si, C.-J.; Shen, S.-Q. A Cotton Organ Segmentation Method with Phenotypic Measurements from a Point Cloud Using a Transformer. *Plant Methods* **2025**, *21*, 37. [CrossRef]
10. Armeni, I.; Sax, S.; Zamir, A.R.; Savarese, S. Joint 2D-3D-Semantic Data for Indoor Scene Understanding. *arXiv* **2017**, arXiv:1702.01105.
11. Dai, A.; Chang, A.X.; Savva, M.; Halber, M.; Funkhouser, T.; Nießner, M. ScanNet: Richly-Annotated 3D Reconstructions of Indoor Scenes. *arXiv* **2017**, arXiv:1702.04405.
12. Fujinaga, T. Strawberries Recognition and Cutting Point Detection for Fruit Harvesting and Truss Pruning. *Precis. Agric.* **2024**, *25*, 1262–1283. [CrossRef]
13. Le Louëdec, J.; Cielniak, G. 3D Shape Sensing and Deep Learning-Based Segmentation of Strawberries. *Comput. Electron. Agric.* **2021**, *190*, 106374. [CrossRef]
14. Patel, A.K.; Park, E.-S.; Lee, H.; Priya, G.G.L.; Kim, H.; Joshi, R.; Arief, M.A.A.; Kim, M.S.; Baek, I.; Cho, B.-K. Deep Learning-Based Plant Organ Segmentation and Phenotyping of Sorghum Plants Using LiDAR Point Cloud. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2023**, *16*, 8492–8507. [CrossRef]
15. Yang, X.; Miao, T.; Tian, X.; Wang, D.; Zhao, J.; Lin, L.; Zhu, C.; Yang, T.; Xu, T. Maize Stem–Leaf Segmentation Framework Based on Deformable Point Clouds. *ISPRS J. Photogramm. Remote Sens.* **2024**, *211*, 49–66. [CrossRef]
16. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Curran Associates, Inc.: Red Hook, NY, USA, 2017; Volume 30.
17. Lin, H.; Zheng, X.; Li, L.; Chao, F.; Wang, S.; Wang, Y.; Tian, Y.; Ji, R. Meta Architecture for Point Cloud Analysis. *arXiv* **2023**, arXiv:2211.14462.
18. Wu, X.; Lao, Y.; Jiang, L.; Liu, X.; Zhao, H. Point Transformer V2: Grouped Vector Attention and Partition-Based Pooling. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 33330–33342.
19. Yang, Y.-Q.; Guo, Y.-X.; Xiong, J.-Y.; Liu, Y.; Pan, H.; Wang, P.-S.; Tong, X.; Guo, B. Swin3D: A Pretrained Transformer Backbone for 3D Indoor Scene Understanding. *Comput. Vis. Media* **2025**, *11*, 83–101. [CrossRef]
20. Thomas, H.; Qi, C.R.; Deschaud, J.-E.; Marcotegui, B.; Goulette, F.; Guibas, L.J. KPConv: Flexible and Deformable Convolution for Point Clouds. *arXiv* **2019**, arXiv:1904.08889.
21. Hu, Q.; Yang, B.; Xie, L.; Rosa, S.; Guo, Y.; Wang, Z.; Trigoni, N.; Markham, A. RandLA-Net: Efficient Semantic Segmentation of Large-Scale Point Clouds. *arXiv* **2020**, arXiv:1911.11236.
22. Li, Y.; Bu, R.; Sun, M.; Wu, W.; Di, X.; Chen, B. PointCNN: Convolution On X-Transformed Points. In Proceedings of the Advances in Neural Information Processing Systems, Montréal, QC, Canada, 3–8 December 2018; Curran Associates, Inc.: Red Hook, NY, USA, 2018; Volume 31.
23. Yan, Y.; Mao, Y.; Li, B. SECOND: Sparsely Embedded Convolutional Detection. *Sensors* **2018**, *18*, 3337. [CrossRef]
24. James, K.M.F.; Heiwolt, K.; Sargent, D.J.; Cielniak, G. Lincoln's Annotated Spatio-Temporal Strawberry Dataset (LAST-Straw). *arXiv* **2024**, arXiv:2403.00566.
25. Kerbl, B.; Kopanas, G.; Leimkuehler, T.; Drettakis, G. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Trans. Graph.* **2023**, *42*, 1–14. [CrossRef]
26. Qi, C.R.; Su, H.; Mo, K.; Guibas, L.J. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. *arXiv* **2017**, arXiv:1612.00593.
27. Shaw, P.; Uszkoreit, J.; Vaswani, A. Self-Attention with Relative Position Representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, New Orleans, LA, USA, 1–6 June 2018; Walker, M., Ji, H., Stent, A., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; Volume 2, pp. 464–468.
28. Yang, B.; Wang, S.; Markham, A.; Trigoni, N. Robust Attentional Aggregation of Deep Feature Sets for Multi-View 3D Reconstruction. *Int. J. Comput. Vis.* **2020**, *128*, 53–73. [CrossRef]
29. Graham, B.; van der Maaten, L. Submanifold Sparse Convolutional Networks. *arXiv* **2017**, arXiv:1706.01307.
30. Pointnet_Pointnet2_pytorch. Available online: https://github.com/yanx27/Pointnet_Pointnet2_pytorch (accessed on 1 April 2025).
31. KPConv-PyTorch. Available online: https://github.com/HuguesTHOMAS/KPConv (accessed on 1 April 2025).
32. Pointcept. Available online: https://github.com/Pointcept/Pointcept (accessed on 1 April 2025).
33. Alberg, A.J.; Park, J.W.; Hager, B.W.; Brock, M.V.; Diener-West, M. The Use of "Overall Accuracy" to Evaluate the Validity of Screening or Diagnostic Tests. *J. Gen. Intern. Med.* **2004**, *19*, 460–465. [CrossRef] [PubMed]
34. Everingham, M.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
35. Task-Aware Sampling Layer for Point-Wise Analysis. Available online: https://ieeexplore.ieee.org/document/9767766 (accessed on 14 May 2025).

36. Xiu, H. Enhancing Local Feature Learning for 3D Point Cloud Processing Using Unary-Pairwise Attention. *arXiv* **2022**, arXiv:2203.00172.
37. Xu, Z.; Liu, R.; Yang, S.; Chai, Z.; Yuan, C. Learning Imbalanced Data With Vision Transformers. *arXiv* **2023**, arXiv:2212.02015.
38. Li, K.; Duggal, R.; Chau, D.H. Evaluating Robustness of Vision Transformers on Imbalanced Datasets. In Proceedings of the 37th AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2023.
39. Chen, R.; Wu, J.; Luo, Y.; Xu, G. PointMM: Point Cloud Semantic Segmentation CNN under Multi-Spatial Feature Encoding and Multi-Head Attention Pooling. *Remote Sens.* **2024**, *16*, 1246. [CrossRef]
40. Alawieh, A.; Condurache, A.P. FA-KPConv: Introducing Euclidean Symmetries to KPConv via Frame Averaging. *arXiv* **2025**, arXiv:2505.04485.
41. Lumban-Gaol, Y.A.; Chen, Z.; Smit, M.; Li, X.; Erbaşu, M.A.; Verbree, E.; Balado, J.; Meijers, M.; Van Der Vaart, N. A Comparative Study of Point Clouds Semantic Segmentation Using Three Different Neural Networks on the Railway Station Dataset. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2021**, *XLIII-B3-2021*, 223–228. [CrossRef]
42. Zhang, X.; Lin, D.; Xue, R.; Soergel, U. Target-Guided Learning for Rare Class Segmentation in Large-Scale Urban Point Clouds. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2023**, *XLVIII-1-W2-2023*, 1693–1698. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.