

## Unsupervised shape-aware SOM down-sampling for plant point clouds

Dawei Li<sup>a,b,c,\*<sup>1</sup></sup>, Zhaoyi Zhou<sup>b,1</sup>, Yongchang Wei<sup>b</sup>

<sup>a</sup> Engineering Research Center of Digitized Textile, Fashion Technology, Ministry of Education, Donghua University, Shanghai 201620, China

<sup>b</sup> College of Information Sciences and Technology, Donghua University, Shanghai 201620, China

<sup>c</sup> State Key Laboratory for Modification of Chemical Fibers and Polymer Materials, College of Information Sciences and Technology, Donghua University, Shanghai 201620, China



### ARTICLE INFO

**Keywords:**  
Sampling  
Self-organizing Map  
Plant Phenotyping  
Point Cloud Data  
Organ Segmentation

### ABSTRACT

Observation of the external 3D shape/structure and some measurable phenotypic traits is of great significance to screening excellent varieties and improving crop yield in agriculture. The dense crop point clouds scanned by 3D sensors not only may include imaging noise, but also contain a large number of redundant points that will put high burden on storage and slow down the speed of algorithm for point cloud segmentation, classification, and other following processing steps. To reduce the complexity of point cloud data and meanwhile better represent the structure under limited resources, this paper presents a new Self-organizing Map (SOM)-based down-sampling strategy that is tailored for plant (or plant-like) point clouds. Our SOM-based sampling works in a purely unsupervised manner and precisely controls the number of points after down-sampling. It obtains shape-aware sampling on irregular plant point clouds by automatically encoding preliminary semantics to different organ types (e.g., stems are sampled as “lines”, and leaves are sampled as folded curves shaped in “surfaces”). Extensive experiments on a multi-species plant dataset were conducted using several popular deep 3D-segmentation networks as the downstream task unit, respectively. The segmentation performance of the SOM-processed dataset outperformed several other mainstream down-sampling strategies. Our SOM strategy with 1D neuron layer can be further generalized to 2D and 3D versions, and also can be extended to a more adaptive framework that automatically picks the most suitable version of SOM for each corresponding local shape component. The proposed strategy also showed good potential in serving different applications including point cloud skeleton extraction, crop main stem length measurement; and presented satisfactory results on point cloud datasets from other domains, indicating its high applicability and good data domain adaptation.

### 1. Introduction

Plants play an important role in agriculture, food industry, forestry, environmental protection, energy, and many other aspects associated with human life and social development (Cappelli et al., 2022; Zandalinas et al., 2018; Lin et al., 2020; Sindelar et al., 2000). Observation of the external shape/structure and some measurable traits (e.g., leaf number, plant height, leaf area index, average fruit size, fruit number, etc.), which are often called plant phenotypes, is of great significance to screening excellent varieties and improving crop yield in agriculture. Due to the fast developing sensing and computer vision technology, we are stepping from the stage of using 2D images for description and analysis to the new era of 3D-based sensing and automated phenotyping on 3D data. The 3D sensing method such as Lidar-scanning can generate

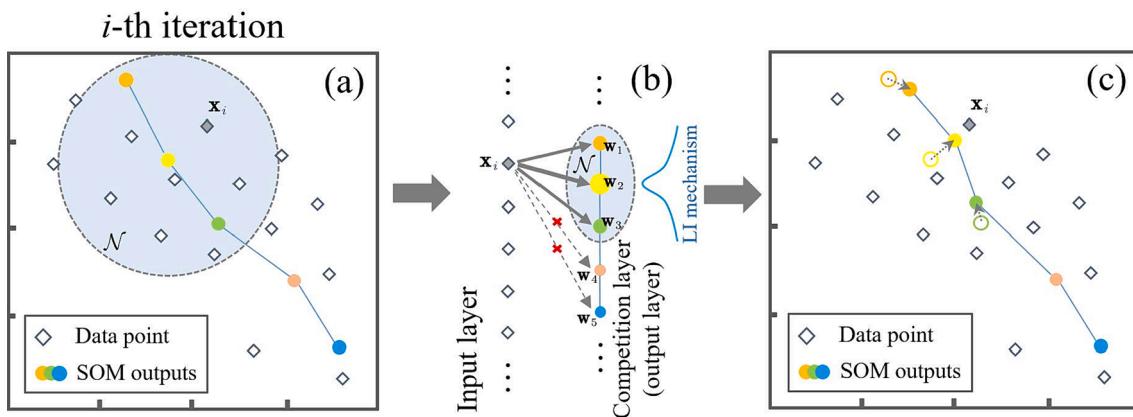
an accurate and all-round point cloud for a crop sample, e.g., a Lidar-scanned point cloud of a plant with a height of less than 0.5 m can contain nearly 1 million points. The dense point clouds not only include imaging noise, but also contain a large number of redundant points that will put high burden on storage and slow down the speed of algorithm for point cloud segmentation, classification, and other following processing steps. Therefore, using a down-sampling strategy to pre-process plant point clouds has become an essential and even unanimous step across almost all existing 3D phenotyping systems.

The down-sampling of point clouds has always been a fundamental topic in 3D graphics as it helps reduce computational and storage requirements for point cloud processing. In the era of 3D deep learning, training frameworks often require that all point clouds for training should possess an equal number of points and similar spatial scales (Guo

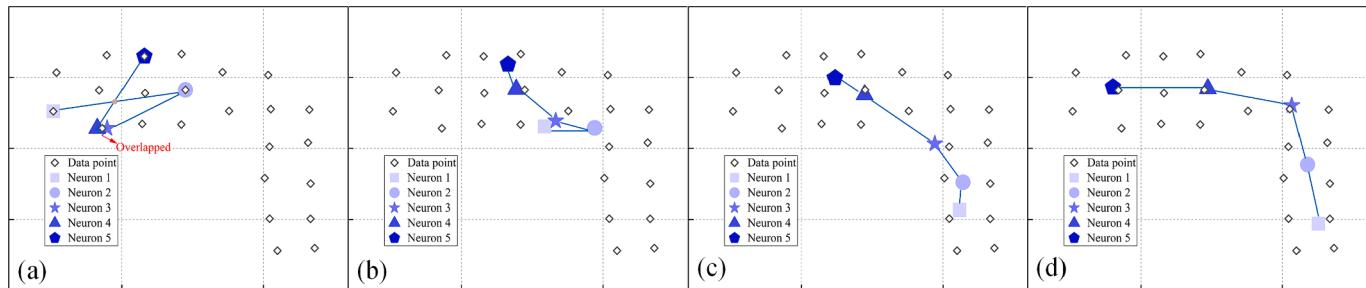
\* Corresponding author.

E-mail address: [daweili@dhu.edu.cn](mailto:daweili@dhu.edu.cn) (D. Li).

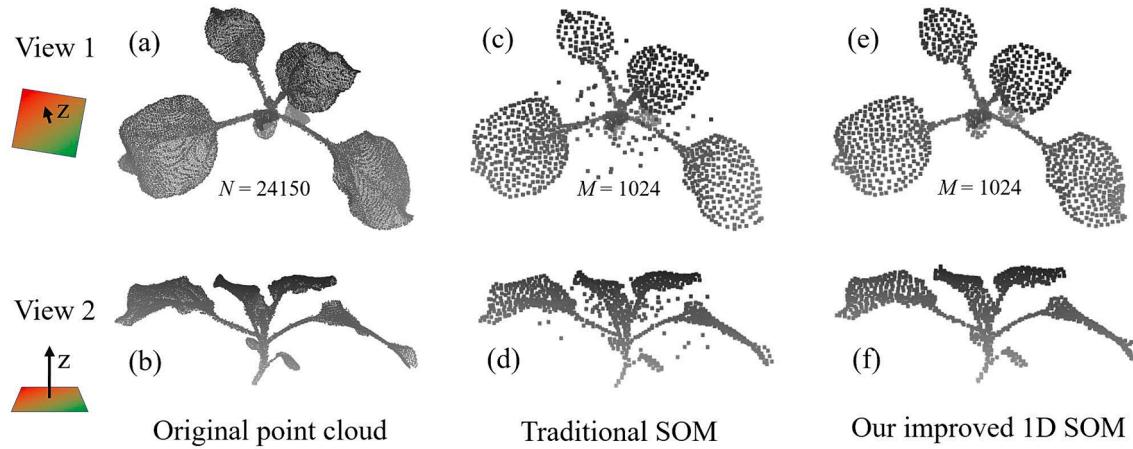
<sup>1</sup> These authors contributed equally to this work.



**Fig. 1.** Illustration for the  $i$ -th iteration of competitive learning for down-sampling a 2D point set using SOM. (a) illustrates the initial distribution of the input point set before the  $i$ -th iteration and the position (weights) of the neurons from the competition (output) layer; (b) is a schematic of the  $i$ -th competitive learning manner (Lateral Inhibition) of the one-dimensionally arranged SOM neurons; (c) illustrates the changes in the weights of the neurons of the competition layer after the  $i$ -th iteration. Only the nodes within the activation field  $\mathcal{N}$  are updated, and the movements are labeled with dotted arrows.



**Fig. 2.** The iterative comparative learning process of traditional 1D SOM on a 2D lattice. The number of points in the point set is  $N = 24$ , and the number of points after down-sampling is fixed to  $M = 5$ . (a) shows the randomly initialized neuron weights, with neurons No. 3 and No. 4 almost overlapped; and the neighborhood connections are also intersected, creating a relatively complicated initialization scenario. (b-c) correspond to the learning results after 8 and 48 iterations, respectively. (d) shows the final weights after 120 iterations. In 1D SOM, each neuron is linked to its adjacent left and right neighbors (and the nodes are incrementally indexed along the 1D layer when initialized), while the two terminal nodes have only one neighbor.



**Fig. 3.** Visualization of the outlier problem of the traditional SOM for sampling. (a) and (b) are an original crop point cloud (containing 24,150 points) visualized from two different views, respectively. (c) and (d) show the sampling result by the traditional SOM at 1024 points from two views, respectively. (e) and (f) show the sampling result by the proposed strategy in this subsection at 1024 points from two views, respectively. The iteration numbers of the two contrasted down-sampling strategies are the same.

et al., 2021b). Therefore, down-sampling algorithms are commonly employed to prepare regular datasets for point cloud deep learning networks. During long time of practice, several key features that an ideal down-sampling strategy should possess have been identified. First, it is suggested to be unsupervised, which can make it easier to adapt well to

different data sources without relying on certain particular prior knowledge or the need for domain adaptation or training (Lugmayr et al., 2019). Second, it should be computationally efficient, creating applicability on low-cost and portable platforms. Third, a certain degree of randomness is favored for facilitating the augmentation of training

**Table 1**

The pseudocode of improved 1D SOM-based sampling algorithm.

**Algorithm 1: Improved 1D SOM for point set down-sampling**

**Input:** Data set with  $N$  points  $\mathcal{X} = \{x_1, \dots, x_N\}$ , Number of points after down-sampling  $M$ .

**Output:** Neuron weight set  $\mathcal{W}$ .

**Initialization:** Randomly select  $M$  points from  $\mathcal{X}$  and sort these points along the direction of the largest PCA component of  $\mathcal{X}$ , and build a 1D neuron set  $\mathcal{W} = \{w_1, \dots, w_M\}$  whose element is initialized as  $w_j^{(1)} \leftarrow x_j | j \in [1, \dots, M]$ .

Maximum iteration number  $MIN = 5N$ .

$\lambda^{(1)} = 0.5$ ; activation radius  $\sigma^{(1)} \in [1, 1.5]$ ;  $t = 1$ .

Inlier neuron set  $\mathcal{I} = \emptyset$ ; outlier neuron set  $\mathcal{O} = \emptyset$ .

**I. Competitive learning:**

1 **While**  $t < MIN$  **do**

2   Draw a point  $x_t$  from  $\mathcal{X}$ .

3   **For**  $j \in [1, \dots, M]$  **do**

4     Find winning neuron  $w_{j^*}^{(t)}$  with  $j^* = \operatorname{argmin}_j \|x_t - w_j^{(t)}\|$

5   **End for**

6   **For**  $j \in [1, \dots, M]$  **do**

7     Compute activation function  $a_j^{(t)}$  and neuron weight  $w_j^{(t)}$ :

8      $a_j^{(t)} = \exp\left[-(j - j^*)^2 / (2\sigma^{(t)} \cdot \sigma^{(t)})\right]$ ,

9      $w_j^{(t+1)} \leftarrow w_j^{(t)} + \lambda^{(t)} a_j^{(t)} (x_t - w_{j^*}^{(t)})$ .

10   **End for**

11    $\lambda^{(t+1)} \leftarrow \lambda^{(t)} / (1 + 2t/MIN)$ ;

12    $\sigma^{(t+1)} \leftarrow \sigma^{(t)} / (1 + 2t/MIN)$ ;

13    $t \leftarrow t + 1$ ;

14 **End while**

**II. Refinement:**

15 **For**  $i \in [1, \dots, N]$  **do**

16   Find the nearest  $w_j \in \mathcal{W}$  to  $x_i$ ,

17   **If**  $w_j \notin \mathcal{I}$

18     Push  $w_j$  into  $\mathcal{I}$ .

19   **End if**

20 **End for**

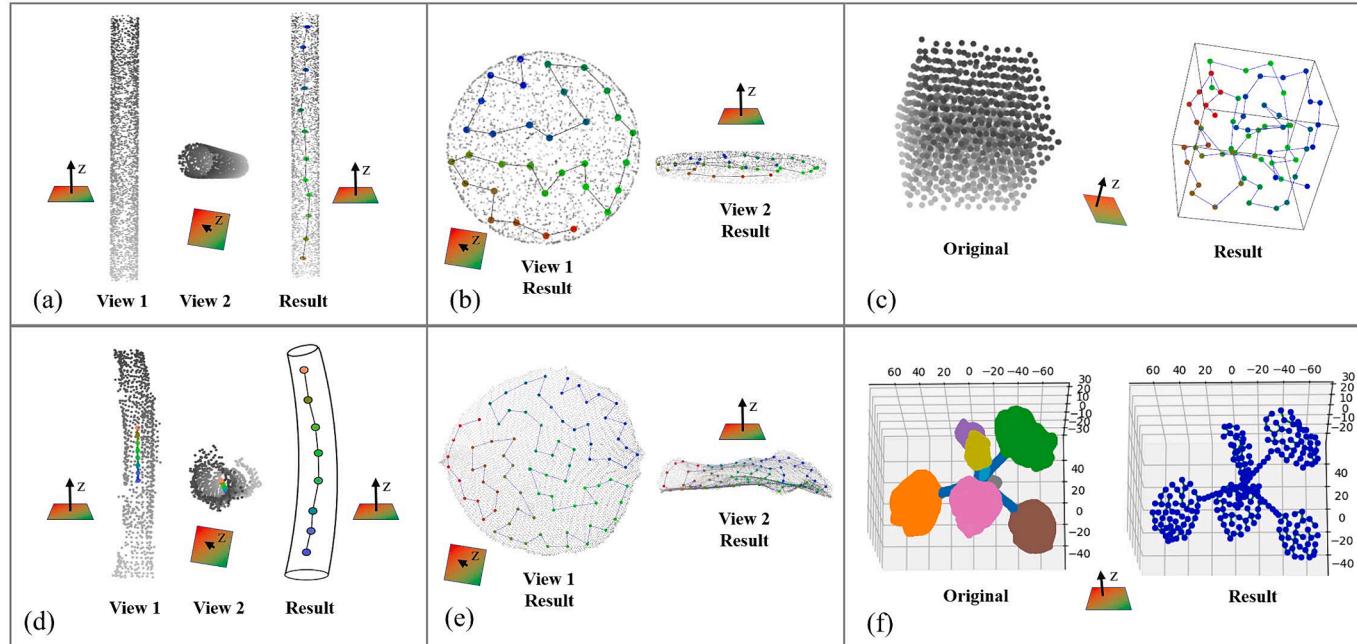
21  $\mathcal{O} \leftarrow \mathcal{W} / \mathcal{I}$ .

22 **For all**  $w \in \mathcal{O}$  **do**

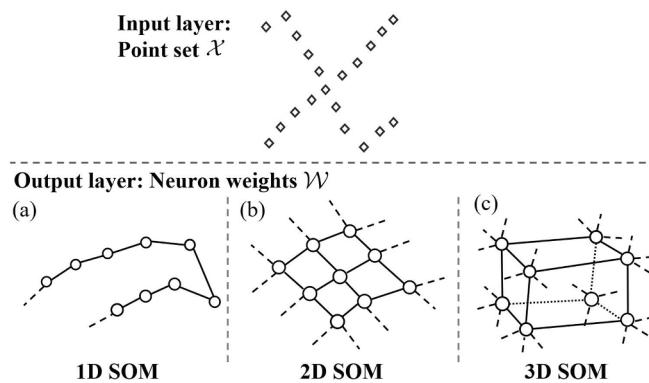
23    $w \leftarrow x_{i^*} | i^* = \operatorname{argmin}_i \|x_i - w\|$ .

24 **End for**

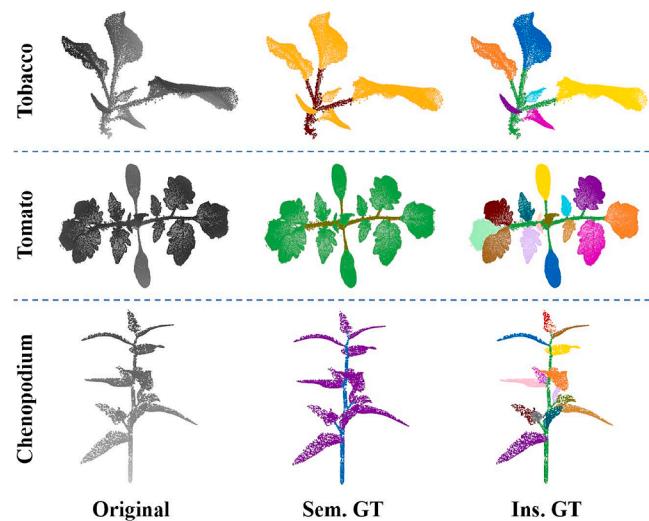
25  $\mathcal{W} \leftarrow \mathcal{O} \cup \mathcal{I}$ .



**Fig. 4.** Demonstration of the shape-aware attribute of our improved 1D SOM. (a) Down-sampling of a pipe-like point cloud using Algorithm 1 to 12 points; (b) Down-sampling of point cloud of a thin plate to 32 points by Algorithm 1; (c) Down-sampling of a cube-like point cloud by Algorithm 1; (d) Down-sampling of a segment of tobacco main stem to 8 points by Algorithm 1; (e) Sampled representation of a tobacco leaf to 81 points using Algorithm 1; (f) Exploiting Algorithm 1 to sample the entire tobacco crop point cloud to 256 points, which well preserves the overall plant structure.



**Fig. 5.** Generalization of SOM competition layers. (a) shows a randomly initialization of a standard 1D SOM competition layer, in which each neuron has a left and a right node neighbor; (b) is an initialization of the 2D SOM competition layer, where each interior neuron has 4 neighbors; (c) is an instance of the 3D SOM competition layer, where the nearest neighbors of each interior neuron has 6 neighboring nodes, and 8 nodes form a voxel structure.



**Fig. 6.** Examples from the crop point cloud dataset in this paper. The 1st row shows a sample of tobacco point cloud, the 2nd row shows a sample of tomato plant, and the 3rd row shows a sample of chenopodium crop. The 1st to 3rd columns are the original point cloud, the semantic Ground Truth labels rendered with different colors, and the instance Ground Truth labels rendered with different colors, respectively.

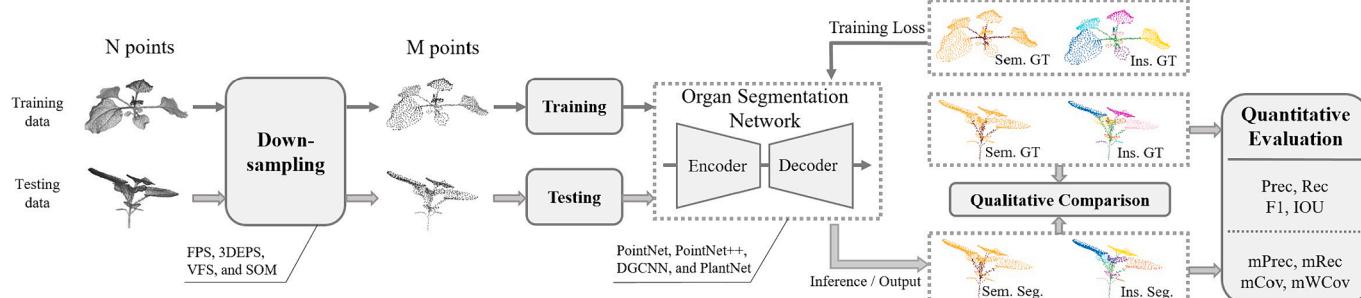
data for deep networks. These features collectively contribute to desirable down-sampling performance for modern point cloud processing.

Despite the ease of use and the high efficiency in implementation, common unsupervised down-sampling methods for point clouds often encounter challenges such as data unevenness and incapable of controlling the number of points after sampling. For instance, the widely used Farthest Point Sampling (FPS) algorithm (Eldar et al., 1994) tends to create large holes on the 3D surface for its greedy farthest-point search mechanism, especially when the down-sample rate is high. Voxel-based Sampling (VBS) (Li et al., 2022b) method can generate regular 3D point clouds but struggles with how to fix the number of points after down-sampling. To fix the point number often introduces extra computation, e.g., like tuning the voxel size or conducting an additional screening measure (similar to FPS). Furthermore, irregularly-shaped point clouds, especially the plants/crops with intricate branching structures and frequently changing surface normal (Ando et al., 2021), raise challenges for traditional down-sampling strategies. Currently, common sampling strategies are yet to achieve satisfactory results on irregular point clouds, which further deteriorates the performance of subsequent supervised learning and analysis on those point clouds.

Effective simplification and representation of crop point clouds face two imminent obstacles. The first is the structural complexity—a crop usually has highly complicated spatial structure with numerous branches and overlapping organs, requiring us to process data both locally and globally. The second is the feature complexity—each organ type differs from the others by shape, volume, and orientation; e.g., a leaf looks differently from a fruit or a branch. It is hard to keep the key features that can distinguish each organ types from the others during sampling. On the face of it, the obstacles seem to be formidable and indeed they have posed challenges to traditional sampling algorithms; however, the complexity of crop point cloud merely lies on the “surface”, i.e., the complexity is due to the 3D point distribution that lies in the low-level feature space. In fact, a human can easily tell which part of the plant is leaf, stem, or fruit because the human brain uses higher-level features to describe the 3D plant structure. For instance, when recognizing the stem segments, we subconsciously use the semantic feature that a stem usually takes 1D linear shape in space; while when recognizing the leaves, we subconsciously use the semantic feature that a leaf is usually 2D-planar in space. Therefore, we guess that if shape-related semantics can be encoded (even if implicitly) during down-sampling of plant point clouds, the performance of the downstream deep networks for 3D organ segmentation and classification can be further enhanced.

Based on the above inspiration, this paper presents improved Self-organizing Map (SOM) down-sampling that is tailored for representing plant (or plant-like) point clouds. The detailed contributions are outlined as follows:

(i) We propose an improved 1D SOM sampling algorithm that works in a purely unsupervised manner and precisely controls the number of



**Fig. 7.** Illustration of the experimental design. The down-sampling strategies are first applied to prepare the dataset, and the downstream deep networks are used to generate organ semantic segmentation and instance segmentation results. Sem. GT stands for the Ground Truth of semantic segmentation. Ins. GT is the Ground Truth for instance segmentation., while Sem. Seg. and Ins. Seg. stand for the semantic segmentation result and instance segmentation result from the networks, respectively.

**Table 2**

Quantitative comparison of our SOM to several other down-sampling strategies at different point resolutions on PointNet (organ semantic segmentation).

Sampled Resolution	Evaluation Metrics	Sampling Strategies	Tobacco		Tomato		Chenopodium		Mean	Diff
			Stem	Leaf	Stem	Leaf	Stem	Leaf		
1024 Points	F1 (%)	3DEPS	<b>74.07</b>	92.27	91.71	96.97	<b>85.33</b>	93.96	<b>89.05</b>	-0.62
		VBS	62.60	93.87	90.63	97.26	79.85	94.38	86.43	-3.24
		FPS	70.77	<b>93.93</b>	<b>91.86</b>	<b>97.52</b>	81.40	<b>94.39</b>	<b>88.31</b>	-1.36
		SOM (ours)	<b>71.30</b>	<b>94.13</b>	<b>92.33</b>	<b>97.58</b>	<b>85.34</b>	93.60	<b>89.05</b>	-0.62
	IoU (%)	3DEPS	<b>58.82</b>	85.65	84.69	94.13	74.43	88.61	<b>81.05</b>	-1.20
		VBS	45.56	88.44	82.86	94.66	66.46	<b>89.36</b>	77.89	-4.36
		FPS	54.76	<b>88.56</b>	<b>84.95</b>	<b>95.15</b>	68.63	<b>89.38</b>	80.24	-2.01
		SOM (ours)	<b>55.40</b>	<b>88.91</b>	<b>85.75</b>	<b>95.27</b>	<b>74.44</b>	87.98	<b>81.29</b>	-0.96
512 Points	F1 (%)	3DEPS	<b>73.98</b>	92.54	91.20	96.94	<b>80.04</b>	92.33	87.84	-1.83
		VBS	63.29	93.44	90.53	97.21	79.51	<b>94.41</b>	86.40	-3.27
		FPS	71.31	<b>94.42</b>	<b>91.68</b>	<b>97.53</b>	79.71	<b>94.25</b>	<b>88.15</b>	-1.52
		SOM (ours)	<b>72.96</b>	<b>94.76</b>	<b>92.71</b>	<b>97.77</b>	<b>81.78</b>	92.09	<b>88.68</b>	-0.99
	IoU (%)	3DEPS	<b>58.70</b>	86.11	83.82	94.06	<b>66.72</b>	85.75	79.19	-3.06
		VBS	46.30	87.69	82.71	94.56	65.98	<b>89.42</b>	77.78	-4.47
		FPS	55.41	<b>89.43</b>	<b>84.64</b>	<b>95.19</b>	66.27	<b>89.13</b>	<b>80.01</b>	-2.24
		SOM (ours)	<b>57.43</b>	<b>90.04</b>	<b>86.42</b>	<b>95.64</b>	<b>69.17</b>	85.33	<b>80.67</b>	-1.58
256 Points	F1 (%)	3DEPS	<b>73.31</b>	93.45	90.81	97.32	<b>81.85</b>	93.86	<b>88.43</b>	-1.24
		VBS	58.57	91.92	89.61	96.87	79.53	<b>94.53</b>	85.17	-4.50
		FPS	71.22	<b>94.43</b>	<b>91.87</b>	<b>97.73</b>	80.17	<b>94.58</b>	88.33	-1.34
		SOM (ours)	<b>72.61</b>	<b>94.52</b>	<b>92.58</b>	<b>97.80</b>	<b>85.99</b>	94.53	<b>89.67</b>	0.00
	IoU (%)	3DEPS	<b>57.87</b>	87.70	83.17	94.77	<b>69.28</b>	88.42	80.20	-2.05
		VBS	41.41	85.04	81.18	93.93	66.02	<b>89.63</b>	76.20	-6.05
		FPS	55.30	<b>89.44</b>	<b>84.97</b>	<b>95.56</b>	66.91	<b>89.71</b>	<b>80.32</b>	-1.93
		SOM (ours)	<b>57.00</b>	<b>89.60</b>	<b>86.18</b>	<b>95.70</b>	<b>75.42</b>	89.62	<b>82.25</b>	0.00

**Table 3**

Quantitative comparison of our SOM to several other down-sampling strategies at different point resolutions on PointNet++ (organ semantic segmentation).

Sampled Resolution	Evaluation Metrics	Sampling Strategies	Tobacco		Tomato		Chenopodium		Mean	Diff
			Stem	Leaf	Stem	Leaf	Stem	Leaf		
1024 Points	F1 (%)	3DEPS	<b>85.37</b>	95.46	94.29	97.86	<b>92.18</b>	97.24	93.73	-1.01
		VBS	77.36	<b>97.15</b>	93.86	98.04	91.31	<b>98.07</b>	92.63	-2.11
		FPS	<b>87.11</b>	<b>97.34</b>	<b>95.23</b>	<b>98.53</b>	92.13	<b>98.07</b>	<b>94.74</b>	0.00
		SOM (ours)	82.95	96.46	<b>94.65</b>	<b>98.28</b>	<b>92.85</b>	97.29	<b>93.75</b>	-0.99
	IoU (%)	3DEPS	<b>74.47</b>	91.32	89.21	95.80	<b>85.49</b>	94.63	88.49	-1.78
		VBS	63.08	<b>94.46</b>	88.43	96.15	84.00	<b>96.21</b>	87.06	-3.21
		FPS	<b>77.16</b>	<b>94.81</b>	<b>90.89</b>	<b>97.10</b>	85.41	<b>96.22</b>	<b>90.27</b>	0.00
		SOM (ours)	70.87	93.17	<b>89.85</b>	<b>96.62</b>	<b>86.65</b>	94.73	<b>88.65</b>	-1.62
512 Points	F1 (%)	3DEPS	82.92	95.00	93.14	97.59	84.51	94.93	91.35	-3.39
		VBS	74.41	96.35	92.31	97.69	<b>86.78</b>	<b>97.15</b>	90.78	-3.96
		FPS	<b>84.23</b>	<b>96.87</b>	<b>93.92</b>	<b>98.20</b>	85.91	<b>96.73</b>	<b>92.64</b>	-2.10
		SOM (ours)	<b>84.17</b>	<b>96.94</b>	<b>94.29</b>	<b>98.23</b>	<b>89.69</b>	96.27	<b>93.27</b>	-1.47
	IoU (%)	3DEPS	70.83	90.47	87.16	95.29	73.18	90.35	84.55	-5.72
		VBS	59.25	92.95	85.72	95.49	<b>76.65</b>	<b>94.46</b>	84.09	-6.18
		FPS	<b>72.76</b>	<b>93.92</b>	<b>88.54</b>	<b>96.47</b>	75.31	<b>93.66</b>	<b>86.78</b>	-3.49
		SOM (ours)	<b>72.67</b>	<b>94.07</b>	<b>89.20</b>	<b>96.52</b>	<b>81.31</b>	92.82	<b>87.76</b>	-2.51
256 Points	F1 (%)	3DEPS	80.18	95.20	92.28	97.75	<b>83.49</b>	95.09	90.66	-4.08
		VBS	64.27	93.23	90.40	97.06	81.63	<b>95.86</b>	87.07	-7.67
		FPS	<b>81.09</b>	<b>96.42</b>	<b>92.48</b>	<b>97.92</b>	81.58	<b>95.74</b>	<b>90.87</b>	-3.87
		SOM (ours)	<b>83.44</b>	<b>96.83</b>	<b>93.88</b>	<b>98.17</b>	<b>87.58</b>	95.40	<b>92.55</b>	-2.19
	IoU (%)	3DEPS	66.91	90.83	85.67	95.60	<b>71.66</b>	90.64	83.55	-6.72
		VBS	47.35	87.33	82.48	94.29	68.96	<b>92.05</b>	78.74	-11.53
		FPS	<b>68.20</b>	<b>93.08</b>	<b>86.01</b>	<b>95.92</b>	68.89	<b>91.82</b>	<b>83.99</b>	-6.28
		SOM (ours)	<b>71.59</b>	<b>93.85</b>	<b>88.46</b>	<b>96.40</b>	<b>77.90</b>	91.21	<b>86.57</b>	-3.70

points after down-sampling. Our improved 1D SOM outperforms the traditional SOM by the suppression of outliers.

(ii) Our 1D SOM strategy is highly shape-aware, and is quite suitable for the down-sampling task of irregular point clouds such as plants and crops. After sampling, the plant stems can be abstracted by linearly connected nodes as lines in 3D, and the leaves can be represented by highly folded curves formed as connected nodes. This not only has biotaxonomic significance on aiding botanists to classify and recognize plant species, but also automatically encodes preliminary semantics into different organ types (e.g., stems are sampled as “lines”, and leaves are sampled as folded curves shaped in “surfaces”). The training data embedded with preliminary semantic shapes (e.g., “1D line”, and “2D surface”) can further improve the performance of downstream deep

networks for phenotyping tasks such as plant organ segmentation (Xue et al., 2023) and recognition (Gong et al., 2021).

(iii) Extensive experiments are conducted on a plant dataset containing three species by using several popular deep 3D-segmentation networks such as PointNet (Qi et al., 2017a), PointNet++ (Qi et al., 2017b), DGCNN (Wang et al., 2019), and PlantNet (Li et al., 2022c). The segmentation performance on the SOM-preprocessed dataset outperformed several mainstream down-sampling strategies.

(iv) The proposed down-sampling strategy has wide applicability to the problems such as point cloud skeleton extraction, crop main stem length measurement. Our sampling strategy also helps the point cloud datasets from other domains to present promising object semantic and instance segmentation results, indicating the good domain adaptation

**Table 4**

Quantitative comparison of our SOM to several other down-sampling strategies at different point resolutions on DGCNN (organ semantic segmentation).

Sampled Resolution	Evaluation Metrics	Sampling Strategies	Tobacco		Tomato		Chenopodium		Mean	Diff
			Stem	Leaf	Stem	Leaf	Stem	Leaf		
1024 Points	F1 (%)	3DEPS	<b>85.32</b>	95.60	94.26	97.91	91.71	97.02	93.64	-1.74
		VBS	75.58	96.18	93.74	98.06	94.20	<b>98.69</b>	92.74	-2.64
		FPS	85.30	<b>97.08</b>	<b>94.97</b>	<b>98.48</b>	<b>94.39</b>	98.59	<b>94.80</b>	-0.58
		SOM (ours)	<b>86.48</b>	<b>97.44</b>	<b>95.60</b>	<b>98.63</b>	<b>95.72</b>	98.44	<b>95.38</b>	0.00
	IoU (%)	3DEPS	<b>74.40</b>	91.57	89.14	95.90	84.69	94.22	88.32	-3.14
		VBS	60.75	92.64	88.21	96.20	89.04	<b>97.42</b>	87.38	-4.08
		FPS	74.37	94.34	<b>90.42</b>	<b>97.00</b>	<b>89.37</b>	<b>97.22</b>	<b>90.45</b>	-1.01
		SOM (ours)	<b>76.18</b>	<b>95.00</b>	<b>91.57</b>	<b>97.29</b>	<b>91.78</b>	96.92	<b>91.46</b>	0.00
512 Points	F1 (%)	3DEPS	83.03	95.12	93.97	97.91	87.79	96.15	92.33	-3.05
		VBS	<b>76.26</b>	96.31	93.05	97.91	<b>92.14</b>	<b>98.24</b>	92.32	-3.06
		FPS	<b>84.96</b>	<b>97.07</b>	<b>94.62</b>	<b>98.42</b>	<b>91.69</b>	<b>98.03</b>	<b>94.13</b>	-1.25
		SOM (ours)	<b>85.92</b>	<b>97.35</b>	<b>95.34</b>	<b>98.57</b>	<b>93.38</b>	97.56	<b>94.69</b>	-0.69
	IoU (%)	3DEPS	70.98	90.70	88.62	95.90	78.24	92.58	86.17	-5.29
		VBS	61.62	92.89	87.01	95.90	<b>85.42</b>	<b>96.54</b>	86.56	-4.90
		FPS	<b>73.86</b>	<b>94.30</b>	<b>89.79</b>	<b>96.89</b>	84.66	<b>96.14</b>	<b>89.27</b>	-2.19
		SOM (ours)	<b>75.31</b>	<b>94.83</b>	<b>91.09</b>	<b>97.18</b>	<b>87.58</b>	95.24	<b>90.21</b>	-1.25
256 Points	F1 (%)	3DEPS	82.34	95.85	93.28	98.06	84.63	95.19	91.56	-3.82
		VBS	<b>73.22</b>	95.22	92.47	97.54	<b>87.97</b>	<b>97.12</b>	90.59	-4.79
		FPS	<b>83.75</b>	<b>96.89</b>	<b>94.26</b>	<b>98.42</b>	86.57	96.60	<b>92.75</b>	-2.63
		SOM (ours)	<b>86.70</b>	<b>97.45</b>	<b>95.45</b>	<b>98.66</b>	<b>91.96</b>	<b>97.09</b>	<b>94.55</b>	-0.83
	IoU (%)	3DEPS	69.98	92.03	87.41	96.20	73.36	90.83	84.97	-6.49
		VBS	57.75	90.88	86.00	95.20	<b>78.52</b>	<b>94.41</b>	83.79	-7.67
		FPS	<b>72.04</b>	<b>93.96</b>	<b>89.15</b>	<b>96.88</b>	76.32	93.43	<b>86.96</b>	-4.50
		SOM (ours)	<b>76.52</b>	<b>95.03</b>	<b>91.30</b>	<b>97.35</b>	<b>85.11</b>	94.35	<b>89.94</b>	-1.52

ability. The improved 1D SOM can be generalized to 2D and 3D sampling, and can even be further extended to a more adaptive sampling framework that automatically picks the most suitable version of SOM for the corresponding shape component, respectively.

The rest of the paper is arranged as follows. Section 2 reviews related studies on three aspects. Section 3 elaborates the proposed SOM-based down-sampling strategy for plant point clouds. Section 4 shows experimental results of the proposed sampling strategy on several different deep learning networks with a multi-species 3D plant dataset. In section 5, we discuss the generalization, extension, and applications of the proposed SOM sampling. Conclusion is drawn in the last section. The notations and nomenclature are explained in Appendix A.1. The detailed qualitative comparisons across different sampling strategies on the organ semantic segmentation task by DGCNN are given in Appendix A.2. The detailed qualitative comparisons across different sampling strategies on the organ instance segmentation task by PlantNet are given in Appendix A.3.

## 2. Related works

### 2.1. Point cloud deep learning

Due to the unordered data structure and uneven distribution of 3D point clouds, direct application of the deep networks designed for 2D images to point clouds is infeasible. To overcome the convolution problem on point clouds, an indirect processing scheme emerged—first converting the point cloud into regular multi-view images (Su et al., 2015; Guerry et al., 2017; Riera et al., 2021) and then processing the images by a standard Convolutional Neural Network (CNN). Su et al. (Su et al., 2015) utilized the Multi-view Convolutional Neural Network (MVCNN) to process multi-view images synthetically generated from the 3D point cloud, and then mapped the results of the classification and segmentation tasks back to the original point cloud. In the field of crop breeding, a framework for soybean yield estimation based on multi-view images (Riera et al., 2021) was developed to evaluate soybean genotypes. Although the indirect deep learning scheme associates efficient CNNs with the unordered point cloud data, it has two limitations—(i) the projection from 3D point cloud to 2D images inevitably leads to loss of geometric information, and the projection angle greatly affects the

quality of projection; (ii) ambiguity exists when returning the 2D classification/segmentation results back to the 3D space, this is because one point can be mapped with pixels from several images.

Another way of indirect deep learning on point clouds is the voxelization method (Zhou et al., 2018; Tchapmi et al., 2017; Le et al., 2018), which refers to transforming an unstructured point cloud into a regular 3D grid where the powerful 3D convolutions can be applied for feature extraction. Tchapmi et al., 2017 proposed SEGCloud, in which a 3D-FCNN was initially operated to predict class probabilities at voxel level, and subsequently, trilinear interpolation was used to map the voxel segmentation result back to the original 3D point cloud. Finally, the labels of all points were updated using a Fully-connected Conditional Random Field (FC-CRF). Zou et al., 2017 proposed a voxel-based deep learning method to identify single trees in complicated forest point clouds. Their method first extracted individual trees via voxel-based upward growth filter (Guan et al., 2015), and then simplified the voxel raster to obtain a projected image of a single tree. The classification of individual trees was finally performed using a deep belief network. Although the voxelization method makes a substantial improvement over the indirect convolutions on multi-view images, the complex calculation and high requirement on graphic memory sometimes restrict its application. In addition, the size of the voxel significantly impacts the computational resolution, causing the voxel size to become a parameter that needs tuning; and it is also harder to reach fine-grained 3D segmentation results on voxelized data than on the original point cloud because the voxelization itself downgrades the resolution.

In order to better extract information from 3D point clouds and to reduce the structural complexity of the networks, PointNet realized classification and semantic segmentation with neither 3D-2D projection nor voxelization, which made it a pioneering network that runs straightly on original point clouds. Its enhanced version PointNet++ employed PointNet module multiple times and incorporated a local aggregation measure as feature encoder to learn local neighborhood features hierarchically. Enlightened by the frameworks of the PointNet family, researchers have proposed abundant improvements from multiple perspectives including point sorting (Li et al., 2018b; Huang et al., 2018; Wu et al., 2019), multi-scale feature aggregation (Ma et al., 2018; Ye et al., 2018; Li et al., 2021a), feature fusion (Jiang et al., 2018; Komarichev et al., 2019; Xu et al., 2018), and graph convolution (Wang

**Table 5**

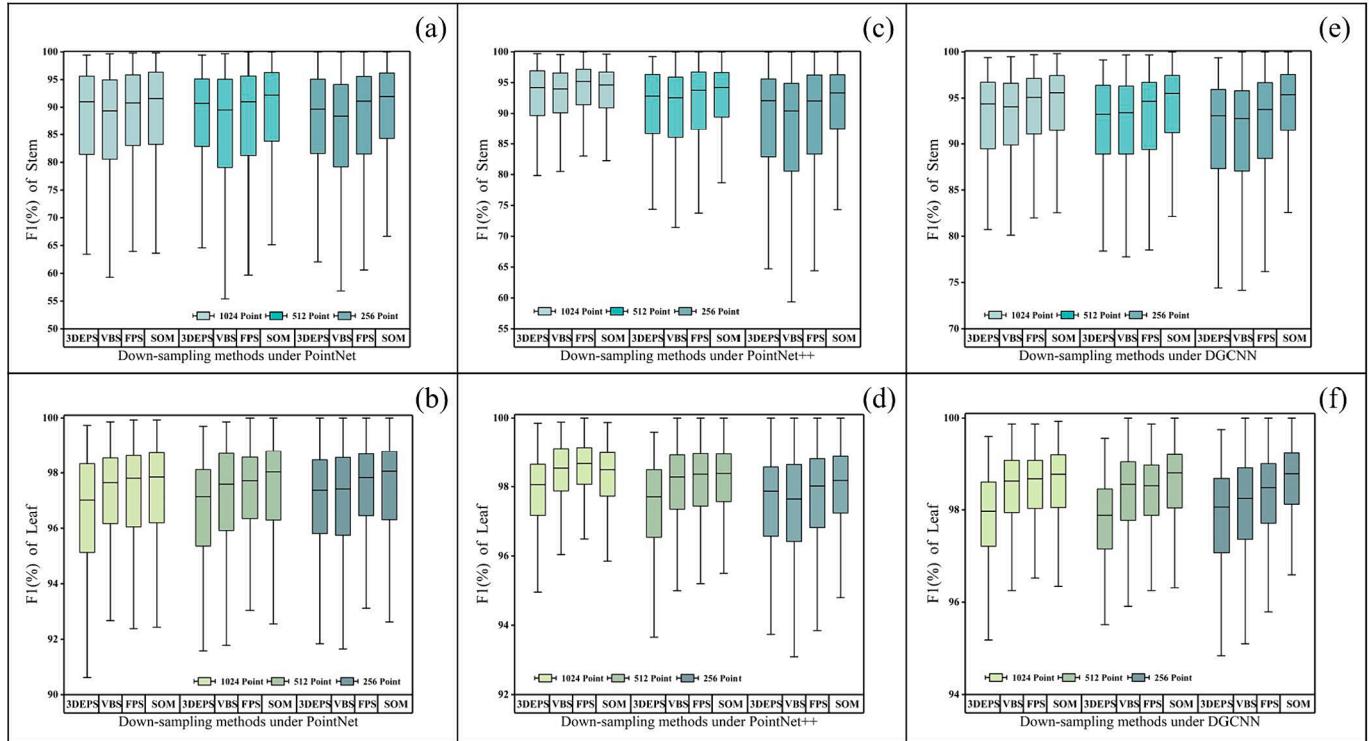
Quantitative comparison of our SOM to several other down-sampling strategies at different point resolutions on PlantNet (organ semantic segmentation and instance segmentation).

Sampled Resolution	Evaluation Metrics	Sampling Strategies	Tobacco		Tomato		Chenopodium		Mean	Diff
			Stem	Leaf	Stem	Leaf	Stem	Leaf		
1024 Points	mCov (%)	3DEPS	68.82	65.88	85.78	80.39	74.11	57.42	72.07	-9.52
		VBS	70.27	67.95	87.90	81.94	76.31	62.50	74.48	-7.11
		FPS	73.39	73.32	89.44	84.46	82.29	69.27	78.69	-2.90
		SOM (ours)	75.41	77.84	90.43	85.98	87.05	72.84	81.59	0.00
	mWCov (%)	3DEPS	68.82	73.85	85.78	85.23	74.11	70.15	76.32	-9.68
		VBS	70.27	78.57	87.90	87.91	76.31	76.03	79.50	-6.50
		FPS	73.39	83.40	89.44	90.05	82.29	81.66	83.37	-2.63
		SOM (ours)	75.41	87.20	90.43	90.52	87.05	85.38	86.00	0.00
	F1 (%)	3DEPS	77.66	86.43	91.79	94.61	84.94	87.59	87.17	-6.35
		VBS	76.52	91.00	93.60	95.70	88.97	92.30	89.68	-3.84
		FPS	82.19	92.86	94.75	96.64	90.95	92.55	91.66	-1.86
		SOM (ours)	84.29	95.81	95.25	97.67	93.45	94.66	93.52	0.00
512 Points	IoU (%)	3DEPS	63.48	76.11	84.83	89.78	73.82	77.93	77.66	-10.46
		VBS	61.97	83.49	87.98	91.75	80.13	85.70	81.84	-6.28
		FPS	69.76	86.67	90.03	93.49	83.41	86.13	84.92	-3.20
		SOM (ours)	72.84	91.96	90.92	95.44	87.71	89.87	88.12	0.00
	mCov (%)	3DEPS	67.50	63.58	86.46	78.84	70.94	65.69	72.17	-9.42
		VBS	69.91	70.69	87.46	82.57	75.88	67.51	75.67	-5.92
		FPS	72.79	74.77	88.80	85.32	79.75	72.90	79.06	-2.53
		SOM (ours)	74.32	77.20	90.31	86.30	83.10	74.75	80.99	-0.60
	mWCov (%)	3DEPS	67.50	70.06	86.46	84.22	70.94	77.22	76.07	-9.93
		VBS	69.91	80.78	87.46	88.90	75.88	77.17	80.17	-5.83
		FPS	72.79	83.13	88.80	90.71	79.75	83.73	83.15	-2.85
		SOM (ours)	73.96	87.84	90.19	90.66	83.04	86.40	85.35	-0.65
256 Points	F1 (%)	3DEPS	76.95	86.69	92.20	94.09	84.31	92.02	87.71	-5.81
		VBS	78.64	93.05	93.69	96.73	88.92	91.65	90.45	-3.07
		FPS	81.76	93.56	94.15	97.02	87.88	92.17	91.09	-2.43
		SOM (ours)	83.22	95.43	95.22	97.60	91.23	95.21	92.99	-0.53
	IoU (%)	3DEPS	62.54	76.51	85.53	88.84	72.87	85.22	78.58	-9.54
		VBS	64.80	87.02	88.14	93.67	80.06	84.60	83.05	-5.07
		FPS	69.15	87.90	88.95	94.22	78.38	85.47	84.01	-4.11
		SOM (ours)	71.26	91.26	90.87	95.32	83.88	90.86	87.24	-0.88
	mCov (%)	3DEPS	62.42	61.67	84.00	78.86	70.39	67.13	70.74	-10.85
		VBS	66.38	67.94	87.02	82.20	77.87	73.92	75.89	-5.70
		FPS	71.27	73.25	87.80	84.24	77.43	77.58	78.60	-2.99
		SOM (ours)	71.88	74.69	90.07	85.54	83.76	77.85	80.63	-0.96
256 Points	mWCov (%)	3DEPS	62.42	68.76	84.00	84.63	70.39	79.34	74.92	-11.08
		VBS	66.38	78.06	87.02	87.92	77.87	83.22	80.08	-5.92
		FPS	71.27	82.77	87.80	89.52	77.43	86.66	82.58	-3.42
		SOM (ours)	71.88	84.60	90.07	90.10	83.76	87.31	84.62	-1.38
	F1 (%)	3DEPS	72.00	87.10	90.48	94.67	82.96	92.99	86.70	-6.82
		VBS	77.00	92.18	93.19	96.33	89.61	95.52	90.64	-2.88
		FPS	81.37	93.90	94.03	96.87	88.74	96.16	91.85	-1.67
		SOM (ours)	81.19	94.01	94.63	97.12	90.25	95.02	92.04	-1.48
	IoU (%)	3DEPS	56.26	77.15	82.61	89.88	70.88	86.91	77.28	-10.84
		VBS	62.60	85.50	87.24	92.93	81.17	91.43	83.48	-4.64
		FPS	68.60	88.51	88.73	93.94	79.76	92.60	85.36	-2.76
		SOM (ours)	68.33	88.69	89.81	94.40	82.23	90.51	85.66	-2.46

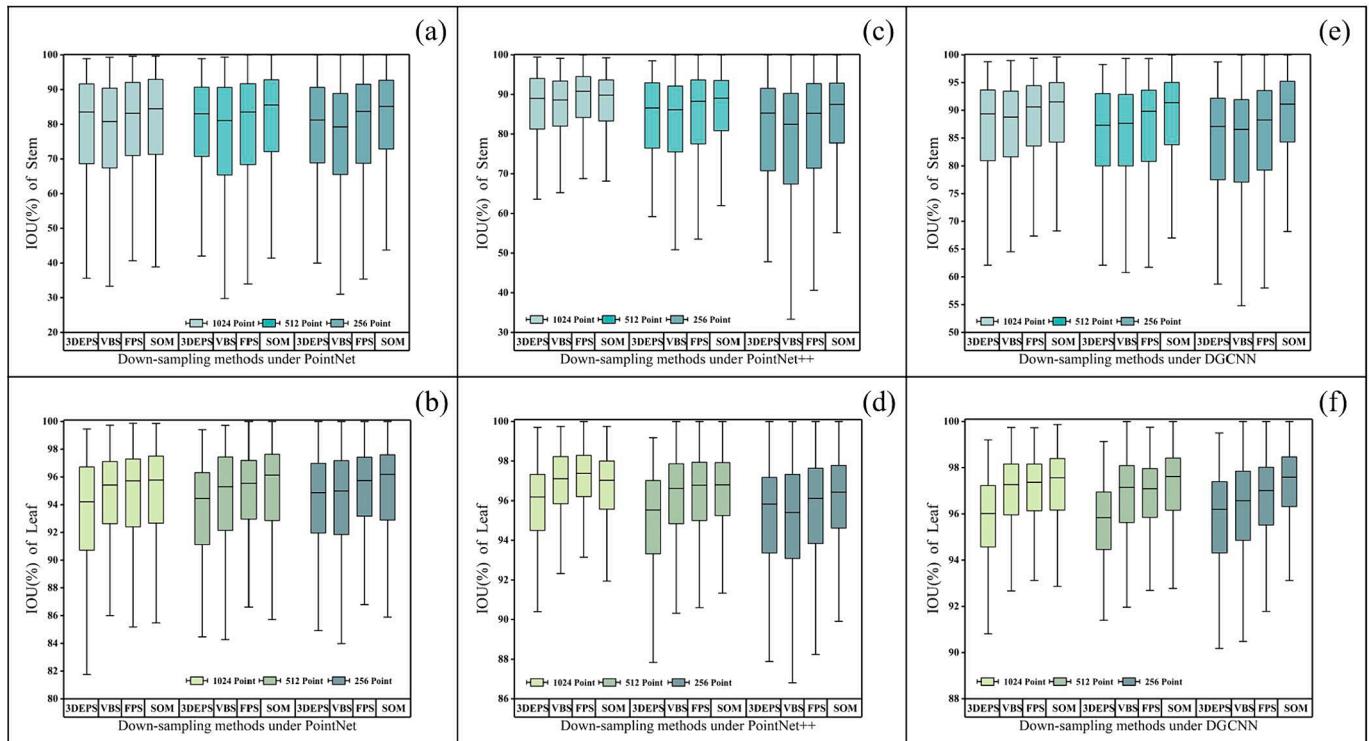
et al., 2019; Qi et al., 2017c; Simonovsky et al., 2017). For instance, PointCNN (Li et al., 2018b) applied  $\mathcal{X}$ -Conv to perform convolution calculations on a point-by-point basis for better feature extraction. PointConv (Wu et al., 2019) introduced an efficient convolution scheme with density weighting, and the learning weights are based on relative coordinates to address the issue of unordered convolution. The 3DMAX-Net (Ma et al., 2018) adopted a multi-scale approach by connecting and merging deep feature maps from adjacent layers to strengthen point cloud feature extraction. Ye et al., 2018 proposed a pyramid hierarchical pooling model with RNN, enabling the aggregation of local neighborhood features at different scales. Motivated by SIFT (Xiang et al., 2018), Jiang et al., 2018 designed a symmetric end-to-end PointSIFT module that integrated information from multiple directions. A-CNN (Komarichev et al., 2019) employed a hierarchical neural network and ring convolution with local area matching, and it is specifically suitable for point cloud semantic segmentation tasks in complex scenes. DGCNN exploited graph edge convolution (EdgeConv) to extract neighboring features in dynamic graph feature spaces, which further prompted point cloud segmentation performance. More recently, influenced by the rapid

development of Vision Transformer (ViT) in the image domain, deep networks on 3D with attention mechanisms have emerged. Point Cloud Transformer (PCT) (Guo et al., 2021a) suggested replacing the MLP layer in the PointNet framework with a self-attention layer, which further refined the feature extraction ability of the network. Hierarchical attention modules, such as offset attention blocks (Lu et al., 2022) and deformable self-attention (Bhattacharyya et al., 2021), effectively calculated the relationship between points without special and complex local network designs.

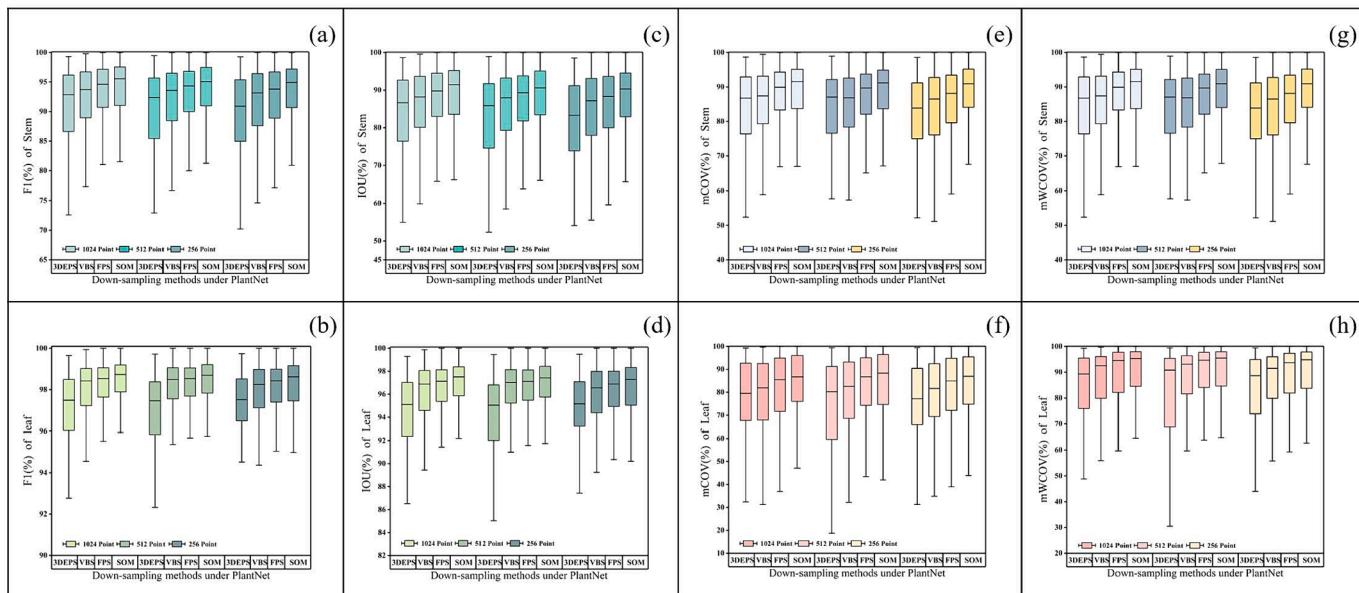
In fact, the advanced deep networks that could directly process original point clouds were quickly implemented in modern agriculture and forestry after publication, which has substantially propelled the development of 3D crop phenotyping technology. In Masuda et al., 2021, PointNet++ was utilized to segment point clouds of tomato crops in greenhouses for leaf area estimation. Li et al., 2022d, employed an improved PointNet model for organ-level point cloud segmentation on high-throughput maize 3D data with a high precision. Ying et al., 2021 proposed a 3D tree detection method for LiDAR-scanned point clouds in complicated scenes based on PointCNN, demonstrating satisfactory



**Fig. 8.** Box plots (of PointNet, PointNet++, and DGCNN) showing the distributions of the F1 score (%) (for all crop samples) of the four compared down-sampling strategies under 256-, 512-, and 1024-point resolutions, respectively. Each box boundary represents the 25th and 75th percentiles; the line in the box represents the median. (a) and (b) are the F1 distributions for the stem class and the leaf class on PointNet, respectively. (c) and (d) are the F1 distributions for the stem class and the leaf class on PointNet++, respectively. (e) and (f) are the F1 distributions for the stem class and the leaf class on DGCNN, respectively.



**Fig. 9.** Box plots (of PointNet, PointNet++, and DGCNN) showing the distributions of the IoU score (%) (for all crop samples) of the four compared down-sampling strategies under 256-, 512-, and 1024-point resolutions, respectively. Each box boundary represents the 25th and 75th percentiles; the line in the box represents the median. (a) and (b) are the IoU distributions for the stem class and the leaf class on PointNet, respectively. (c) and (d) are the IoU distributions for the stem class and the leaf class on PointNet++, respectively. (e) and (f) are the IoU distributions for the stem class and the leaf class on DGCNN, respectively.



**Fig. 10.** Box plots showing the distributions of F1 (%), IoU (%), mCOV (%), mWCOV (%) (on all crop samples) of the four down-sampling strategies on PlantNet under 256-, 512-, and 1024-point resolutions, respectively. Each box boundary represents the 25th and 75th percentiles; the line in the box represents the median. (a) and (b) are the F1 distributions for the stem class and the leaf class on PlantNet, respectively. (c) and (d) are the F1 distributions for the stem class and the leaf class on PlantNet, respectively. (e) and (f) are the mCov distributions for the stem class and the leaf class on PlantNet, respectively. (g) and (h) are the mWCov distributions for the stem class and the leaf class on PlantNet, respectively.

detection accuracy and generality. Additionally, Li et al., 2022a designed a novel plant point cloud segmentation network called MASPC-Transform which incorporated multi-head attention separation and position encoding, and reported promising performance on the ROSE-X dataset (Dutagaci et al., 2020).

It is anticipated that the deep learning methods that directly work on original point clouds will continue to be an important research direction in field of crop 3D phenotyping for the foreseeable future. The state-of-the-art deep networks predominantly rely on a fixed-point and fixed-scale data input constraint for training and testing purposes. Consequently, it becomes crucial to generate well-structured datasets from raw point clouds with a highly efficient and precise down-sampling strategy. Hence, conducting comprehensive research on down-sampling strategies holds immense significance to the further enhancement of the network performance.

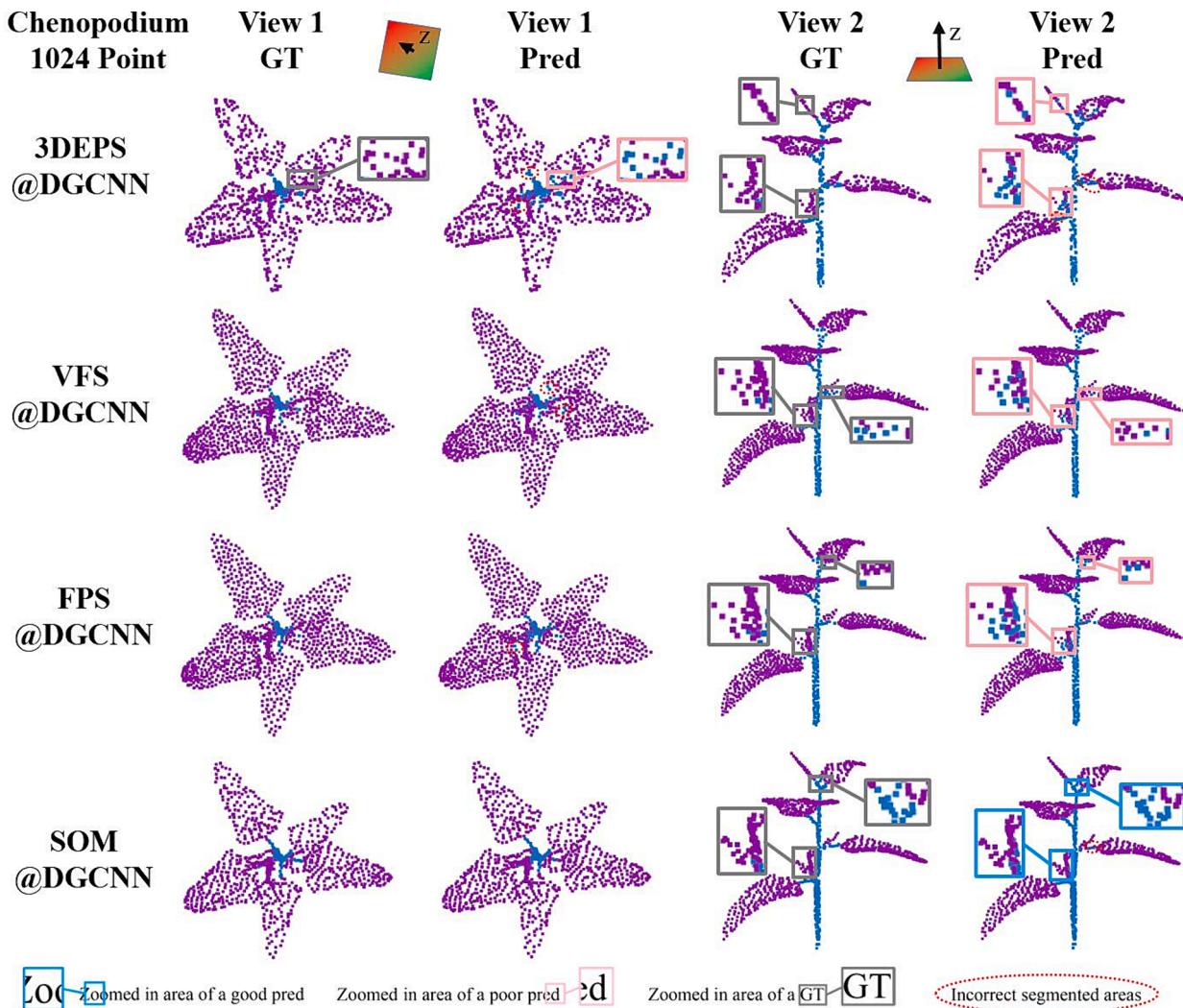
## 2.2. Down-sampling on point clouds

Aiming to reduce storage volume and accelerate data processing, unsupervised down-sampling methods have been a popular means to simplify point cloud data. These methods can be broadly categorized as Random Sampling (RS), Uniform Sampling (US), grid-based down-sampling, down-sampling by geometric features, etc. Although the RS strategy (Vitter, 1984), may sometimes amplify unevenness in distribution of points after sampling due to its inherent randomness, the fast computation as well as the precise control over the number of points make RS a popular choice for real-time applications. For instance, in order to achieve target detection from sparse LiDAR point clouds, VoxelNet used RS to divide the LiDAR point cloud into equally spaced 3D voxels to save computational overhead and reduce sampling bias. Similarly, RandLA-Net (Hu. et al., 2020), a lightweight network designed for large-scale street point cloud semantic segmentation, adopted RS to construct the dataset. RandLA-Net also incorporated a local feature aggregation module to compensate the geometric feature loss caused by RS. In the P2B framework (Qi et al., 2020) for 3D object tracking in point clouds, randomly sampled seed points were generated from the template-defined object and the search area, respectively. These seed points were then used to determine the potential object centers,

facilitating the correlation and verification of 3D targets in the next stage.

Farthest Point Sampling (FPS) is a widely used strategy for unsupervised uniform sampling. A typical implementation procedure of FPS is as follows—first, a seed point is randomly selected from the point set  $\mathcal{P}$  to establish an initial sampled point set  $\mathcal{A}$ ; second, the distance from each point in the set  $\mathcal{P}/\mathcal{A}$  to all points in set  $\mathcal{A}$  is calculated, and the minimum distance is defined as the distance from that point to the point set  $\mathcal{A}$ ; third, the point in the set  $\mathcal{P}/\mathcal{A}$  that is the farthest to set  $\mathcal{A}$  is moved to set  $\mathcal{A}$ , and meanwhile removed from set  $\mathcal{P}/\mathcal{A}$ . The second and third steps are repeated until the desired number of sampled points is reached. Because FPS generates even sampling results, it can be inserted after the feature extraction module of the encoder of a deep network to further abstract the key features in the point cloud. These FPS-sampled high-dimensional key points play a big role in aggregating semantic information, the examples can be found in PointNet++, PoinCNN, and PointConv. It is worth noting that FPS is usually based on Euclidean distance in the feature space, which tends to overlook sparse and irregular structures as well as important details of non-Euclidean surfaces. Additionally, when the desired number of sampled points is much smaller than the total number of original points, FPS tends to form a porous structure, resulting in uneven sampling.

Voxel-based Sampling (VBS) (Zheng et al., 2012) is a classic strategy for unsupervised grid-based down-sampling. The VBS process first divides the point cloud into voxels, where each voxel block has parameters  $l_x$ ,  $l_y$ , and  $l_z$  for length, width, and height, respectively. The points within each voxel block are then replaced with the centroid of that block. Although voxel-based down-sampling can produce a standard and uniform sampling result, it has several disadvantages—(i) The sampling highly depends on the size parameters of the voxel, making it hard to control the exact number of sampled points; (ii) when the size of voxel is large, VBS strongly smooths the shape of the point cloud, and causes loss of fine-grained information; (iii) The use of uniform voxel blocks generates array-like point sets, weakens the geometric diversity of training samples and may further negatively influence the training of deep networks. To assure the consistency of the point number, some approaches combined VBS with post-processing. For instance, the Voxelized Farthest Point Sampling (VFPS) proposed in PSegNet (Li et al., 2022b)



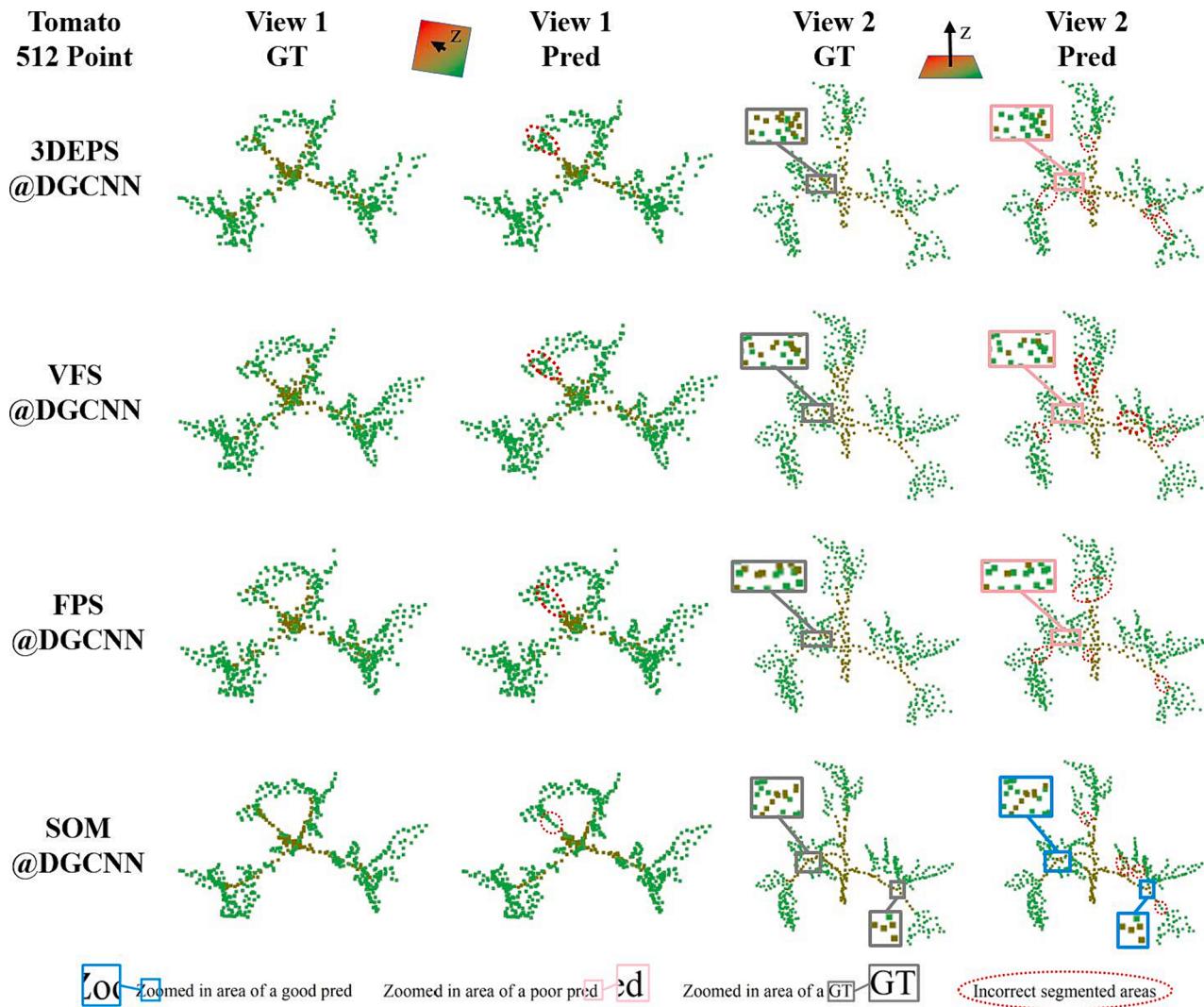
**Fig. 11.** The qualitative semantic segmentation results of a chenopodium crop by DGCNN at the 1024-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details.

first performed VBS, and then carried out another round of FPS on voxels to achieve the desired number of points. This hybrid approach not only absorbs the advantage of stable and standardized distribution of points, but also introduces some sort of randomness to diversify the point distribution by incorporating randomly initialized seed point of FPS.

Geometric features can also be utilized for unsupervised point cloud down-sampling. One such method is the Normal Space Sampling (NSS) (Kwok, 2018), which leveraged normal vector field for down-sampling. NSS involves several steps—first, the main curvature of each point is approximated by calculating the normal vector with its neighborhood; second, a threshold is set up to segment regions with large curvature variations; and at last, points are sampled proportionally according to the regions, with a preference of collecting more points in regions with larger curvature changes as they may preserve crucial structures. Since NSS captures more points from regions with large curvature variations, the density of points after down-sampling may vary from region to region, resulting in uneven sampling. In fact, the outline, which is also a primary geometric feature of the object, also can be beneficial to shape recognition. The 3DEPS operator (Li et al., 2022c) drew inspiration from sketching; it exerted precise control over the ratio of edge points to the total sampled points to intentionally save more information on the edges and outline of 3D objects. Although 3DEPS proves that a moderately increase of the edge point proportion (i.e., to save more outline

information) can further improve the accuracy of point cloud segmentation networks such as PlantNet, the ratio of edge points still requires manual tuning. In summary, the low-level geometric features such as normal and curvature require additional computation during point cloud processing, and selection of neighborhoods and parameters (for calculating geometric features) on irregular shapes also requires tuning. Li et al., 2023 evaluated five different unsupervised down-sampling strategies on several popular deep segmentation networks for plant phenotyping, and concluded that the optimal down-sampling strategy might vary among different networks.

Recently, the emergence of task-specific supervised down-sampling methods has caught eyes. S-Net, proposed by Dovrat et al., 2019, is a point cloud simplification deep learning module that is optimized for a downstream task network such as for classification, retrieval or reconstruction. S-Net's framework utilized a task loss function to train the downstream backbone network for a specific task and meanwhile used a sampling loss to adjust the simplification module. The simplified point set by S-Net is not guaranteed to be a strict subset of the original point set; therefore, a matching operation was added to establish correspondences between the input points and the reduced points. However, the matching operation is unable to propagate the task loss, leading to a performance disparity between the training and the testing phases. In order to constrain the sampled point set to approximate the original set,



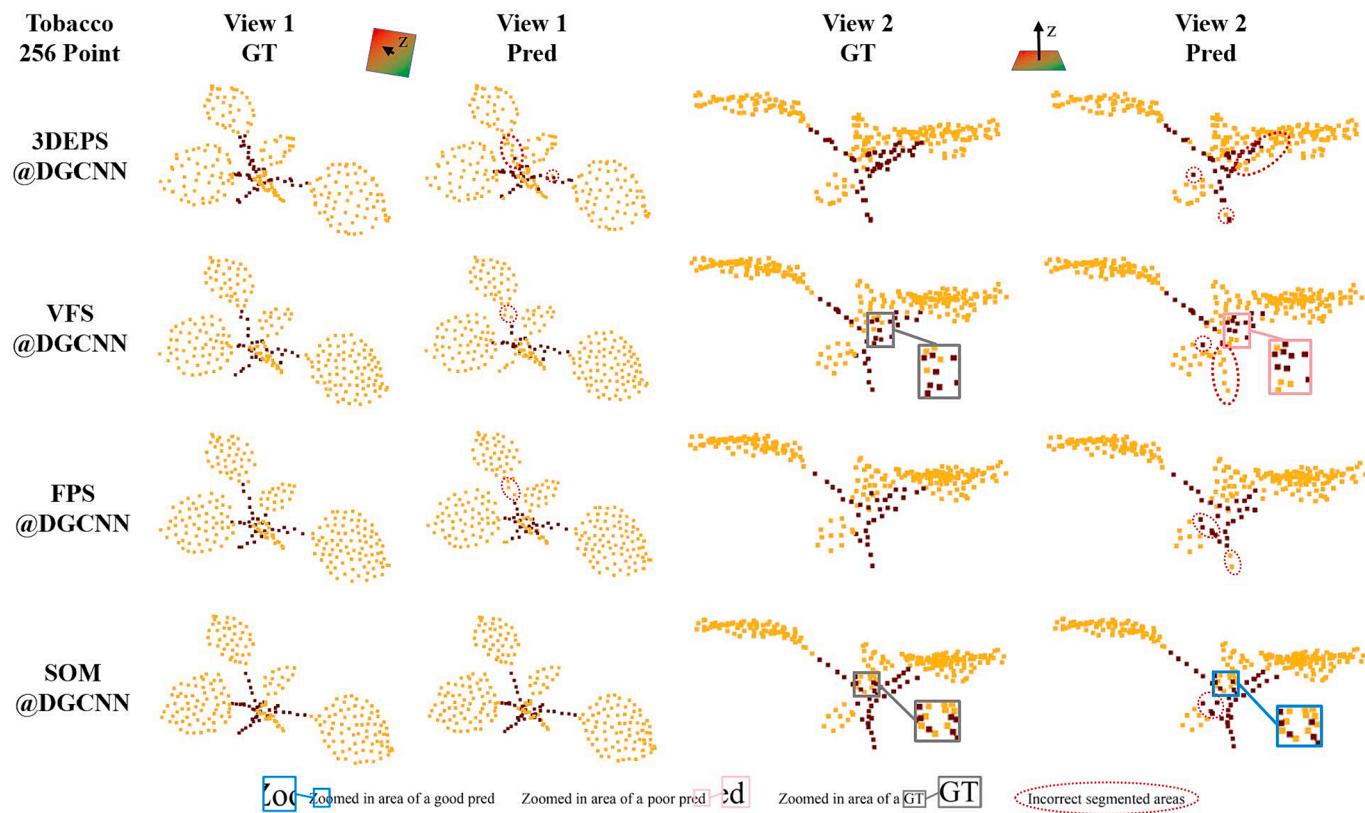
**Fig. 12.** The qualitative semantic segmentation results of a tomato crop by DGCNN at the 512-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details.

S-Net's subsequent work SampleNet (Lang et al., 2020) designed a soft projection module that is differentiable to enhance the generated point distribution. SampleNet is able to generate sampled points that are both highly task-specific and faithful to the original input point cloud. Enlightened by applied the Point Cloud Transformer (PCT), PST-NET (Wang et al., 2021) redesigned the point simplification module of S-Net using a transformer unit for self-attention and local feature extraction, aiming to generate the optimal re-sampling distribution. Another method called Critical Points Sampling Layer (Nezhadarya et al., 2020) introduced a critical point layer that assessed the feature contribution of each input point via its importance, and increased the selection probability of critical points during down-sampling. Despite the task-specific attribute, the practice of supervised down-sampling methods still requires an unsupervised pre-sampling step (such as FPS) to unify all point clouds in the dataset to have a fixed point number, on which the learning-based sampling and downstream tasks are then carried out. Furthermore, supervised sampling has to construct a sampling module and a task network, and the training process and the inference can be quite different. In addition, the change of data domain usually means a do-over training for supervised sampling frameworks. Therefore, purely unsupervised and efficient down-sampling methods remain to be a crucial preprocessing technique for point cloud data before deep

learning. The objective of this paper is to develop an unsupervised point cloud down-sampling strategy that has high efficiency, high data generality, and low structural information loss.

### 2.3. Self-organizing maps

The main structure of Self-organizing Map (SOM) can be regarded as a backpropagation-free two-layer neural network which comprises an input layer and an output layer (also referred as the competition layer). Though SOM resembles a simple neural network in structure, the two differ a lot in the way of learning. SOM is updated with unsupervised competitive learning instead of the loss-guided supervised learning adopted in common deep neural networks. For example, the simplest SOM (Kohonen, 1998) arranges output neurons in a “1D” pattern in which each non-terminal node is connected with its left and right neighbors on a line (illustrated in Fig. 1). When an input is received, the neuron whose weight is the closest to the input value is activated in the output layer. Subsequently, only the weights of this neuron and the neurons within its activation field are updated. This iterative calculation process inputs one by one and continues until convergence is reached. The activation mechanism of neurons in SOM borrows the biological observation of the well-known Lateral Inhibition (LI) phenomenon that



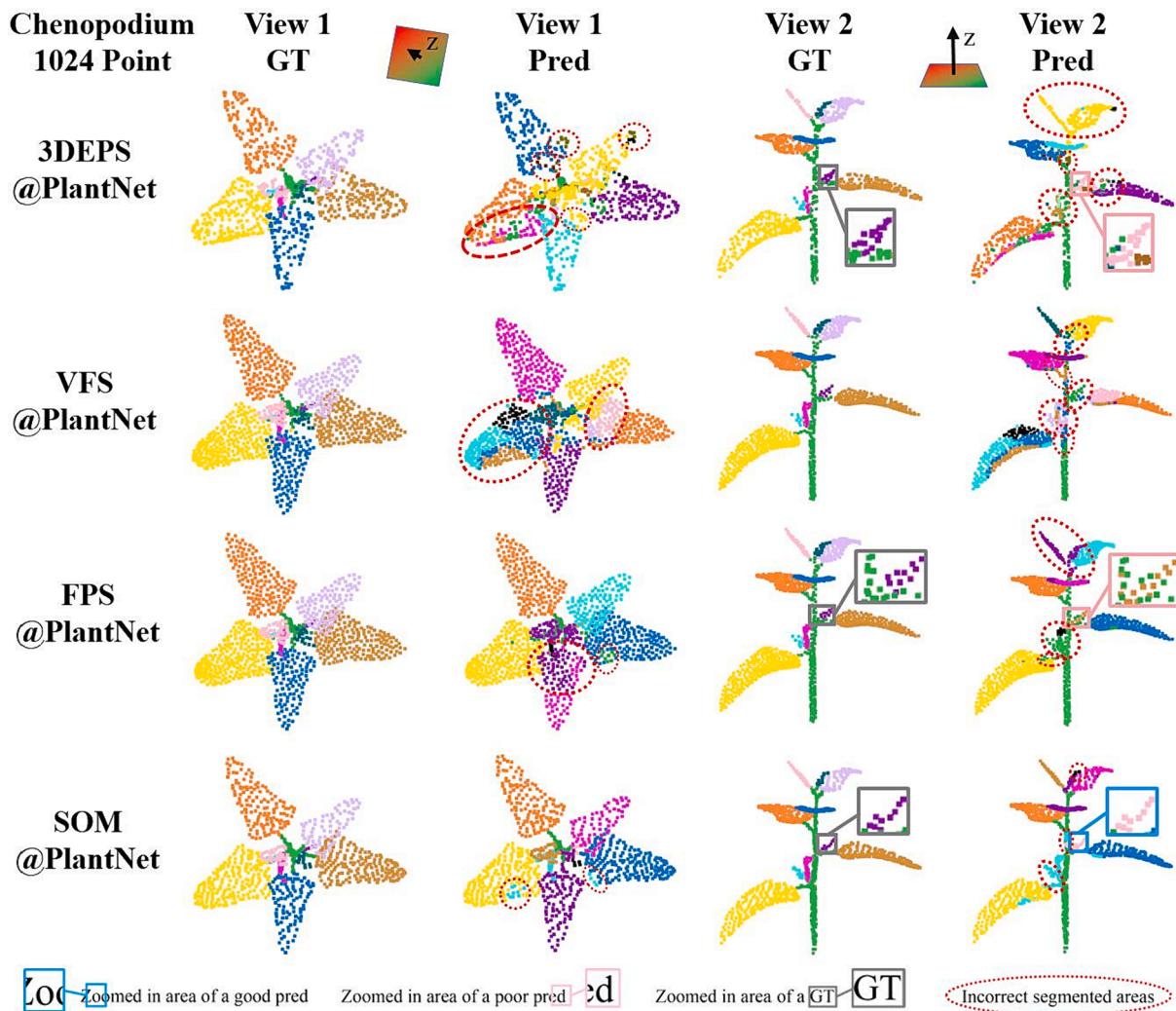
**Fig. 13.** The qualitative semantic segmentation result of a tobacco crop by DGCNN at the 256-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details.

is pervasive in visual neural pathways (Cook and McReynolds, 1998; Protti et al., 2014), auditory neural pathways (Levy and Reyes, 2011), and the primary sensory cortex (DeLaRocha et al., 2008). The discovery and confirmation of LI were initially reported by Hartline et al., 1956, through experiments on the visual physiology of the limulus. In general, when a certain region of the cortex is stimulated, the excitatory neurons will strongly inhibit the activities of peripheral nerves on the same layer. Fig. 2 visualizes the iterative competitive learning process as well as the simulated LI of the simplest SOM on a 2D lattice input. It is evident that the global structure is quickly captured by SOM, followed by fine-tunes on the perception of local structures, and ultimately a compact skeleton-like result is obtained after only 120 iterations.

Since the first version of SOM proposed by Kohonen, a considerable amount of works has focused on applying SOM to specific engineering problems such as defect detection, pattern recognition, 3D reconstruction. Joutsiniemi et al., 1995 applied standard SOM to the recognition of topographic patterns of EEG spectra. Based on routine clinical EEG's of several subjects, SOM is able to recognize similar topographic spectral patterns in different EEG's. The learned SOM differentiates between individual background activities and recognizes eye movements and muscle artifacts. Singh et al., 2000 used kernel-based SOM to extract skeleton from 2D sparse point sets, and the scheme can be applied to character recognition on scanned texts. To more effectively detect all the abnormalities of chip locations within the wafer, Lee et al., 2001 extracted the spatial defect features in the historical wafer bin map data and cluster the chip locations having similar defect features through SOM neural network. Rego et al., 2010 proposed a surface reconstruction method based on growing self-organizing maps (SOMs), called growing self-reconstruction maps (GSRMs). It can produce a triangular two-manifold mesh representation of a target object given an unstructured point cloud of its surface. An improved version of GSRM, called GSRM-IDT (DoRêgo et al., 2011), is an incremental SOM that produces

finer 2-manifold meshes that concerns the reconstruction of real-world human anatomical structures. SOM can also be used to generate skeletonized point set from 3D plant models (Chebrolu et al., 2021; Magistri et al., 2020) to serve the automated organ growth tracking.

Theoretical extensions and generalizations of SOM were also actively studied by the research community in past decades. Benson and Hu, 2000 proposed asynchronous SOM neural network for sampling of spatial data, which is a convergence-guaranteed generalization of the standard SOM. Self-organizing Mixture Network (SOMN) (Yin and Allinson, 2001) is derived for learning arbitrary density functions. The first layer of SOMN uses similar standard SOM competitive learning to learn density parameters, and the second layer accumulates the responses of these local nodes weighted by the learned mixing parameters to approximate the density function. Visualization-induced SOM (ViSOM) (Yin, 2002) was used to overcome the distortion problem in visualization of high-dimensional data after competitive learning. The topologies of the common SOM map for analyzing 2D datasets are nearly always square or, more rarely, hexagonal. López-Rubio and Ramos, 2014 studied alternative grid topologies that are derived from geometrical theory of tessellations, and discovered that the optimal topology depends on the problem at hand. Burguillo, 2014 explored the use of complex network topologies like small-world, scale-free or random networks for connecting the neurons in the SOM. The results suggest that the most regular network topology usually bring best prediction performance for Time Series Prediction (TSP). The SO-Net (Li et al., 2018a) modeled the spatial distribution of point cloud by building a self-organizing map (SOM). Hierarchical feature extraction is performed in SO-Net on individual points and SOM nodes to realize point cloud segmentation and classification, and the training speed is faster than existing point cloud recognition networks because of the parallelizability and simplicity of the architecture. For some data sets and applications, it is known beforehand that some regions of the input space



**Fig. 14.** The qualitative instance segmentation results of a chenopodium crop by PlantNet at the 1024-point resolution with four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details. Different colors represent different leaf instances, and those different leaf colors are only used for visualization, and there is no specific label meaning for each color.

cannot contain any samples, i.e., the forbidden regions. Ramos et al., 2020 modified the standard SOM to avoid learning from the forbidden regions defined.

In summary, the majority of previous SOM-related works focus on either specific engineering problems or the connection between SOM and neural networks by examining the influence of a node-connecting topology of SOM on learned result. The analysis of SOM's extraordinary manifold learning ability on data spaces with different dimensionalities has long been overlooked.

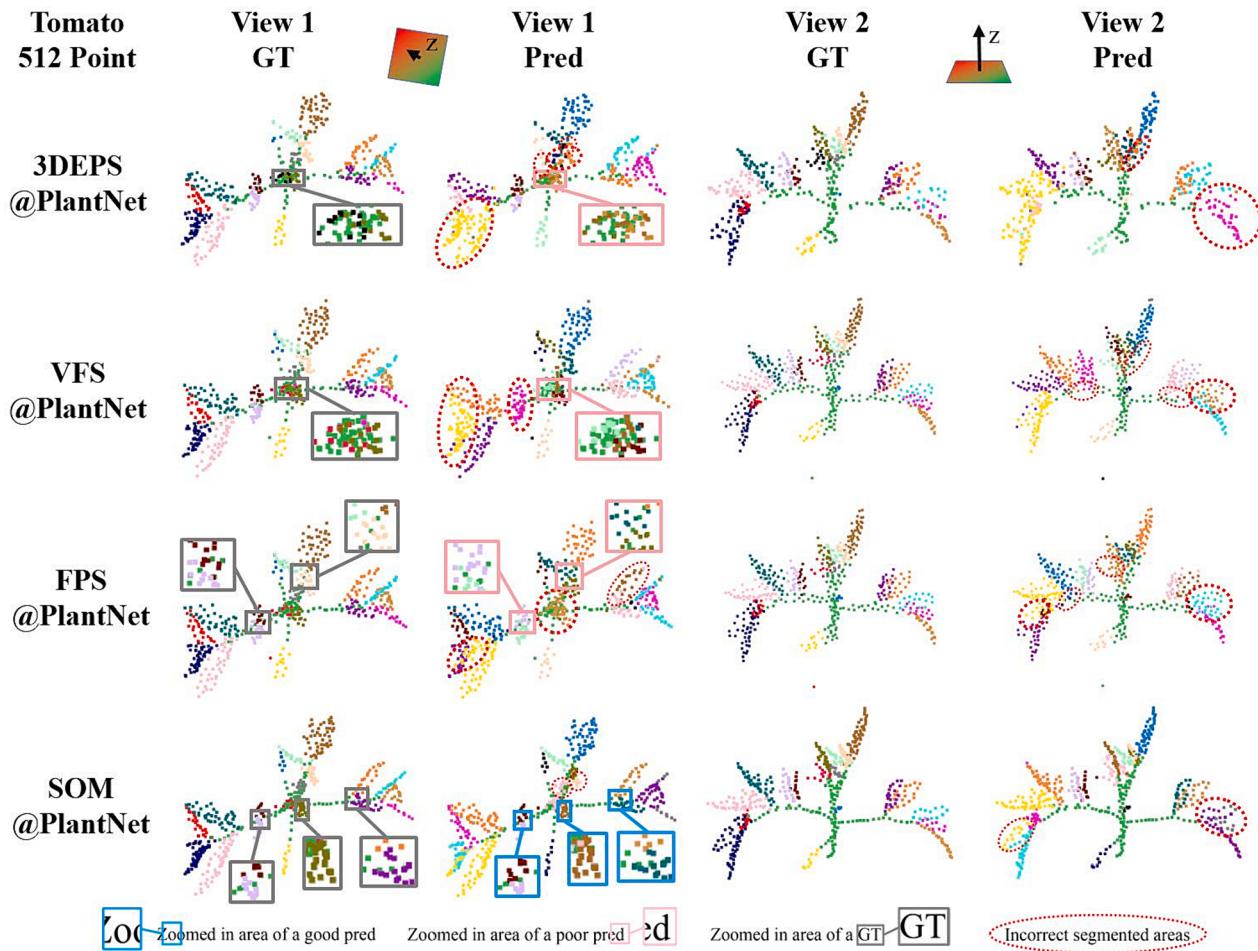
### 3. Method

#### 3.1. Improved SOM-based down-sampling

The traditional SOM algorithm usually has outliers in output (see Fig. 3) and sometimes suffers from slow convergence. In this section we propose an improved unsupervised Self-organizing Map (SOM) strategy tailor-made for down-sampling of plant point clouds. The proposed algorithm—improved 1D SOM-based down-sampling, is summarized as pseudocode in Table 1 as two connecting steps. The proposed algorithm generally differs from the traditional SOM in two ways—(i) the neurons are initialized along the direction of the largest variance of the data distribution in the competitive learning step, giving faster convergence;

and (ii) a refinement step is added after competitive learning to suppress the outliers. Explicitly, an iterative approach is first developed to update the neuron weights in the 1D output layer in competitive learning. And after that, the output nodes are not strictly a subset of the original point set and may still contain outliers. Therefore, we designed a downstream refinement procedure to pick out the outliers from the converged neurons and move each outlier to the closest point in the original point set, which constrains the output of Algorithm 1 to the original input topology. The objective number of sampled points  $M$  is set as the quantity of neurons in the output layer, and the weight vectors of the output neurons after Algorithm 1 represent the coordinates of the down-sampled points in the space.

Some implementation details of our algorithm should also be noted. In Step I of Table 1, the total number of iterations— $MIN$  is divided into multiple batches of size  $N$ . Within each batch, a point  $x_t$  is randomly drawn from a copy of the original point set  $\mathcal{X}$  without replacement for calculation. This approach ensures that each batch of iterations covers each input point exactly once for unbiased learning. During each competitive learning iteration of Algorithm 1, only the neuron  $w_j$  that is the closest to the input wins. To reduce computational complexity, we employ a Gaussian function centered at  $w_{j^*}$  (Row 8 of Table 1) to simulate the sombrero-shaped Lateral Inhibitory neuron excitation



**Fig. 15.** The qualitative instance segmentation results of a tomato crop by PlantNet at the 512-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details. Different colors represent different leaf instances, and those different leaf colors are only used for visualization, and there is no specific label meaning for each color.

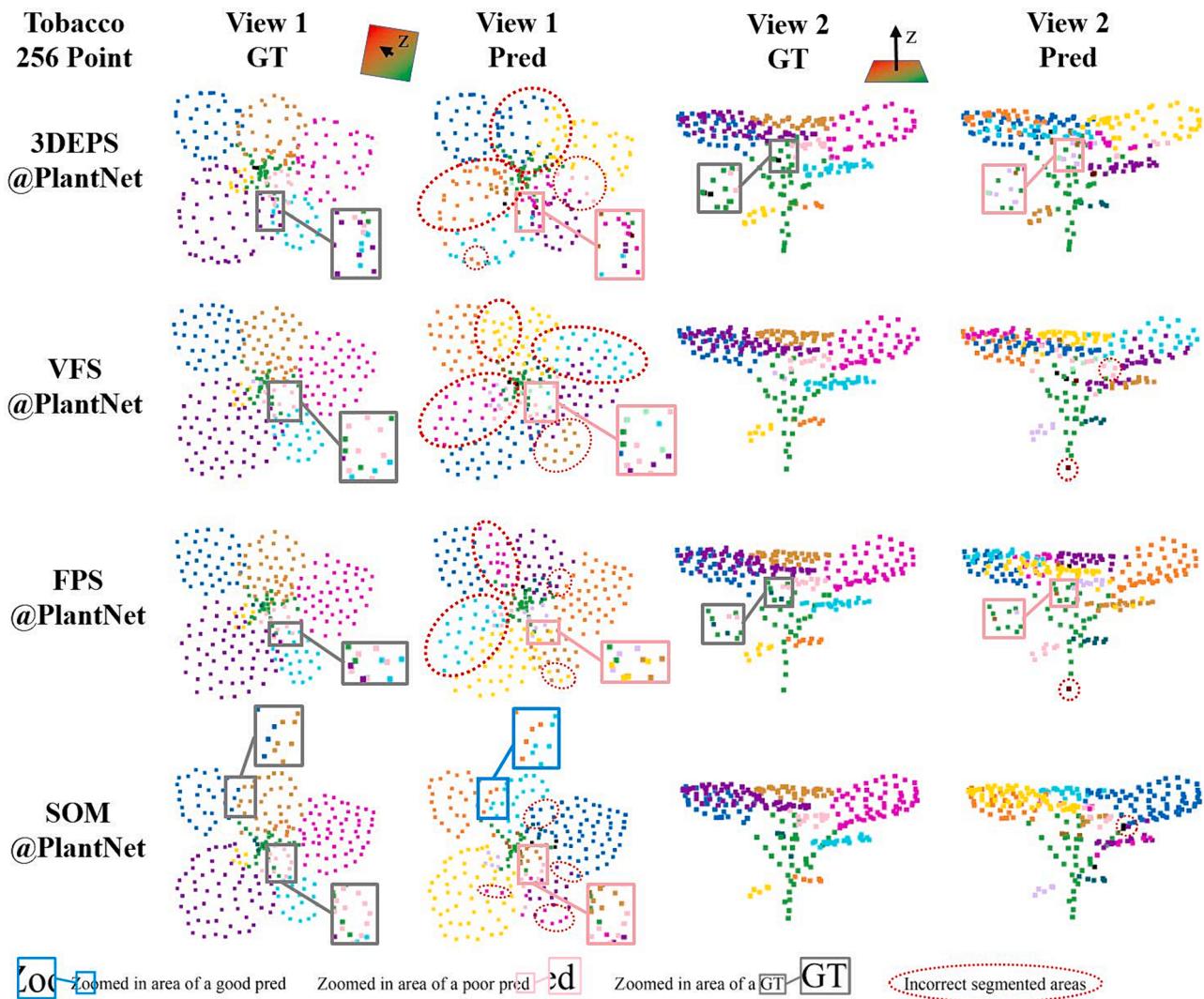
pattern. Hence, though we allow all  $M$  neurons to be updated during Rows 6–10 in Step I, the Gaussian forms a 3-sigma activation field  $\mathcal{N}_{j^*}$  (see Fig. 1(b)), making only the nodes inside the field to be actually updated, and all neurons outside remain almost unchanged. As the iterations progress, the size of  $\mathcal{N}_{j^*}$  gradually decreases and finally contains merely the nearest neighbors. The Step II of Table 1 (Refinement) can be further divided into two sub-steps. The first sub-step scans the competitive learning result, and keeps only the neurons that are near to any real original points in  $\mathcal{I}$ , and the rest of neurons are very likely to be outliers. The second sub-step then handles the rest of neurons pushed in set  $\mathcal{O}$ , relocates each of them to the nearest point in the original input set  $\mathcal{X}$ . The unoptimized implementation of Step I of Algorithm 1 has a complexity of  $\mathcal{O}(\text{MIN}\cdot M)$ , and the unoptimized complexity of Step II of Algorithm 1 is around  $\mathcal{O}(MN)$  because the outliers are few. The total unoptimized time complexity of Algorithm 1 is about  $\mathcal{O}(\text{MIN} + N)$ . If a more efficient data structure (such as KdTree) is used, the total complexity can be reduced to around  $\mathcal{O}(\text{MIN}\cdot \log M + N\cdot \log M)$ . And if  $M \ll N$ , the complexity approaches  $\mathcal{O}(N)$ , which makes the proposed SOM algorithm quite efficient in practice.

### 3.2. Shape-aware attribute

The improved 1D SOM in Algorithm 1 possesses good shape-aware attribute. The common basic local shapes found in complicated 3D objects include slender spatial structures (e.g., lines, curves, and pipes), flat and curved surfaces, and solid 3D sub-objects. Real point clouds can be

approximated as a combination of these basic shapes. Our SOM with 1D structured neurons is able to abstract the linear 3D structure using well-arranged linear output layer nodes (Fig. 4(a)). A plane-like surface can be uniformly approximated by a 2D manifold formed by repetitive folded output neurons of Algorithm 1 (Fig. 4(b)). Furthermore, after learning, the 1D SOM can even densely fill a solid 3D object within only a few iterations (see the cube in Fig. 4(c)). Therefore, SOM exhibits remarkable ability in preserving the fundamental linear, planar, and spatial features during abstraction. For crop point clouds, the stem system is often regarded as a tree-like network formed by interconnected line segments, and the leaves typically occur as 3D surfaces. Hence, Algorithm 1 is inherently well-suited for the down-sampling task of the two types of organs—stem and leaves. Fig. 4(d) displays the down-sampling result of a main stem segment of a tobacco crop by Algorithm 1, and Fig. 4(e) shows the down-sampling result for a tobacco leaf by Algorithm 1. Our 1D SOM represent uniform abstraction ability on both the stem and the leaf in space. The sampling of a whole tobacco point cloud by Algorithm 1 is shown in Fig. 4(f). It can be seen that after a certain number of iterations, the output not only automatically fits the stem system with linearly aligned nodes but also uniformly covers all leaf surfaces.

Additionally, SOM-based strategy can also better accommodate different spatial topological forms by modifying the connection topology of neuron nodes in the output layer. For instance, Fig. 4 and Fig. 5(a) show how the fundamental 1D-structured SOM competition layer approximate different shapes. The 1D output layer can be naturally



**Fig. 16.** The qualitative instance segmentation results of a tobacco crop by PlantNet at the 256-point sampling resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details. Different colors represent different leaf instances, and those different leaf colors are only used for visualization, and there is no specific label meaning for each color.

generalized to a 2D lattice referred as in Fig. 5(b), or a 3D grid as illustrated in Fig. 5(c). The effectiveness of these generalized SOMs on sampling certain types of point clouds will be further discussed in Section 5.1.

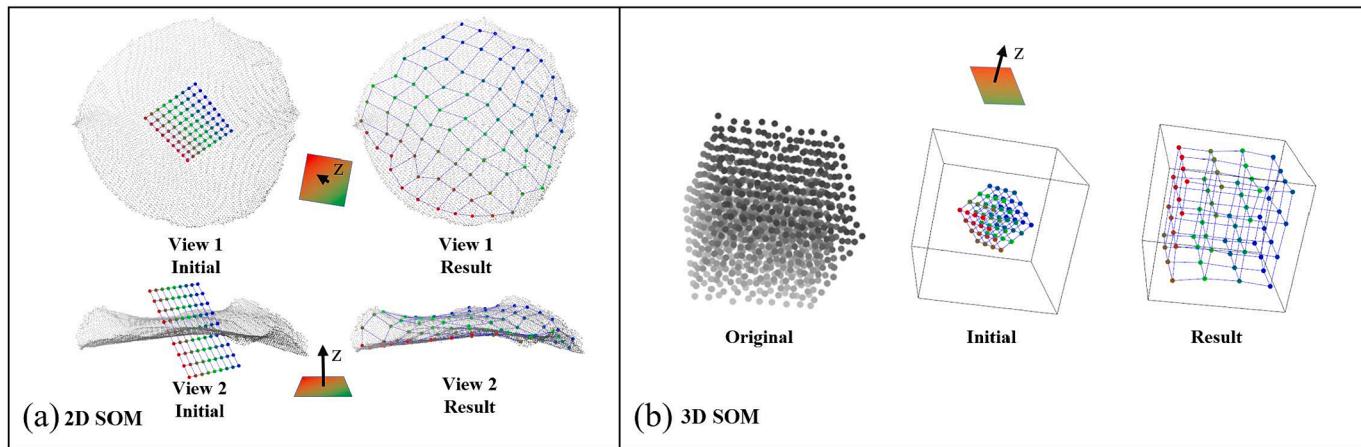
#### 4. Experiments and results

##### 4.1. Dataset and preparation

The plant dataset used in this paper comprises three crop species with different shapes, containing a total of 441 point clouds (105 tobaccos, 312 tomatoes, and 24 chenopodiums). The original point cloud data of tobacco and tomato plants come from (Conn et al., 2017), and were scanned by high-precision LiDAR in different growth environments and with multiple growth stages. Each sample point cloud of the two species contains approximately 10,000 to 100,000 points. The data of chenopodium species, obtained from (Mirande et al., 2022), were generated from multiple view images. Compared with the previous two species, the point cloud data is sparser, and each sample contains about 10,000 points. Fig. 6 shows example point clouds of the three species in our dataset, in which the plant structures are quite different among species. Notably, tobacco has broad leaves, which usually

produce dense canopies that are prone to occlusion. Tomato leaves are mainly compound leaves with jagged edges, which poses a challenge for almost all segmentation frameworks. The chenopodium data has a lower resolution and may cause segmentation/classification difficulties at the junction of stem and leaf. To annotate the organs of these crops, we employed the Semantic Segmentation Editor (SSE) (<https://github.com/Hitachi-Automotive-And-Industry-Lab/semantic-segmentation-editor>) and CloudCompare software (<https://github.com/cloudcompare/cloudcompare>). The semantic labeling categorizes each crop into two semantic classes: the stem system (including petioles) and leaves (including top leaflets and buds). For the three different species, there are a total of 6 semantic categories in the dataset after semantic labeling. Instance labeling was further performed to annotate each instance of leaves based on the previously assigned ‘leaf’ semantic labels, and each single leaf in the three leaves-related semantic categories is assigned a different instance index. The dataset was divided into a training set and a testing set by a 2:1 ratio to facilitate subsequent downstream tasks such as segmentation based on neural networks.

The experimental design of this chapter can be shown by Fig. 7. The main task in our experiment is to utilize Algorithm 1 (named with ‘SOM’ for short) for down-sampling and preprocessing crop point cloud datasets that are then used to train and test multiple segmentation



**Fig. 17.** Generalized SOMs show good shape-aware ability on different 3D point clouds. (a) shows the competitive learning results of 2D SOM (Algorithm 2) on a tobacco leaf. Although the initial planar lattice, consisted of 81 neurons ( $9 \times 9$  lattice), looks very differently from the leaf shape in terms of both orientation and size, it perfectly covers the leaf surface after convergence; (b) presents the competitive learning results of 3D SOM (Algorithm 3) on a point set in a cube area. Although the initial 3D grid competition layer, consisted of 64 neurons ( $4 \times 4 \times 4$  grid), is very different from the cube shape in terms of both orientation and size, it then perfectly aligns the solid space after convergence.

networks for plant organ phenotyping. Furthermore, the segmentation results of our SOM-based sampling and several other down-sampling strategies are carefully evaluated (both quantitatively and qualitatively) to verify the superiority of SOM.

This section implements four representative deep networks to segment crop organs on our dataset. Three of the networks are designed for semantic segmentation, including the pioneering point-level deep segmentation/classification network—PointNet (Qi et al., 2017a) and its updated version PointNet++ (Qi et al., 2017b), and the dynamic graph deep network DGCNN (Wang et al., 2019). The last network is a dual-function segmentation network—PlantNet (Li et al., 2022c), which is able to simultaneously conduct semantic and instance segmentation tasks through parallel pathways. PointNet, PointNet++, and DGCNN are trained and tested using merely semantic organ labels, while PlantNet employs both semantic labels and instance labels. The segmentation performance of SOM-generated dataset is compared with datasets generated by three other popular down-sampling methods—FPS (Qi et al., 2017b), VBS (Zheng et al., 2012), and 3DEPS (Li et al., 2022c), on every aforementioned network. For 1D SOM, the only two parameters that should set before initialization are the initial value of learning rate  $\lambda$  and the activation radius  $\sigma$ . Throughout the experiments in this paper, we fix the initial value of  $\lambda$  as 0.5, and the value of  $\sigma$  has two choices at 1.2 and 1.5 for two independent experiments. To comprehensively evaluate the strengths and weaknesses of all down-sampling strategies, the point number after down-sampling is fixed at three resolutions—1024, 512, and 256, respectively. This forms datasets at three different scales. Quantitative and qualitative experimental results are then given under different point resolutions.

To better serve the uniform-scale requirement of data for deep network training, the scales of all point clouds were normalized to a unit space prior to down-sampling. Afterward, the normalized dataset was down-sampled and simultaneously augmented by 10 times to enhance data diversity before feeding into any of the compared networks. The data augmentation steps for different down-sampling strategies are not alike due to different point distribution patterns. Take SOM as an example, we executed 10 separate random initializations for the output layer for every input point cloud, then parallelly carried out iterative competitive learning to produce 10 independent sampled point subsets. By data augmentation, each of down-sampling strategy in the experiment has a dataset containing 4,410 plant point clouds. Finally, for each network, the training set will comprise 2,940 plant point clouds (tobacco 700, tomato 2,080, and Chenopodium 160), and the testing (inference) set will contain 1,470 plant point clouds (tobacco 350,

tomato 1,040, and Chenopodium 80).

All experiments in this study were conducted on a server equipped with a 16-core (32-thread) CPU, 128 GB memory, and 4 NVIDIA GeForce RTX 2080Ti GPUs. The server was operated under the Ubuntu 20.04 system, and the training framework was TensorFlow 1.13.1. The training of these deep networks adopted the recommended parameters and configuration from their original papers, and the batch size of the training phase was fixed at 16, while testing phase was fixed at 1. All point clouds feed to the networks only contains 3D coordinates and label information, without any auxiliary information such as color or normal vectors.

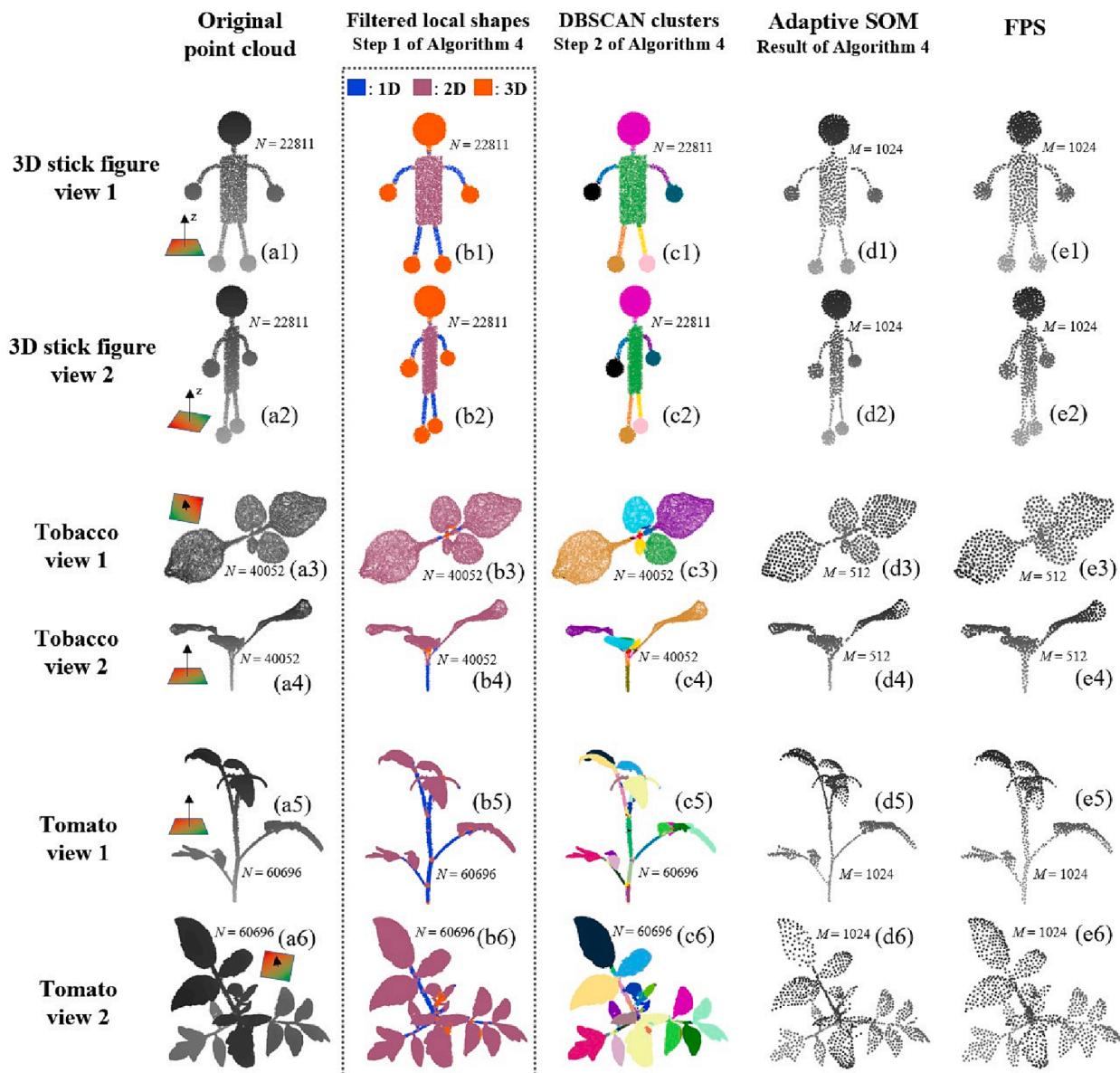
#### 4.2. Evaluation metrics

For the semantic and instance segmentation tasks of three plant species, four comprehensive quantitative metrics are selected to evaluate the task performance of the network under differently sampled input. For the semantic segmentation, F1-score (F1) and Intersection over Union (IoU) are calculated at the point level on each semantic category. In order to explain how to compute F1 and IoU, five different metrics should be first introduced. True Positive (TP) represents the number of points in the semantic class that is correctly identified by the model. False Positive (FP) is the number of points that are predicted as the current class but actually should belong to other semantic classes. False Negative (FN) is the number of points that the model fails to recognize in the current semantic class. Precision (Prec) refers to the proportion of correctly classified points of a class to all points in the same class. Recall (Rec) is the proportion of correctly classified points of a class to the total points of that class in Ground Truth (GT). Then, F1-score is defined as the harmonic mean of Precision and Recall. IoU measures the overlap between the predicted area of each semantic category and the corresponding real area. The equations for computing F1 and IoU are as follows:

$$F1 = 2 \cdot \frac{Prec \cdot Rec}{Prec + Rec} \quad (1)$$

$$IOU = \frac{TP}{TP + FP + FN} \quad (2)$$

For the instance segmentation task, the point-level instance prediction is evaluated by the mean coverage (mCov) and the mean weighted coverage (mWCov). The two metrics are defined as follows:



**Fig. 18.** Step results and final results of the adaptive SOM for down-sampling. The 1st and 2nd rows show results of a stick figure point cloud from two views, respectively. The 3rd and 4th rows show results of a Tobacco point cloud from two views, respectively. The 5th and 6th rows show results of a Tomato point cloud from two views, respectively. The 1st column shows the original point clouds. The 2nd column renders the three types of local shapes labeled after Step 1 in Algorithm 4 by different colors. The 3rd column renders each DBSCAN cluster after Step 2 with a different color. The 4th column shows the final results of Algorithm 4. The 5th column shows the FPS results for contrast.

$$mCov = \frac{1}{I} \sum_{m=1}^I \max_n [\text{IOU}(IG_m, IP_n)] \quad (3)$$

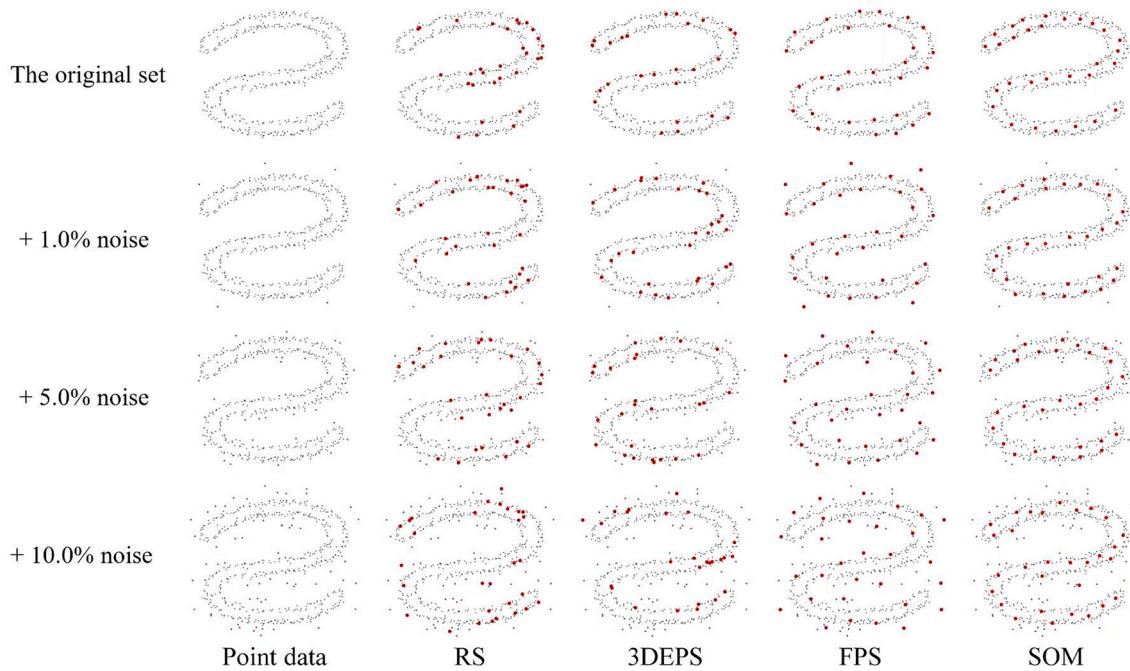
$$mWCov = \sum_{m=1}^I \frac{IG_m}{\sum_{i=1}^I IG_i} \cdot \max_n [\text{IOU}(IG_m, IP_n)] \quad (4)$$

where  $I$  is the number of all instances contained in a semantic category of GT,  $IG_m$  represents the region of the  $m$ -th instance in the current semantic category of GT, and  $IP_n$  denotes the region of the  $n$ -th instance predicted by the model. The function  $\text{IoU}(\cdot, \cdot)$  takes a set of ground truth points and a set of predicted points as two inputs, and performs standard intersection-over-union calculation as (2) on the two inputs. Please note that in all our experiments, the stem class of each crop sample actually only has one instance—the stem, because the stem segments are interconnected and the meaning of separation into stem parts is not clear. Therefore, for the stem organ class, the  $mCov$  and  $mWCov$  are the same

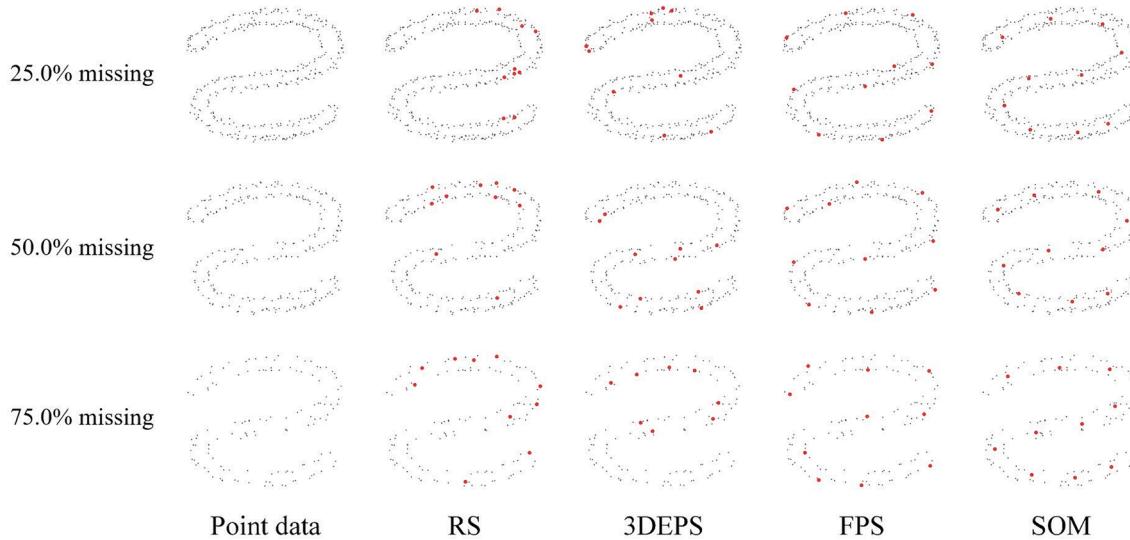
according to equations (3) and (4).

#### 4.3. Quantitative comparison against other strategies

This provides a comprehensive quantitative comparison between improved 1D SOM (Algorithm 1) and three other popular unsupervised down-sampling strategies on different deep networks for segmenting crop point clouds. The comparison is independently conducted at three different sampling resolution levels: 1024 points, 512 points, and 256 points. The quantitative comparison is first performed on the PointNet semantic segmentation network. Table 2 presents the results regarding to the average F1 and IoU for all crop semantic classes. In the table, the best result is in boldface, while the second-best result is underlined. The “Mean” column is the average performance across all semantic classes, and the “Diff” value indicates the average performance difference (on the same metric) between the current strategy and the best performer across all resolutions. Obviously, our strategy is superior to the other



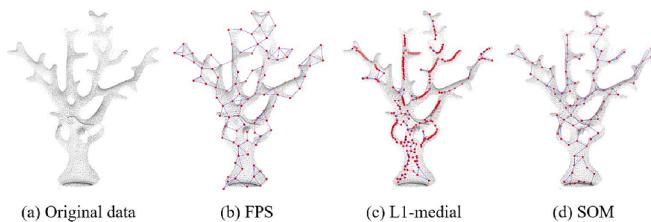
**Fig. 19.** Comparison of our improved 1D SOM and other three down-sampling strategies in the presence of noise. The points from the original “2”-shaped data space are represented by small gray dots, and the random noise points are small purple points. The sampled points are labeled with bigger red points. In this experiment, the down-sampled point scale is fixed to 30 points. The 1st row shows the sampling results of the original point set without noise; from the 2nd to the 4th row are the sampling results under randomly added noise level at 1.0%, 5.0%, and 10.0%, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



**Fig. 20.** Comparison of our improved 1D SOM and other three down-sampling strategies in the presence of missing data. The points from the original “2”-shaped dataset are represented by small gray dots, and the sampled points are labeled with bigger red points. In this experiment, the down-sampled point scale is fixed to 10 points. From the 1st row to the 3rd row we show the down-sampling results on the point sets that randomly loses 25.0%, 50.0%, and 75.0% of the points from the original “2”-shaped set, respectively. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

three strategies on both F1 and IoU metrics at all three resolutions, showing good potential to optimize the segmentation performance of PointNet. Moving on to the PointNet++ semantic segmentation network, Table 3 displays the quantitative comparison across the down-sampling strategies on PointNet++. It can be observed that at both resolutions of 512 points and 256 points, SOM obtains the best average performance. Though slightly outcompeted by FPS at the resolution of 1024, SOM still secures the second place in average performance. In general, SOM shows better segmentation performance on PointNet++ comparing to the other three strategies. The quantitative comparison of

the down-sampling strategies on the DGCNN semantic segmentation network is shown in Table 4. Across all three resolutions, our SOM-based sampling strategy ranks first in the average quantitative segmentation results with a significant lead against the second-best strategy. Finally, the quantitative comparison of down-sampling strategies performed on the PlantNet segmentation network is shown in Table 5, including both the instance and semantic tasks. Notably, the datasets prepared using the SOM-based sampling strategy consistently deliver the best average semantic and instance segmentation results across all three resolutions. This further consolidates the superiority of our SOM on optimizing the



**Fig. 21.** Comparison of skeleton extraction methods on a coral point cloud. (a) The original data is a tree-like coral point cloud containing 14,207 points; (b) shows the skeleton built on nodes (red points) extracted by FPS, the number of nodes is fixed at 96. (c) is the skeleton extracted by the L1-medial algorithm. L1-medial is unable to precisely control over the number of skeleton nodes, there are 251 points in this skeleton. Three main parameters  $\mu, h_0$  and  $\Delta h$  of L1-medial are fixed by the recommended tuning scheme from the original paper, with values at 0.35, 0.2169, and 0.1084, respectively. (d) is the skeleton formed on nodes sampled by our improved 1D SOM, the number of nodes is also fixed at 96. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

segmentation performance of the dual-function deep network.

We also made box plots to visualize the statistical properties of the quantitative measures on all four networks across different down-sampling strategies. The box plots for the PointNet, PointNet++, and DGCNN networks are shown in Fig. 8 and Fig. 9. And Fig. 10 shows the box plots for PlantNet. Fig. 8 contains box plots showing the distributions of the F1 score of the four compared down-sampling strategies under 256-, 512-, and 1024-point resolutions, respectively. In all 18 comparative F1 cases of Fig. 8, our SOM is the best performer on 16 cases, and is the second best for the rest 2 cases on PointNet++ (1024-point stem and 1024-point leaf). Fig. 9 contains box plots showing the distributions of the IoU of the four compared down-sampling strategies under three point resolutions, respectively. In all 18 comparative IoU cases of Fig. 9, our SOM is the best performer on 16 cases, and is the second best for the rest 2 cases on PointNet++ (1024-point stem and 1024-point leaf). Fig. 10 contains box plots showing the distributions of F1 (%), IoU (%), mCov (%), mWCov (%) scores of the four down-sampling strategies on PlantNet under three different point resolutions, respectively. Our SOM is always the best among all 24 comparative cases of Fig. 10.

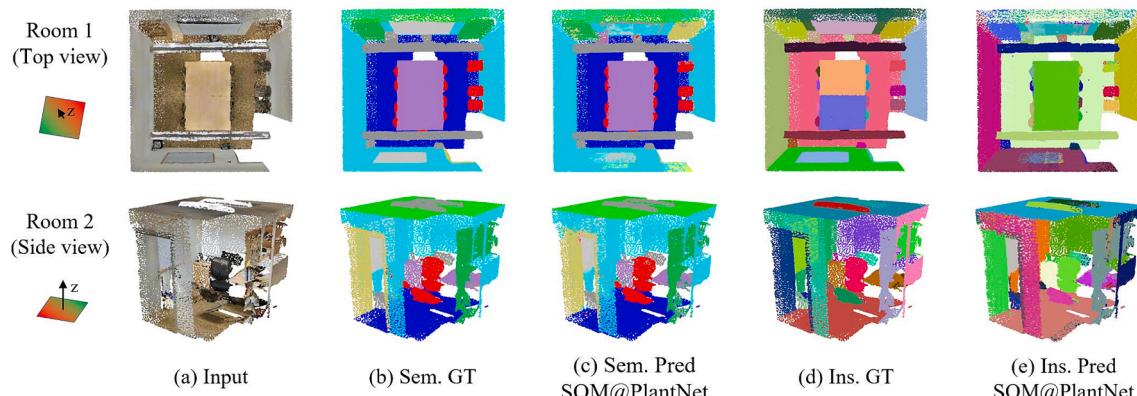
Overall, our SOM-based sampling consistently achieves the best average performance in all quantitative experiments shown in Tables 2–5 and Figs. 8–10, except for several cases of PointNet++ at the 1024-point resolution. This highlights the good capability of SOM in preserving plant point cloud information. The plausible reason for not taking

the best result at 1024-point resolution of PointNet++ is that PointNet++ employs multiple rounds of FPS for high-dimensional feature abstraction in the encoder, which possibly causes the PointNet++ to have the best compatibility with FPS-sampled 3D crop point clouds as input. In other words, PointNet++ is specifically designed to fit FPS on both point cloud preprocessing and network architecture. Though FPS outperforms our SOM on the 1024-point segmentation of PointNet++, ours is still the second best across all four sampling methods on 1024-point resolution, and our SOM quickly outperforms FPS when the resolution goes down.

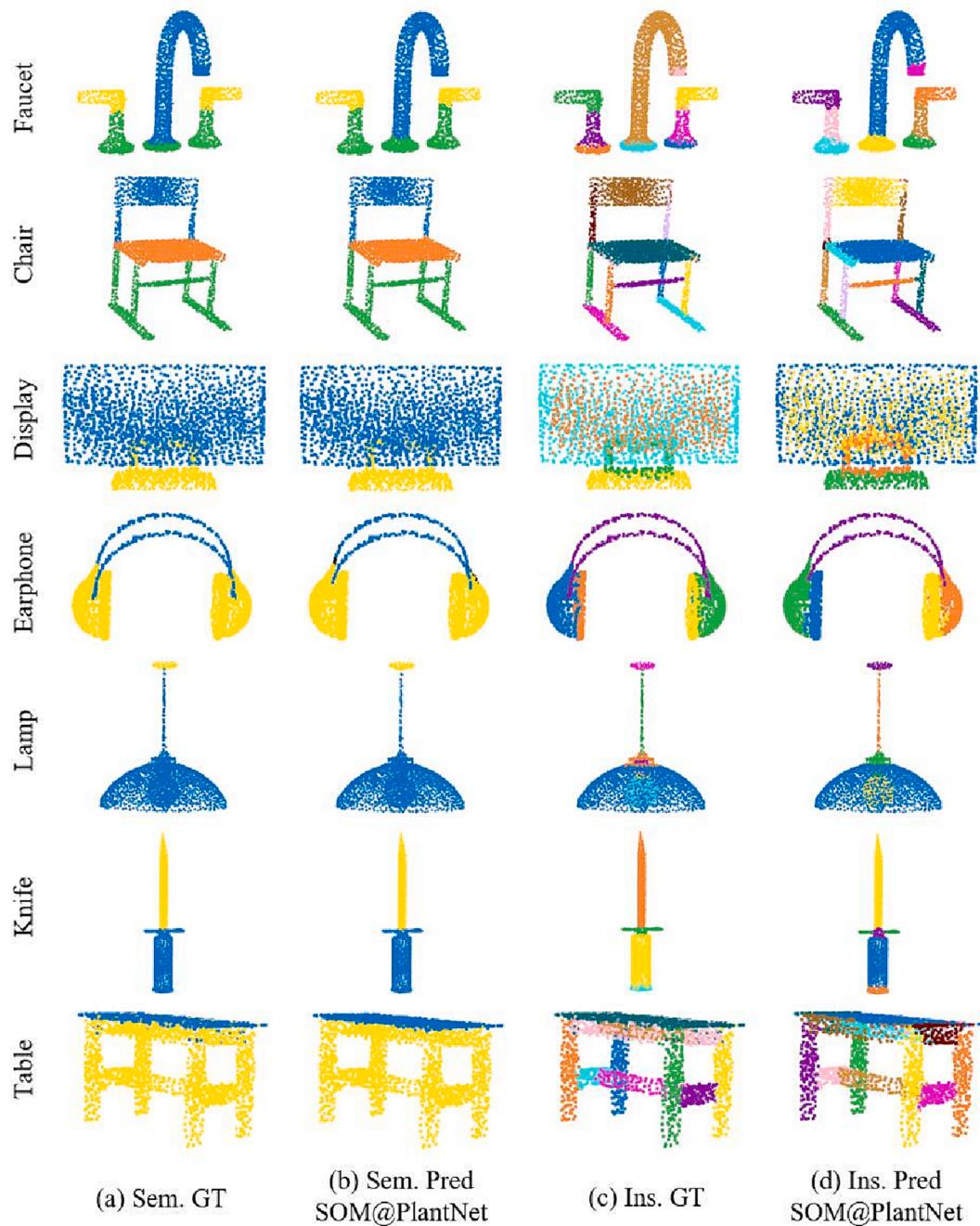
By examining the ‘‘Diff’’ data columns from Tables 2–5, it becomes clear that as the resolution decreases from 1024 points to 256 points, our strategy enjoys the least degradation on network segmentation performance across all compared strategies. In fact, on the contrary, the semantic segmentation performance of SOM even improves with decreasing resolution on the PointNet network. This proves that SOM is able to minimize feature loss even under very high compression rates. Across the strategies participated in the comparison, SOM constantly enjoys the best structural information preserving ability, which benefits network learning.

#### 4.4. Qualitative comparison with other strategies

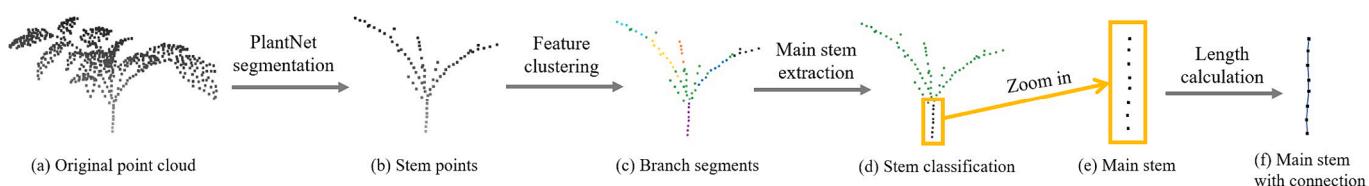
As the amount of experimental samples and results is large, we attempt to select representative qualitative results on only two networks for concise visualization. The qualitative results from DGCNN are selected to visualize the differences of down-sampling strategies on the semantic segmentation task, and the qualitative results from the PlantNet network are used to visualize the performance differences on the instance segmentation task. Limited by the paper length, this sub-section only reports the comparison of each plant species at only one sampling resolution for each network. Qualitative comparison of samples belonging to all three species at all three resolutions for both DGCNN and PlantNet on a broader scope can be found in Appendix A.2 and A.3. Fig. 11 illustrates the semantic segmentation performance comparison on chenopodium plant using DGCNN at 1024-point sampling resolution from two different views. Fig. 12 illustrates the semantic segmentation comparison on tomato plant using DGCNN at 512-point sampling resolution from two different views. Moreover, Fig. 13 displays the semantic segmentation comparison on tobacco plant using DGCNN at 256-point sampling resolution from two different views. Moving on to the instance segmentation task, Fig. 14 visualizes the instance segmentation performance comparison on chenopodium plant using PlantNet at 1024-point sampling resolution from two different views, and Fig. 15



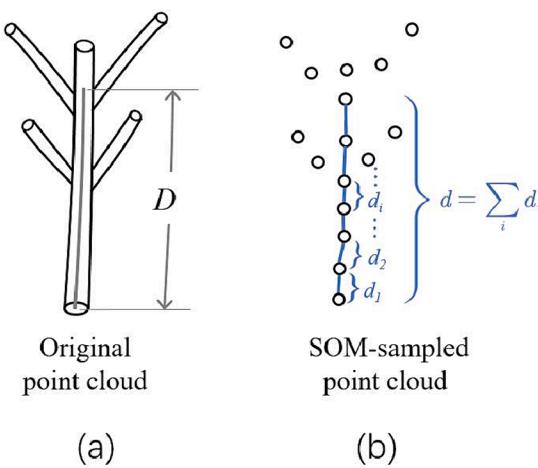
**Fig. 22.** Semantic and Instance segmentation results by PlantNet on the 1D SOM-sampled S3DIS dataset. The two rows in the figure are two different rooms, respectively. Each room is sampled to 2048 points. Column (a) is the original room point clouds with real color rendering; column (b) is the semantic GTs of the rooms; column (c) is the semantic segmentation results by PlantNet; column (d) is the GTs of instance segmentation; column (e) is the instance segmentation results predicted by PlantNet. Note that the colors in the instance segmentation results are used to distinguish different instances, and there is no fixed instance label correspondence with color.



**Fig. 23.** Semantic and instance segmentation results by PlantNet on the 1D SOM-sampled PartNet dataset. Each row stands for a 3D object, and each point cloud contains 2048 points. Column (a) is the semantic GTs of the objects; column (b) shows the semantic segmentation results predicted by PlantNet; column (c) is the GTs of instance segmentation; column (d) is the instance segmentation results predicted by PlantNet. Note that the colors in the instance segmentation results are used to distinguish different instances, and there is no fixed instance label correspondence with color.



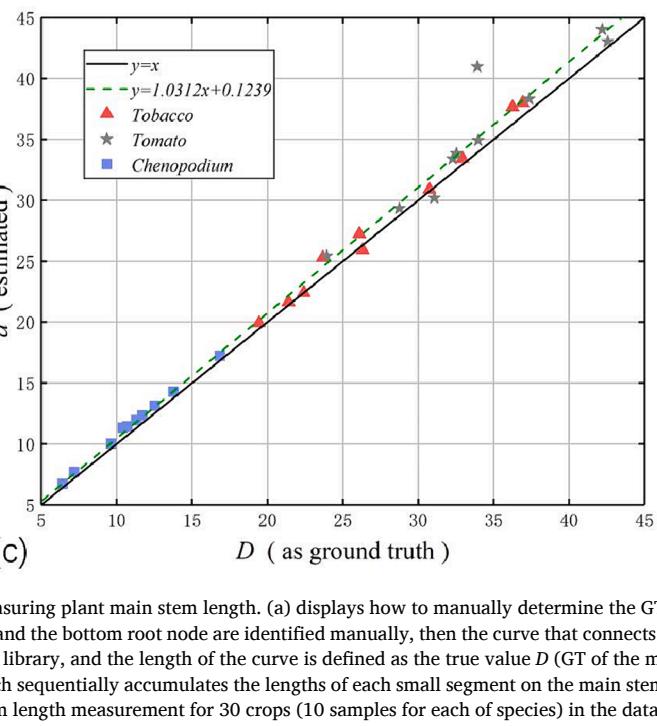
**Fig. 24.** The automated main stem length measurement pipeline based on 1D SOM-sampled plant point cloud. (a) is a SOM-sampled tomato crop point cloud containing 600 points; (b) is the stem system point set segmented by a pretrained PlantNet; (c) is the branch segments set by clustering on low-level geometric features of points; (d) classifies the set of line segments into the main stem and the lateral branches; (e) zooms in on the main stem region; (f) connects the points sequentially along the main stem nodes and measure the cumulated length.



**Fig. 25.** The accuracy regression of the automated 1D SOM-based pipeline for measuring plant main stem length. (a) displays how to manually determine the GT of main stem length on the original point cloud; first the uppermost bifurcation point and the bottom root node are identified manually, then the curve that connects the two points along the vertical stem is fitted by the least squares curve in the Python library, and the length of the curve is defined as the true value  $D$  (GT of the main stem length). (b) is the length calculation of the proposed automated pipeline, which sequentially accumulates the lengths of each small segment on the main stem to form the estimated length  $d$ . (c) is the regression of the scatter plot of the main stem length measurement for 30 crops (10 samples for each of species) in the dataset; the regression line is fitted and compared with the  $y = x$  line. Most of the points closely distribute around the  $y = x$  line.

visualizes the instance segmentation comparison on tomato plant using PlantNet at 512-point sampling resolution from two different views. Finally, Fig. 16 exhibits the instance segmentation comparison on tobacco plant using PlantNet at 256-point sampling resolution from two different views. It should be noted that the different colors in Figs. 14–16 are used just for differentiating adjacent leaf instances, and there is no instance-to-color correspondence. Upon analyzing Fig. 11 and Fig. 14, it becomes evident that the key of a good segmentation of chenopodium lies in the accurate recognition of the connection between the leaves class and the stem system. The use of SOM reduces the difficulty in segmentation by representing the leaves with surfaces while representing the stem with a line structure, which improves the segmentation accuracy of connection points. The results in Fig. 12 and Fig. 15 show that the structure of tomato plants is complex, the segmentation difficulties include irregularly-shaped compound leaves and naturally bent-down side branches. Our SOM clearly captures the bent trait of tomato branches by representing them with well-arranged linear nodes, exhibiting the fewest errors among all the compared strategies. The other three down-sampling strategies also generates a large number of segmentation errors at the junctions of compound leaves and side branches. As seen in Fig. 13 and Fig. 16, the challenges for segmentation of tobacco plants locate at the connections between the leaves and stems, as well as the edges of the overlapping leaves. Our SOM perfectly assists the networks in successfully separating the tobacco leaf edge points from branches, resulting in significantly fewer segmentation errors compared to the other three down-sampling strategies.

In general, the qualitative results show an evident edge of SOM sampling over other compared unsupervised sampling strategies. The reason is actually not hard to find. The main structure of a plant is mostly composed of the stems and the flat leaves. After the proposed SOM-based sampling, the stems can be abstracted by linearly connected nodes as lines in 3D, and the leaves can be represented by highly folded curves formed as connections of nodes. SOM automatically encodes preliminary semantics to different organ types; stems are sampled as “lines”, and leaves are sampled as “surfaces”. Thereby, SOM-based strategy is able to produce 3D data that can further improve the performance of deep networks for phenotypic tasks, because it reduces learning difficulty by embedding the preliminary semantic shapes (e.g.,



“line-shape”, and “surface-shape”) into training data. This shape-aware attribute of SOM also has biotaxonomic significance to botanists, because a practitioner can easily generate a SOM-sample standard model from any dense plant sample, and such model is easy to be further scrutinized for botanical purposes such as classifying and recognizing plant species.

To sum up, the shape-aware ability of SOM in point cloud down-sampling endows plant organs with different preliminary geometric semantic features. This enhances the discernibility between the organ classes of the leaves and the stem systems; thus, improves organ segmentation performance across multiple popular deep networks. This improvement in network performance is purely data-driven and is independent of the network architecture and training hyper-parameters, which shows a promising application prospect.

## 5. Discussion

### 5.1. Generalized shape-aware sampling and adaptive SOM sampling

This sub-section expands the discussion of the relationship between the competition layer topology of SOM and the shape-aware ability in sampling. Fig. 4 has exemplified the shape-aware perception of 1D SOM on the slender linear structure, the planar and curved surface structures, and the solid 3D objects in space during down-sampling. Based on these observations, we delved into the concept further with depictions in Fig. 5, which explores the generalization of the 1D competition layer to a 2D competition layer or even a 3D competition structure. We generalize Algorithm 1 to better down-sample planar (or curved surface-like) objects and solid 3D objects, respectively. The Algorithm 2, called 2D SOM for 2D-shape point set down-sampling, is given in detail in Appendix A.4. Similar to 1D SOM, Algorithm 2 can also be divided into two steps—competitive learning and refinement. The competitive learning step iteratively updates the neuron layer that is initialized as a regular 2D lattice, and the refinement step reduce the neuron number to exact  $M$ . In Fig. 17(a), we initialize a regular 2D planar lattice and perform Algorithm 2 on a piece of tobacco leaf. After the convergence of competitive learning, the 2D lattice evenly covers the entire leaf surface, forming a highly-ordered and uniform manifold learning output in

space. The Algorithm 3, called 3D SOM for 3D solid object down-sampling, is given in detail in [Appendix A.4](#). Algorithm 3 also contains two steps—competitive learning and refinement. Similar to the counterparts of its 1D and 2D version, the competitive learning of 3D SOM iteratively updates the neuron structure initialized as a 3D grid (cube), and the refinement step finally constrains the total down-sample point number to exact  $M$ . In [Fig. 17\(b\)](#), we use Algorithm 3 to sample a solid 3D object shaped as a cube. After convergence, the 3D grid naturally fills the cube, exhibiting a uniform and regular structure. The tests in [Fig. 17](#) prove the potential of utilizing specialized SOM competition layers to conduct shape-aware sampling for optimized point distribution.

Based on the above generalizations of SOM, an inspiration comes naturally to us. Since 1D, 2D, and 3D shapes can be well represented by 1D SOM, 2D SOM, and 3D SOM, respectively. Why not design a framework that automatically recognizes all shape types on the object and then applies the most suitable SOM algorithm to each shape for optimal sampling? Therefore, we design Algorithm 4 (in [Appendix A.4](#))—the adaptive SOM for point set down-sampling, to recognize different local shape patterns and adaptively apply corresponding SOM sampling on each of shape. Algorithm 4 can be divided into four steps. The first step uses likelihoods of three types of shapes  $L_i^{1D}, L_i^{2D}, L_i^{3D}$  ([Zhu et al., 2020](#)) to classify each point  $x_i$  with a shape label. The second step uses DBSCAN algorithm ([Ester et al., 1996](#)) to cluster all points into local subsets according to both shape label and the position of every point. Step 3 carries out the adaptive shape-aware down-sampling. If a local cluster contains 1D shape points, then we apply Algorithm 1 for sampling; if a cluster contains 2D shape points, we apply Algorithm 2, and we apply Algorithm 3 to the 3D shape cluster. All sampled clusters are combined to form the set  $\mathcal{W}$ . The final refinement step constrains the number of points in  $\mathcal{W}$  to exactly  $M$ . We test Algorithm 4 on several point clouds that contains different local shape types and the qualitative results are shown in [Fig. 18](#). The first and second row of [Fig. 18](#) show the between-step and final results of Algorithm 4 on a 3D stick figure point cloud, as well as the contrast with FPS sampling. The head, hands, and feet of the stick figure are solid sphere-like regions formed by dense points; the body of the figure is a planar region formed by points, while the arms and legs of the figure are curved pipes formed by dense points. Step 1 of Algorithm 4 correctly renders the stick figure with 1D, 2D, 3D local shape areas ([Fig. 18\(b1\)](#) and ([b2](#))) on which Step 2 then conducts satisfactory clustering by DBSCAN ([Fig. 18\(c1\)](#) and ([c2](#))). The final result of Algorithm 4 ([Fig. 18\(d1\)](#) and ([d2](#))) encodes the stick figure with implicit semantic shape information, e.g.—the limbs are represented by 1D linear objects, while the planar body is a 2D surface, and the rest of parts are solid point sets. The third to sixth rows of [Fig. 18](#) show the between-step and final results of Algorithm 4 on a tobacco and a tomato plant. Step 1 of Algorithm 4 correctly recognizes the planar leaves of both plants as 2D shape, and the stem segments as 1D shape. The complex 3D structures like buds ([Fig. 18\(b3\)](#)) and stem nodes ([Fig. 18\(b5\)](#)) are rendered as 3D shape. The final results of Algorithm 4 ([Fig. 18\(d3\)](#) to ([d6](#))) also encode crop structure with implicit semantic shape information, e.g.—the stems and branches are represented by 1D lines, while the planar leaves are represented by 2D surfaces.

## 5.2. Robustness tests

This sub-section discusses the robustness of competitive learning in improved 1D SOM (Algorithm 1) against noise and missing data. Due to factors such as the precision of point cloud scanning devices and the diversified environments, noise often appears in form of outliers in datasets. We expect a down-sampling strategy to possess a certain level of noise resistance, it should satisfy two requirements simultaneously—(i) the sampled morphology should adhere to the original noise-free data distribution, and (ii) the sampled point set include as few noise points as possible.

In order to compare the robustness of the improved 1D SOM-based

sampling with other strategies in the presence of noise, a “2”-shaped test point set is constructed in [Fig. 19](#). Aiming to reduce the number to 30 points, four down-sampling strategies are contrasted on the original point set, as well as on polluted point sets added with 1.0 %, 5.0 %, and 10.0 % added random noise, respectively. From the qualitative comparison results in [Fig. 19](#), it can be observed that the samplings of RS and 3DEPS are not uniform, and the sampled morphologies differs from the original “2” shape; both results fail to accurately describe the original data. As the noise level increases, FPS tends to sample outliers rather than the real points, causing noise-polluted results. In contrast, the proposed 1D SOM can still extract a correct “2” shape with high precision even under the highest presence of 10.0 % noise. Only 1 out of 30 points obtained belongs to the noise area, the noise rate even drops from 10.0 % to 3.3 % after SOM down-sampling. This demonstrates that our strategy has the highest resistance to outliers comparing to other unsupervised down-sampling strategies.

Additionally, to compare the robustness of the four down-sampling strategies against data missing, we design missing data cases upon the same “2”-shaped test point set in [Fig. 20](#). A certain proportion of points are randomly removed to simulate data incompleteness; we test with 25.0 %, 50.0 %, and 75.0 % of the points randomly removed from the original point set, respectively. Qualitative comparisons of down-sampling are then conducted with the objective of 10 points left after down-sampling. It is observed that RS and 3DEPS fail to obtain reliable shape representations of the original data. FPS and SOM are both less affected by the missing data and generally maintain the integrity of the “2” shape form. However, FPS is not as good as SOM in terms of evenness. In summary, our 1D SOM is more robust to noise and missing data than other compared unsupervised down-sampling methods.

## 5.3. Skeleton extraction

Inspired by the work of ([Chebrolu et al., 2021](#); [Magistri et al., 2020](#)) on skeletonizing crop point clouds, we explore possible extension of improved 1D SOM on skeleton extraction from tree-like point cloud in this sub-section. [Fig. 21](#) shows the comparison among 1D SOM, FPS, and L1-medial ([Huang et al., 2013](#)) (a popular skeleton extraction algorithm) on a high-precision tree-like coral point cloud. For all three compared methods, we used a same scheme to generate skeleton via nodes. The scheme first samples the point cloud to a fixed number of nodes, then connects the nodes by following several steps to form a skeleton. The specific steps for connecting nodes are—Step 1, perform KNN with  $K = 2$  on each node to identify the two nearest neighboring nodes, forming two outgoing arcs. Step 2, check the indegree of all nodes; and if a node has an indegree of 3 or more, perform KNN with  $K = 4$  on that node and add the nearest and the 4th nearest neighboring nodes as outgoing arcs; Step 3, visualize all the edges (arcs) to obtain the full skeleton (note that overlapping edges may exist). It can be seen from [Fig. 21\(b\)](#) that FPS generates many erroneous skeleton segments across empty areas, and the skeleton of L1-medial in [Fig. 21\(c\)](#) exhibits isolated areas in several parts, resulting in disconnections among the skeleton regions. In [Fig. 21\(d\)](#), the number of connection errors of SOM is less than those of FPS, and no isolated region is seen in our SOM skeleton. Therefore, 1D SOM may shed new insight on the design of new skeleton extraction algorithms on point clouds.

## 5.4. Applicability on other domains

This sub-section discusses the applicability of SOM-based sampling on datasets from other domains. The Stanford Large-Scale 3D Indoor Spaces (S3DIS) dataset ([Armeni et al., 2016](#)) and the PartNet dataset ([Mo et al., 2019](#)) (a 3D dataset containing many common objects) are tested. S3DIS consists of 6 large indoor areas scanned with high precision, including 271 rooms such as conference rooms, offices, and hallways. All points in S3DIS are divided into 13 semantic classes (floor, table, window, etc.) with both semantic and instance labels. In training

and testing of S3DIS, each room is subdivided into  $1 \times 1 \times h$  blocks in meter that do not overlap each other, where  $h$  is the height of each room. And then we use improved 1D SOM (Algorithm 1) to down-sample each block to 2048 points for standardizing all data. Each point holds 9-dimensional input features including the XYZ coordinates, RGB color, and a 3-dimensional normal vector. The testing dataset is Area 5, and the training set including all rooms except from Area 5. PlantNet model is used to carry out semantic and instance segmentation simultaneously on S3DIS. PartNet dataset has a total of 26,671 point clouds in 24 object classes. Each point cloud of PartNet contains XYZ coordinates information, and has both semantic and instance labels. We select 14 classes including Bag, Bowl, Faucet, Keyboard, Laptop, Mug, Chair, Display, Earphone, Hat, Knife, Lamp, Scissors, and Table for evaluation, and divide the training set, validation set, and testing set with a proportion of 7:1:2. The exact training data contains 14,179 point clouds, and the testing data contains 4031 point clouds. All PartNet point clouds are sampled with Algorithm 1 to 2048 points and then fed into PlantNet for semantic and instance segmentation.

Fig. 22 shows visualized semantic and instance segmentation results by PlantNet for two 1D SOM-sampled S3DIS rooms. Our strategy seems to be well suited to 3D indoor room data, especially helpful to segment furniture such as chairs. Fig. 23 shows visualized semantic and instance segmentation results for several PartNet objects by PlantNet, and the predictions by PlantNet are close to GTs, respectively. The prediction results show that the improved 1D SOM sampling can be applied to 3D deep learning tasks that serve different other domains, proving its good generalization ability on data.

### 5.5. Main stem length measurement

SOM-based sampling strategy also can be used directly for plant phenotyping. In this sub-section, we employ Algorithm 1 to measure the main stem length of the three crop species in our dataset. An automated main stem length measurement pipeline (shown in Fig. 24) is proposed based on 1D SOM sampling. It involves several steps. First, a semantic segmentation network—PlantNet (or any stem detection method) is used to separate stem system from the 1D SOM-sampled crop point cloud (Fig. 24(a)), in this case we have used the PlantNet to conduct stem system segmentation. Second, clustering is performed based on the low-level geometric features of the segmented stem system points (Fig. 24(b)) to generate a set of branch segments (Fig. 24(c)). Third, the main stem (Fig. 24(d)) is identified among all stem/branch segments by considering the vertical growth direction and the position of the root node. Then, all segments can be classified into two types: the main stem and the lateral branches. At last, we connect all nodes on the main stem sequentially to obtain the final linear representation (Fig. 24(f)). The length of each small line segment is accumulated to calculate the main stem length.

To evaluate the accuracy of the length measurement, we compare the measured main stem length  $d$  by the above pipeline with corresponding true length  $D$  on a number of samples from our three-species dataset. How to calculate both  $d$  and  $D$  are illustrated in Fig. 25(a). The regression result of the proposed main stem length measurement pipeline is presented in Fig. 25(b), which demonstrates satisfactory measurements on the main stem length across three plant species with a number of samples. The crop point cloud obtained via our 1D SOM-based sampling effectively abstracts the linear stem segments from the topological structure (Fig. 25(c)); SOM naturally guarantees precise calculation of phenotypic indicators such as stem length, leaf number, and even leaf area, from the data level at the very beginning.

### 5.6. Implications

This SOM-based research may be further extended by focusing on the following three aspects. First, we have noticed that the SOM-based sampling did not show evident advantage over FPS on segmenting

monocotyledonous crops such as sorghum in some tests. The reason may be that the monocotyledonous crops usually have long and slender leaves that are down-sampled by SOM as simple lines just similar to the stem segments when the objective number of points is small. This results in ambiguity in encoded low-level semantics; i.e., when a leaf and the stem connected to itself are both represented by a single line of nodes, it is hard to teach deep networks to distinguish leaf from stem because the difference in point distribution is missing. Therefore, space still remains to improve the SOM-based sampling for learning morphological characteristics of different organs on monocotyledonous crops. Second, since we have only tested only a small portion of 3D data in the field, expanding the application of the SOM-based sampling to a wider range of crop species as well as exploring the possibility of application on 3D data from other domains are worth trying. Third, considering the shape-aware sampling ability of generalized SOMs and the adaptive SOM, they may show good prospect on application domains where a large amount of 3D point cloud data exists, such as digital cities (Lafarge and Mallet, 2012), autonomous driving (Li et al., 2021b), etc. Following this way, the more generalized versions—high-dimensional SOMs may become a future member of the SOM family for analyzing high-dimensional data space, which can facilitate research in the field of data analysis and optimization.

## 6. Conclusion

This paper proposes an unsupervised down-sampling strategy family based on Self-Organizing Maps that originates from the biological Lateral Inhibition mechanism. Our strategy has a fast and iterative framework capable of precisely control over the point number after down-sampling. The good shape-aware attribute empowers SOM-based sampling to automatically preserves important low-level spatial features. Specifically, when applying the improved 1D SOM to plant point clouds, the stems can be abstracted by linearly connected nodes as lines in 3D, and the leaves can be represented by highly folded curves formed by connection of nodes. SOM-based strategy is able to produce 3D data that can further improve the performance of deep networks for plant phenotyping tasks, because it reduces learning difficulty by embedding the preliminary semantic shapes (e.g., 1D shape and 2D planar shape) into training data. This shape-aware attribute of SOM also has biotaxonomic significance to botanists, because a practitioner can easily generate a SOM-sample standard model from any dense plant sample, and such model is easy to be further scrutinized for botanical purposes such as classifying and recognizing plant species. The experiments on several different segmentation networks have shown that SOM performs better than popular down-sampling strategies both qualitatively and quantitatively, and SOM particularly benefits fine-grained organ instance segmentation. Furthermore, SOM-based sampling shows potential in versatility; for example, it can be applied to point cloud skeleton extraction, main stem length measurement on crops, and deep learning-based segmentation on datasets from other domains.

## CRediT authorship contribution statement

**Dawei Li:** Conceptualization, Funding acquisition, Methodology, Supervision, Writing – original draft, Writing – review & editing. **Zhaoyi Zhou:** Data curation, Formal analysis, Methodology, Software, Visualization, Writing – original draft, Writing – review & editing. **Yongchang Wei:** Data curation, Software, Validation, Visualization.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

**Acknowledgement**

Grant No. 21QA1400100.

This research is supported by Shanghai Rising-Star Program under

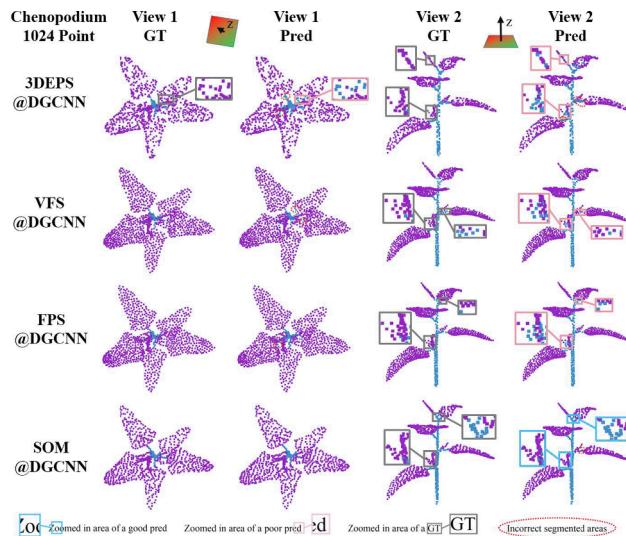
**Appendix A****A.1. Notations and nomenclature****Table A1**

Notations and nomenclature used in this paper.

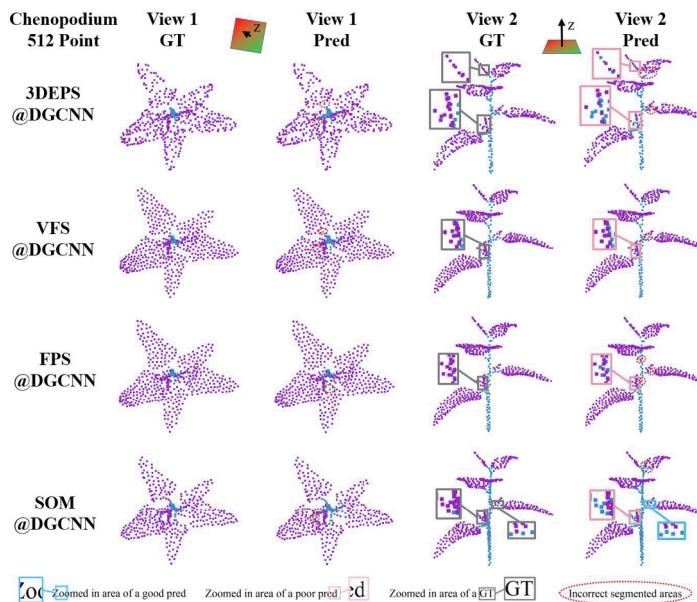
SOM	Self-organizing Map
FPS	Farthest Point Sampling
VBS	Voxel-based Sampling
CNN	Convolutional Neural Network
MVCNN	Multi-view Convolutional Neural Network
FC-CRF	Fully-connected Conditional Random Field
VIT	Vision Transformer
PCT	Point Cloud Transformer
3DEPS	3D Edge-preserving Sampling
RS	Random Sampling
US	Uniform Sampling
VFPS	Voxelized Farthest Point Sampling
NSS	Normal Space Sampling
LI	Lateral Inhibition
GSRMs	Growing Self-reconstruction Maps
SOMN	Self-organizing Mixture Network
ViSOM	Visualization-induced SOM
TSP	Time Series Prediction
SSE	Semantic Segmentation Editor
GT	Ground truth
IoU	Intersection over Union
TP	True Positive
FP	False Positive
FN	False Negative
Prec	Precision
Rec	Recall
mCov	The mean coverage
mWCov	The mean weighted coverage
Diff	The average performance difference between the current strategy and the best
PCA	Principle Component Analysis
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
S3DIS	Stanford Large-Scale 3D Indoor Spaces
KNN	K-nearest neighbors
$\mathcal{P}, \mathcal{X}$	The original point set or point cloud
$\mathcal{A}$	A down-sampled point set
$\mathcal{P}/\mathcal{A}$	The point set $\mathcal{P}$ without points listed in $\mathcal{A}$
$l_x, l_y, l_z$	Three side lengths of the voxel
$\mathcal{N}$	The activation field in the competition learning of SOM
$N$	The number of points in the original point cloud
$M$	The number of points after down-sampling
$x_i$	The $i$ -th point in the point set
$\mathcal{W}$	The neuron weight set of SOM (also the output)
$w_j$	The $j$ -th neuron of the output of SOM
MIN	Maximum iteration number of SOM
$\mathcal{I}, \mathcal{O}$	Neuron subsets
$\lambda^{(t)}, \sigma^{(t)}$	Parameters used in competitive learning at the $t$ -th iteration
$\alpha^{(t)}$	Activation function value at the $t$ -th iteration
$\delta(w_i, w_j)$	The distance between two neurons on the competition layer
$\mathcal{N}e(w_i)$	The closest neighbors of a neuron $w_i$ on the competition layer
$s$	The side length of the initialized 2D neuron lattice or the 3D grid
$\beta$	A parameter controls the number of neighbor points in a sphere
$d_{ave}$	The average spacing of the point set $\mathcal{X}$
$\mathcal{R}_i$	All points in the neighbor sphere of $x_i$
$\lambda_{i,1}, \lambda_{i,2}, \lambda_{i,3}$	Three descending PCA eigenvalues of $\mathcal{R}_i$
$L_i^{1D}, L_i^{2D}, L_i^{3D}$	likelihoods of three types of shapes for each point $x_i$
$l_i$	The shape label of $x_i$
$\lfloor \cdot \rfloor$	The round-up operator
$M_j$	The number of down-sampled points for the cluster $\mathcal{X}_j$
$\mu, h_0, \Delta h$	Three parameters used in L1-medial algorithm
$d$	Estimated main stem length
$D$	The ground truth of the main stem length
$I$	The number of all instances contained in a semantic category of GT
$IG_m$	The region of the $m$ -th instance in the current semantic category of GT
$IP_n$	The region of the $n$ -th instance predicted by the network

#### A.2. Qualitative comparisons across plant point clouds sampled by different strategies on the organ semantic segmentation task by DGCNN

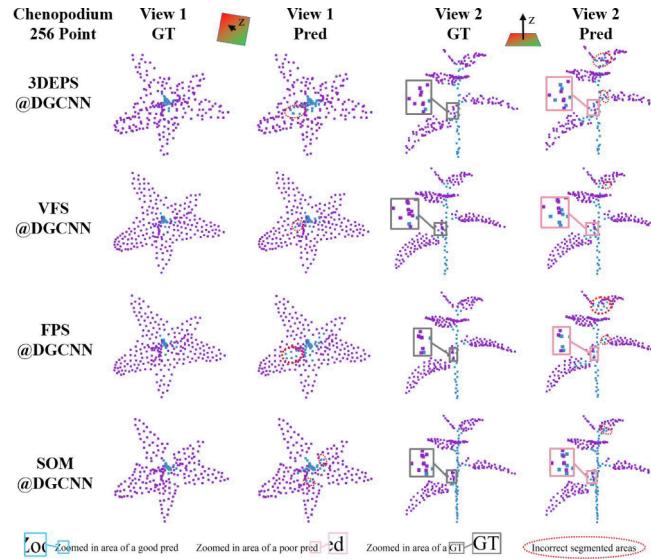
In this sub-section, we provide more detailed comparisons of the qualitative results for semantic segmentation by DGCNN network under different down-sampling strategies. The qualitative results are displayed in the following order: 1024-point resolution for a chenopodium crop (Fig. A1), 512-point resolution for the same chenopodium crop (Fig. A2), 256-point resolution for the same chenopodium crop (Fig. A3); 1024-point sampling resolution for a tomato crop (Fig. A4), 512-point resolution for the same tomato crop (Fig. A5), 256-point sampling resolution for the same tomato crop (Fig. A6); 1024-point resolution for a tobacco crop (Fig. A7), 512-point resolution for a tobacco crop (Fig. A8), and 256-point resolution for a tobacco crop (Fig. A9). In Figs. A1–A9, most of the qualitative comparison results demonstrate the superiority of the proposed SOM-based sampling strategy with less segmentation errors. Note that the DGCNN network only performs semantic segmentation.



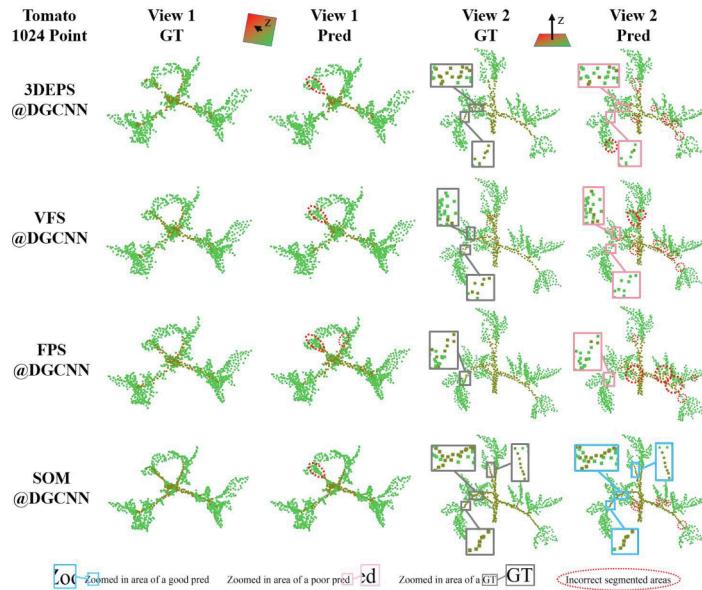
**Fig. A1.** The qualitative semantic segmentation results of a chenopodium crop by DGCNN at the 1024-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details. This figure is also the Fig. 11 in the main body of the paper.



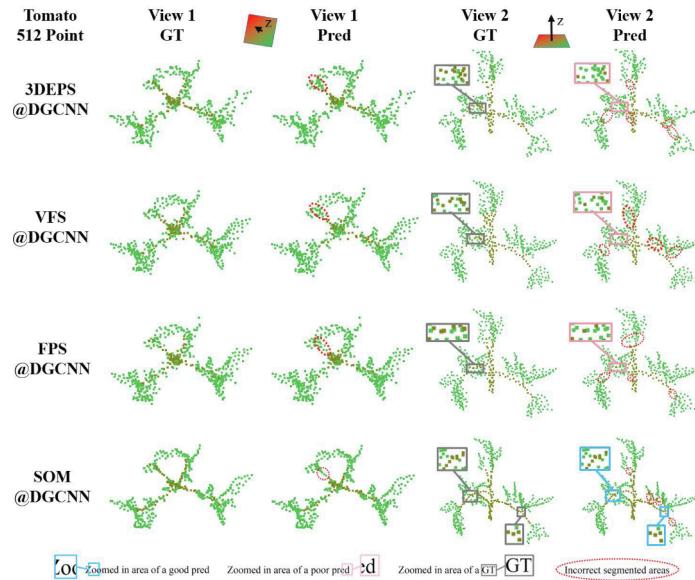
**Fig. A2.** The qualitative semantic segmentation results of a chenopodium crop by DGCNN at the 512-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details.



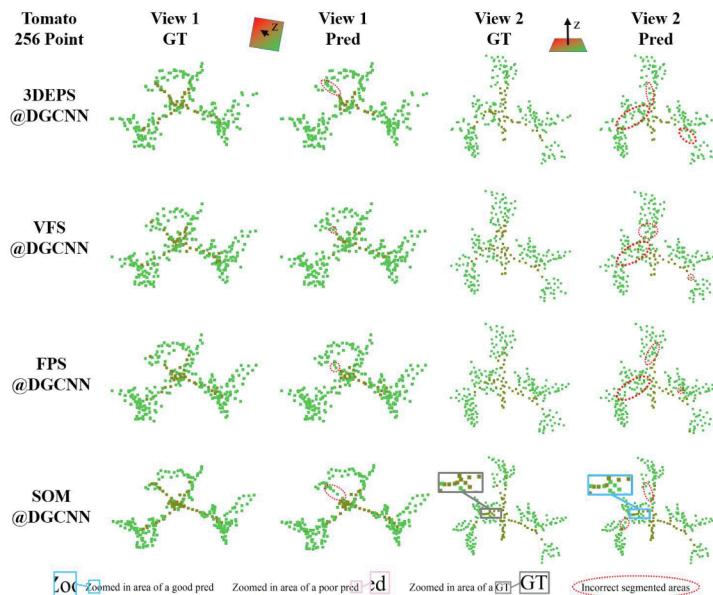
**Fig. A3.** The qualitative semantic segmentation results of a chenopodium crop by DGCNN at the 256-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details.



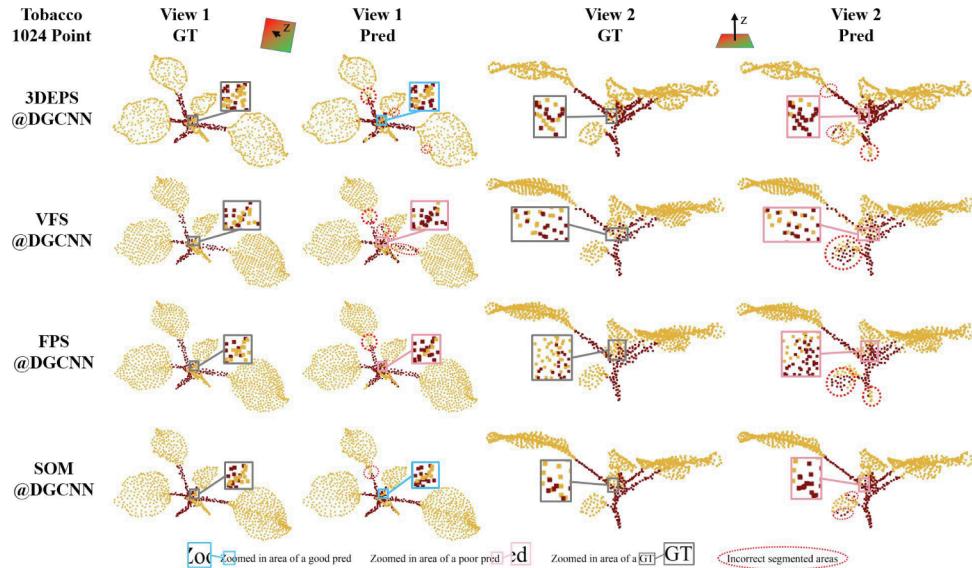
**Fig. A4.** The qualitative semantic segmentation results of a tomato crop by DGCNN at the 1024-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details.



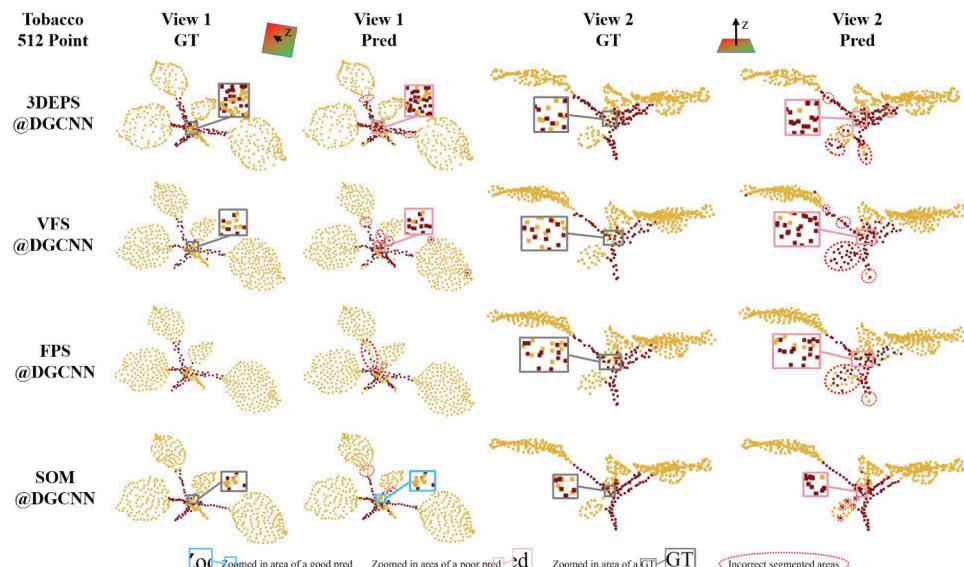
**Fig. A5.** The qualitative semantic segmentation results of a tomato crop by DGCNN at the 512-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details. This figure is also the Fig. 12 in the main body of the paper.



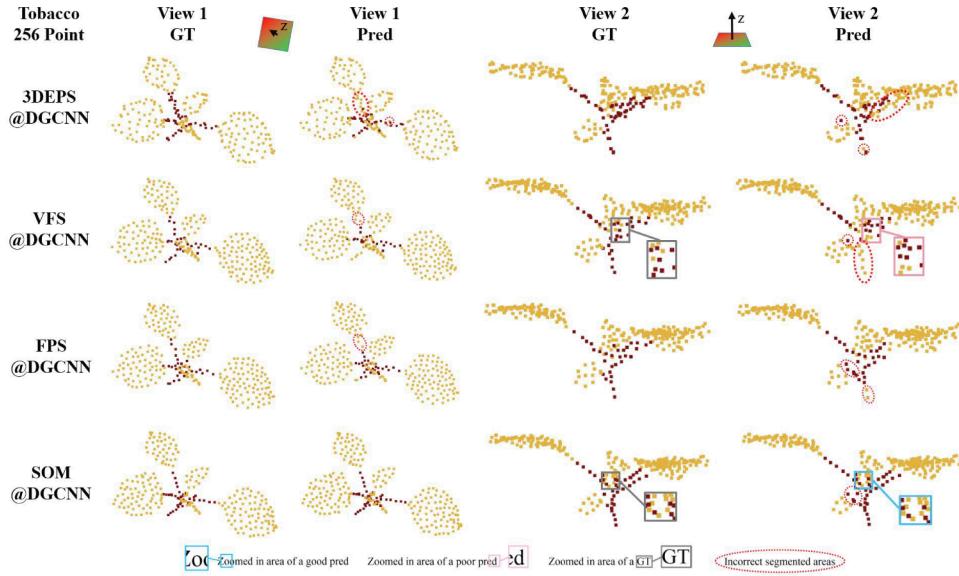
**Fig. A6.** The qualitative semantic segmentation results of a tomato crop by DGCNN at the 256-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details.



**Fig. A7.** The qualitative semantic segmentation results of a tobacco crop by DGCNN at the 1024-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details.



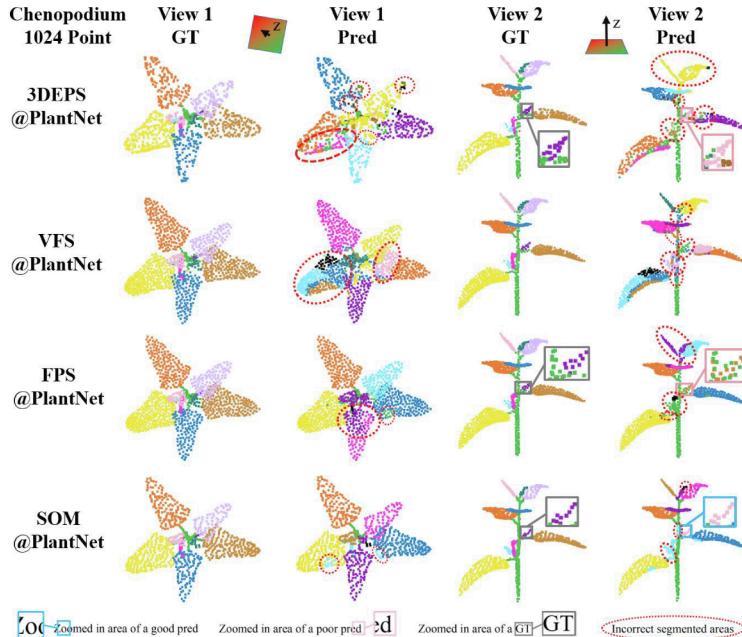
**Fig. A8.** The qualitative semantic segmentation results of a tobacco crop by DGCNN at the 512-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details.



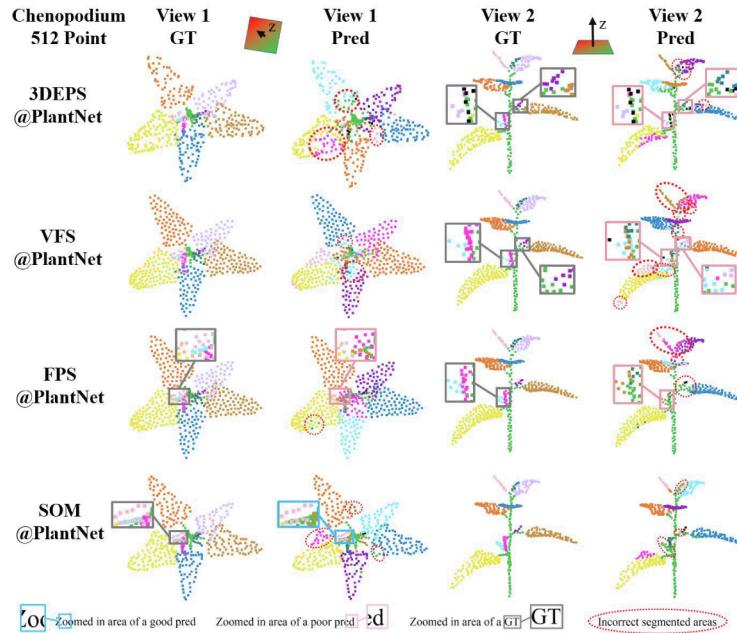
**Fig. A9.** The qualitative semantic segmentation results of a tobacco crop by DGCNN at the 256-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details. This figure is also the Fig. 13 in the main body of the paper.

#### A.3. Qualitative comparisons across plant point clouds sampled by different strategies on the organ instance segmentation task by PlantNet

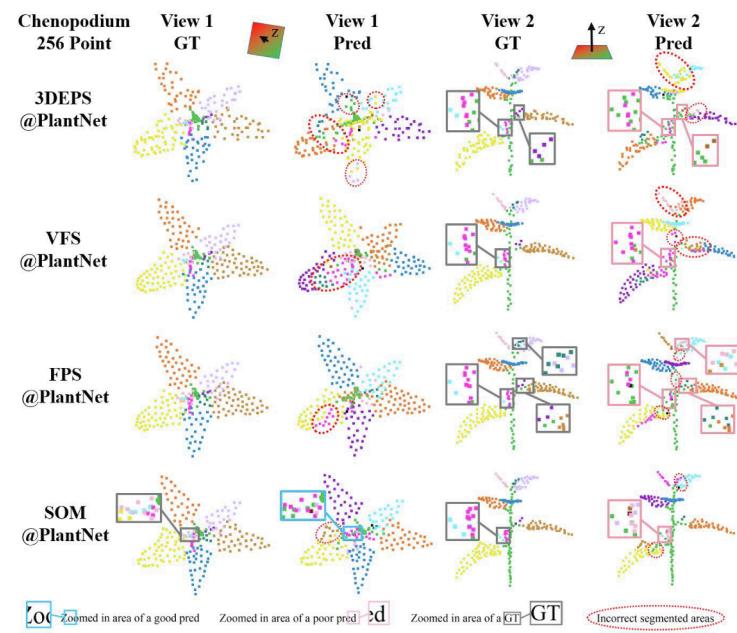
In this sub-section, we provide more detailed comparisons of the qualitative results for leaf organ instance segmentation by the PlantNet network under different down-sampling strategies. The qualitative results are displayed in the following order: 1024-point sampling resolution for a chenopodium crop (Fig. A10), 512-point resolution for the same chenopodium crop (Fig. A11), 256-point resolution for the same chenopodium crop (Fig. A12); 1024-point resolution for a tomato crop (Fig. A13), 512-point resolution for the same tomato crop (Fig. A14), 256-point sampling resolution for the same tomato crop (Fig. A15); and 1024-point resolution for a tobacco crop (Fig. A16), 512-point resolution for the same tobacco crop (Fig. A17), and 256-point resolution for the same tobacco crop (Fig. A18). In Figs. A10–A18, most of the qualitative comparison results demonstrate the superiority of the proposed SOM-based sampling strategy with less segmentation errors. Note that though the PlantNet network can perform both semantic and instance segmentation tasks, we only show the instance segmentation results to avoid redundancy with DGCNN predictions.



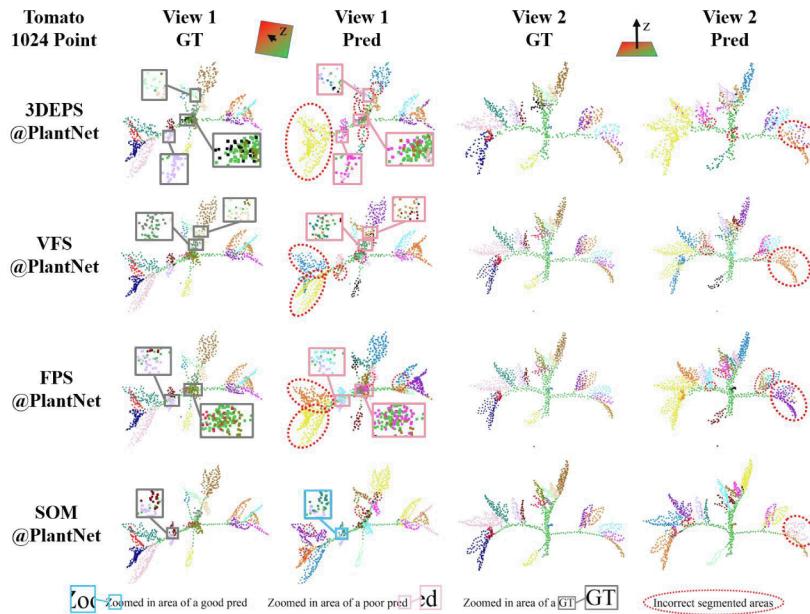
**Fig. A10.** The qualitative semantic segmentation results of a chenopodium crop by PlantNet at the 1024-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details. Different colors are designed to differentiate adjacent leaf instances, and there is no fixed color-label correspondence. This figure is also the Fig. 14 in the main body of the paper.



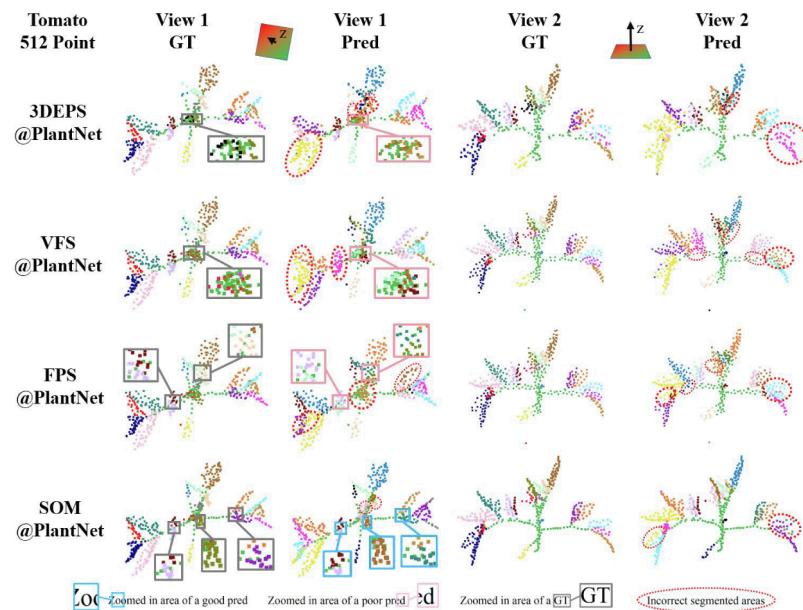
**Fig. A11.** The qualitative semantic segmentation results of a chenopodium crop by PlantNet at the 512-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details.



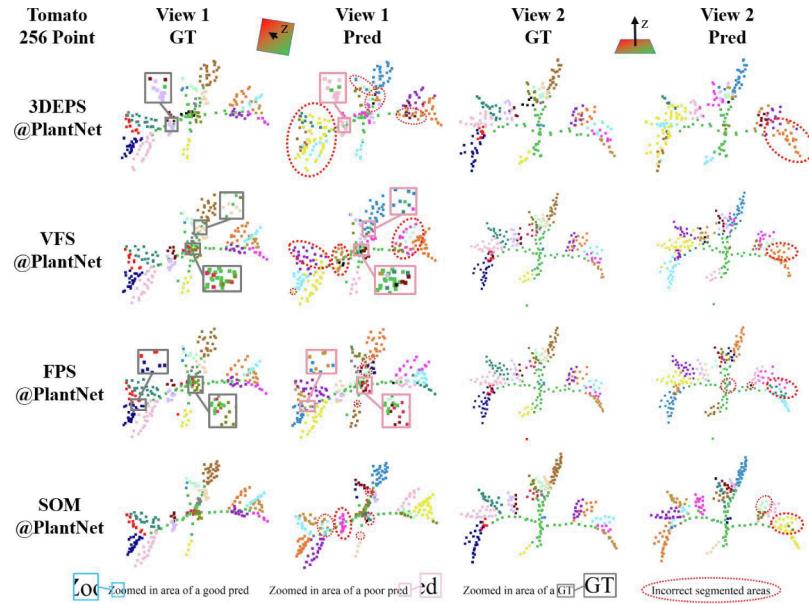
**Fig. A12.** The qualitative semantic segmentation results of a chenopodium crop by PlantNet at the 256-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details.



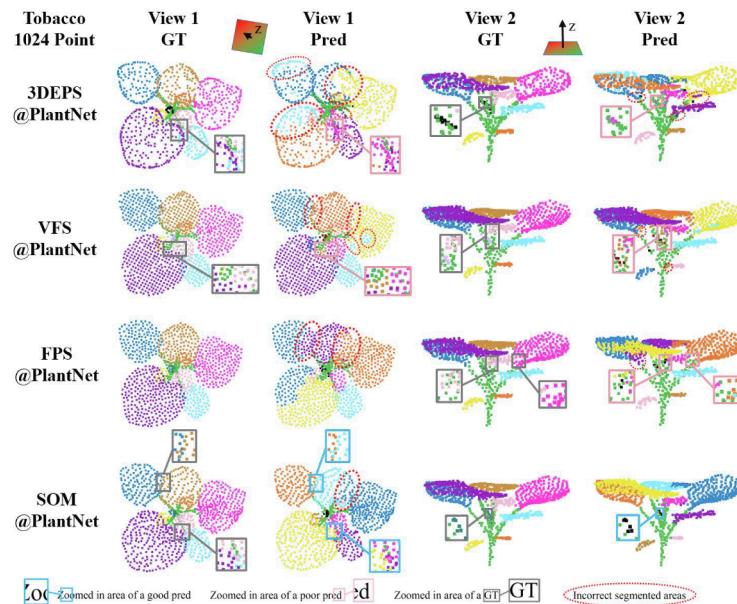
**Fig. A13.** The qualitative semantic segmentation results of a tomato crop by PlantNet at the 1024-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details. Different colors are designed to differentiate adjacent leaf instances, and there is no fixed color-label correspondence.



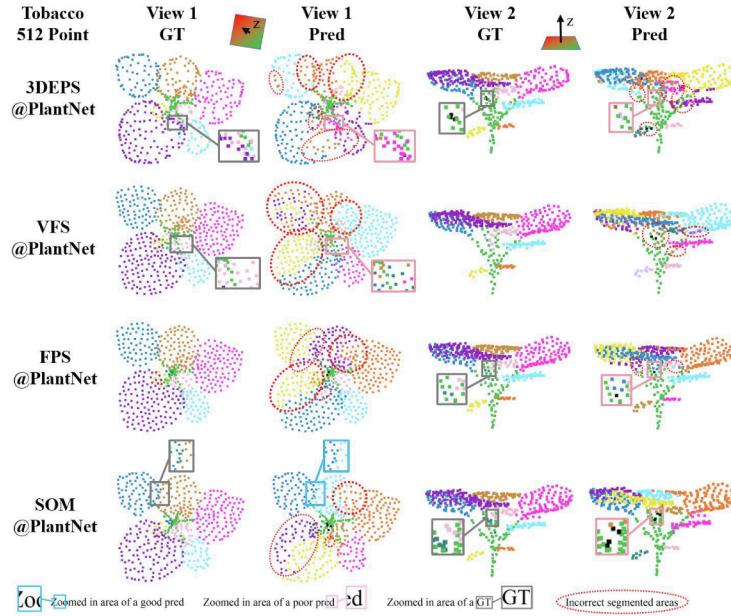
**Fig. A14.** The qualitative semantic segmentation results of a tomato crop by PlantNet at the 512-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details. This figure is also the Fig. 15 in the main body of the paper.



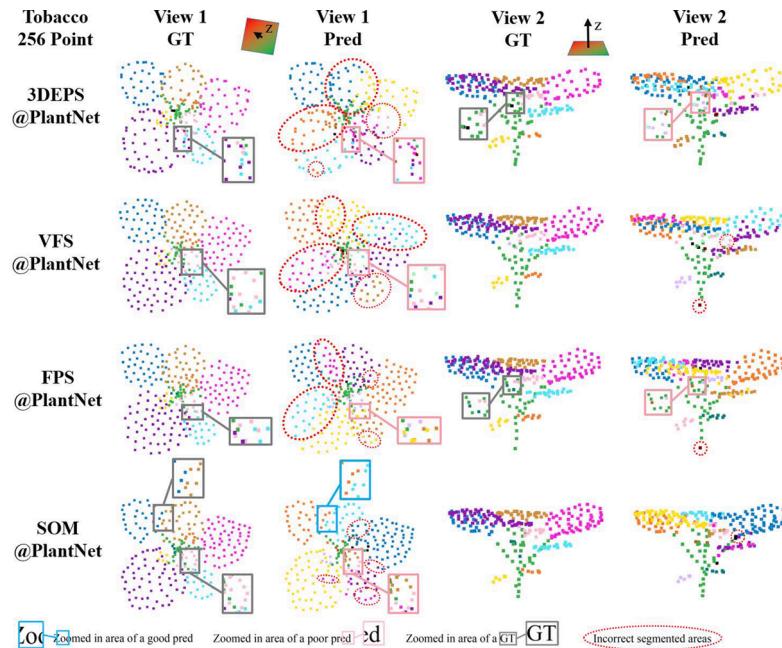
**Fig. A15.** The qualitative semantic segmentation results of a tomato crop by PlantNet at the 256-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details.



**Fig. A16.** The qualitative semantic segmentation results of a tobacco crop by PlantNet at the 1024-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details. Different colors are designed to differentiate adjacent leaf instances, and there is no fixed color-label correspondence.



**Fig. A17.** The qualitative semantic segmentation results of a tobacco crop by PlantNet at the 512-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details.



**Fig. A18.** The qualitative semantic segmentation results of a tobacco crop by PlantNet at the 256-point resolution under four different down-sampling strategies. The same crop is visualized from two different views—View 1 and View 2. Pred is the segmentation result predicted by the network and GT is the Ground Truth labels manually annotated. We zoomed in on some key areas for a better look on details. This figure is also shown as the Fig. 16 in the main body of the paper.

#### A.4. Detailed algorithms of generalized SOM sampling

**Table A2**

The pseudocode of 2D SOM for 2D-shape point set down-sampling.

**Algorithm 2: 2D SOM for 2D-shape point set down-sampling**

**Input:** Data set with  $N$  points  $\mathcal{X} = \{x_1, \dots, x_N\}$ , Number of points after down-sampling  $M$ .

**Output:** Neuron weight set  $\mathcal{W}$ .

**Initialization:** Build a 2D neuron lattice  $\mathcal{W} = \{w_1, \dots, w_{s^2}\}$  whose elements are initialized as nodes on a grid. The lattice center is aligned with the center of the point set. The side length of the lattice is set as half of the data scale measured along the longest PCA direction. The value  $s$  is fixed as the minimum integer that satisfies  $s^2 \geq M$ . Each inside neuron is connected to 4 neighbors. The distance of each neuron to its neighbors  $\delta(w_i, e(w_i))$  is set to 1, and the distance of two neurons  $\delta(w_i, w_j)$  is defined as the shortest path on the lattice between  $w_i$  and  $w_j$ .

(continued on next page)

**Table A2 (continued)****Algorithm 2: 2D SOM for 2D-shape point set down-sampling**


---

Maximum iteration number is defined as  $MIN = 5N$ .  
 $\lambda^{(1)} = 0.5$ ;  $\sigma^{(1)} \in [1, 1.5]$ ;  $t = 1$ .

**I. Competitive learning:**

- 1 While  $t < MIN$  do
- 2 Draw a point  $x_t$  from  $\mathcal{X}$ .
- 3 For  $j \in [1, \dots, s^2]$  do
- 4 Find winning neuron  $w_{j^*}^{(t)}$  with  $j^* = \operatorname{argmin}_j \|x_t - w_j^{(t)}\|$
- 5 End for
- 6 For  $j \in [1, \dots, s^2]$  do
- 7  $\alpha_j^{(t)} = \exp\left[-\delta^2(w_j^{(t)}, w_{j^*}^{(t)})/(2\sigma^{(t)} \cdot \sigma^{(t)})\right]$ ,
- 8  $w_j^{(t+1)} \leftarrow w_j^{(t)} + \lambda^{(t)} \alpha_j^{(t)} (x_t - w_{j^*}^{(t)})$ .
- 9 End for
- 10  $\lambda^{(t+1)} \leftarrow \lambda^{(t)} / (1 + 2t/MIN)$ ;
- 11  $\sigma^{(t+1)} \leftarrow \sigma^{(t)} / (1 + 2t/MIN)$ ;
- 12  $t \leftarrow t + 1$ ;
- 13 End while

**II. Refinement:**

- 14 Randomly delete  $s^2 - M$  nodes from  $\mathcal{W}$ .

**Table A3**

The pseudocode of 3D SOM for 3D solid object down-sampling.

**Algorithm 3: 3D SOM for 3D solid object down-sampling**


---

**Input:** Point set with  $N$  points  $\mathcal{X} = \{x_1, \dots, x_N\}$ , Number of points after down-sampling  $M$ .  
**Output:** Neuron weight set  $\mathcal{W}$ .

**Initialization:** Build a 3D neuron grid  $\mathcal{W} = \{w_1, \dots, w_{s^3}\}$  whose elements are initialized as nodes in a cube. The grid center is aligned with the center of the object. The side length of the cubic is set as half of the data scale measured along the longest PCA direction. The value  $s$  is fixed as the minimum integer that satisfies  $s^3 \geq M$ . Each inside neuron is connected to 6 neighbors. The distance of each neuron to its neighbors  $\delta(w_i, \mathcal{N}(w_i))$  is set to 1, and the distance of two neurons  $\delta(w_i, w_j)$  is defined as the shortest path on the lattice between  $w_i$  and  $w_j$ .  
Maximum iteration number  $MIN = 5N$ .  
 $\lambda^{(1)} = 0.5$ ;  $\sigma^{(1)} \in [1, 1.5]$ ;  $t = 1$ .

**I. Competitive learning:**

- 1 While  $t < MIN$  do
- 2 Draw a point  $x_t$  from  $\mathcal{X}$ .
- 3 For  $j \in [1, \dots, s^3]$  do
- 4 Find winning neuron  $w_{j^*}^{(t)}$  with  $j^* = \operatorname{argmin}_j \|x_t - w_j^{(t)}\|$
- 5 End for
- 6 For  $j \in [1, \dots, s^3]$  do
- 7  $\alpha_j^{(t)} = \exp\left[-\delta^2(w_j^{(t)}, w_{j^*}^{(t)})/(2\sigma^{(t)} \cdot \sigma^{(t)})\right]$ ,
- 8  $w_j^{(t+1)} \leftarrow w_j^{(t)} + \lambda^{(t)} \alpha_j^{(t)} (x_t - w_{j^*}^{(t)})$ .
- 9 End for
- 10  $\lambda^{(t+1)} \leftarrow \lambda^{(t)} / (1 + 2t/MIN)$ ;
- 11  $\sigma^{(t+1)} \leftarrow \sigma^{(t)} / (1 + 2t/MIN)$ ;
- 12  $t \leftarrow t + 1$ ;
- 13 End while

**II. Refinement:**

- 14 Randomly delete  $s^3 - M$  nodes from  $\mathcal{W}$ .

**Table A4**

The pseudocode of Adaptive SOM sampling.

**Algorithm 4: Adaptive SOM for point set down-sampling**


---

**Input:** Point set with  $N$  points  $\mathcal{X} = \{x_1, \dots, x_N\}$ , Number of points after down-sampling  $M$ .  
**Output:** Neuron weight set  $\mathcal{W}$ .

**Initialization:** Shape label set  $\mathcal{L} = \{l_1, \dots, l_N\}$ ; calculate the average spacing  $d_{ave}$  of the point set; parameter  $\beta \in [5.0, 25.0]$ ;

**I. Filtering:**

- 1 For  $i \in [1, \dots, N]$  do
- 2 Find neighboring points of  $x_i$  within radius  $\beta \cdot d_{ave}$  in  $\mathcal{X}$  as  $\mathcal{R}_i$
- 3 Do PCA on  $\mathcal{R}_i$ ; obtain three descending eigenvalues as  $\lambda_{i,1}, \lambda_{i,2}, \lambda_{i,3}$
- 4 Compute the likelihood of shapes for each point:
- 5  $L_i^{1D} = (\sqrt{\lambda_{i,1}} - \sqrt{\lambda_{i,2}}) / \sqrt{\lambda_{i,1}}$
- 6  $L_i^{2D} = (\sqrt{\lambda_{i,2}} - \sqrt{\lambda_{i,3}}) / \sqrt{\lambda_{i,1}}$
- 7  $L_i^{3D} = \sqrt{\lambda_{i,3}} / \sqrt{\lambda_{i,1}}$

(continued on next page)

**Table A4 (continued)**


---

**Algorithm 4: Adaptive SOM for point set down-sampling**

---

```

8   Set shape label  $l_i \leftarrow 0$ 
9   End for
10  For  $i \in [1, \dots, N]$  do
11    If  $(L_i^{3D} > L_i^{2D}) \cap (L_i^{3D} > L_i^{1D})$ 
12       $l_i \leftarrow 3$ ;  $x_i$  is classified as a point from 3D shape
13    Else If  $(L_i^{1D} > 2.5 \cdot L_i^{2D}) \cap (L_i^{1D} > L_i^{3D})$ 
14       $l_i \leftarrow 1$ ;  $x_i$  is classified as a point from 1D shape
15    Else
16       $l_i \leftarrow 2$ ;  $x_i$  is classified as a point from 2D shape
17    End If
18  End If
19 End for

II. Clustering
20 Do DBSCAN on  $\mathcal{X}$  according to both  $l_i$  and  $x_i$ ;
 $\mathcal{X}$  is divided into clusters as  $\{\mathcal{X}_1, \mathcal{X}_2, \dots, \mathcal{X}_O\}$ 
21 For  $j \in [1, \dots, O]$ 
22 Determine the number of points after down-sampling for each cluster;
 $M_j = \lfloor M \cdot |\mathcal{X}_j| / |\mathcal{X}| \rfloor$ ; ( $\lfloor \cdot \rfloor$  is round-up)
23 End For

III. Adaptive shape-aware down-sampling:
24 For  $j \in [1, \dots, O]$ 
25   If any point  $x_i \in \mathcal{X}_j$  has  $l_i = 1$ 
26     Do Algorithm 1 on  $\mathcal{X}_j$  to generate  $\mathcal{W}_j$  that has  $M_j$  points;
27   If any point  $x_i \in \mathcal{X}_j$  has  $l_i = 2$ 
28     Do Algorithm 2 on  $\mathcal{X}_j$  to generate  $\mathcal{W}_j$  that has  $M_j$  points;
29   If any point  $x_i \in \mathcal{X}_j$  has  $l_i = 3$ 
30     Do Algorithm 3 on  $\mathcal{X}_j$  to generate  $\mathcal{W}_j$  that has  $M_j$  points;
31   End If
32 End for
33  $\mathcal{W} \leftarrow \{\mathcal{W}_1, \dots, \mathcal{W}_O\}$ 

IV. Refinement:
34 Randomly delete  $\sum_{j=1}^O M_j - M$  nodes from  $\mathcal{W}$ 

```

---

## References

- Ando, R., Ozasa, Y., Guo, W., 2021. Robust surface reconstruction of plant leaves from 3D point clouds. *Plant Phenomics*. <https://doi.org/10.34133/2021/3184185>.
- Armeni, I., Sener, O., Zamir, A.R., Jiang, H., Brilakis, I., Fischer, M., Savarese, S., 2016. 3d semantic parsing of large-scale indoor spaces. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1534–1543.
- Benson, M.W., Hu, J., 2000. Asynchronous self-organizing maps. *IEEE Trans. Neural Netw.* 11 (6), 1315–1322.
- Bhattacharyya, P., Huang, C., Czarnecki, K., 2021. Sa-det3d: Self-attention based context-aware 3d object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3022–3031.
- Burguillo, J.C., 2014. Using self-organizing maps with complex network topologies and coalitions for time series prediction. *Soft Comput.* 18, 695–705.
- Cappelli, S.L., Domeignoz-Horta, L.A., Loaiza, V., Laine, A.-L., 2022. Plant biodiversity promotes sustainable agriculture directly and via belowground effects. *Trends Plant Sci.* 27 (7), 674–687.
- Chebrolu, N., Magistri, F., Läbe, T., Stachniss, C., 2021. Registration of spatio-temporal point clouds of plants for phenotyping. *PLoS One* 16 (2), e0247243.
- Conn, A., Pedmale, U.V., Chory, J., Navlakha, S., 2017. High-resolution laser scanning reveals plant architectures that reflect universal network design principles. *Cell Syst.* 5 (1), 53–62.
- Cook, P.B., McReynolds, J., 1998. Lateral inhibition in the inner retina is important for spatial tuning of ganglion cells. *Nat. Neurosci.* 1 (8), 714–719.
- DeLaRocha, J., Marchetti, C., Schiff, M., Reyes, A.D., 2008. Linking the response properties of cells in auditory cortex with network architecture: Cotuning versus lateral inhibition. *J. Neurosci.* 28 (37), 9151–9163.
- DoRêgo, R.L., Ferreira, P.H., Araújo, A.F., 2011. Reconstructing anatomical structures with growing self-reconstruction maps. In: 2011 IEEE International Conference on Systems, Man, and Cybernetics, Anchorage, AK, USA, pp. 2976–2981.
- Dovrat, O., Lang, I., Avidan, S., 2019. Learning to sample. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2760–2769.
- Dutagaci, H., Rasti, P., Galopin, G., Rousseau, D.J., 2020. Rose-X: An annotated data set for evaluation of 3d plant organ segmentation methods. *Plant Methods* 16 (1). <https://doi.org/10.1186/S13007-020-00573-W>.
- Eldar, Y., Lindenbaum, M., Porat, M., Zeevi, Y.Y., 1994. The farthest point strategy for progressive image sampling. In: Proceedings of the 12th IAPR International Conference on Pattern Recognition, Vol. III- Conference C: Signal Processing, Conference D: Parallel Computing , pp. 93–97.
- Ester, M., Kriegel, H.-P., Sander, J., Xu, X., 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining. Aug , pp. 226–231.
- Gong, L., Du, X., Zhu, K., Lin, K., Lou, Q., Yuan, Z., Huang, G., Liu, C.J., 2021. Panicle-3d: Efficient phenotyping tool for precise semantic segmentation of rice panicle point cloud. *Plant Phenomics*. <https://doi.org/10.34133/2021/9838929>.
- Guan, H., Yu, Y., Ji, Z., Li, J., Zhang, Q., 2015. Deep learning-based tree classification using mobile lidar data. *Remote Sens. Lett.* 6 (11), 864–873.
- Guerry, J., Boulch, A., Le Saux, B., Moras, J., Plyer, A., Filliat, D., 2017. SnapNet-R: Consistent 3d multi-view semantic labeling for robotics. In: 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), pp. 669–678.
- Guo, M.H., Cai, J.X., Liu, Z.N., Mu, T.J., Martin, R., Hu, S.M., 2021a. PCT: Point cloud transformer. *Comput. Vis. Media* 7 (2), 187–199.
- Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L., Bennamoun, M., 2021b. Deep learning for 3d point clouds: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.* 43 (12), 4338–4364.
- Hartline, H.K., Wagner, H.G., Ratliff, F., 1956. Inhibition in the eye of limulus. *J. Gen. Physiol.* 39 (5), 651–673.
- Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., Markham, A., 2020. RandLA-Net: Efficient semantic segmentation of large-scale point clouds. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 11105–11114.
- Huang, H., Wu, S., Cohen-Or, D., Gong, M., Zhang, H., Li, G., Chen, B.J., 2013. L1-medial skeleton of point cloud. *ACM Transactions on Graphics*, 32 (4) (2013) 1–8.
- Huang, Q., Wang, W., Neumann, U., 2018. Recurrent slice networks for 3d segmentation of point clouds. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2626–2635.
- Jiang, M., Wu, Y., Zhao, T., Zhao, Z., Lu, C., 2018. Pointsift: A sift-like network module for 3d point cloud semantic segmentation. arXiv:1807.00652. [Online]. Available: <https://arxiv.org/abs/1807.00652>.
- Joutsniemi, S.-L., Kaski, S., Larsen, T. A., 1995. Self-organizing map in recognition of topographic patterns of EEG spectra. *IEEE Transactions on Biomedical Engineering*, Nov, 42 (11), pp. 1062–1068.
- Kohonen, T., 1998. The self-organizing map. *Proc. IEEE* 21 (1–3), 1–6.
- Komarichev, A., Zhong, Z., Hua, J., 2019. A-cnn: Annular convolutional neural networks on point clouds. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7413–7422.
- Kwok, T.H., 2018. DNSS: dual-normal-space sampling for 3-D ICP registration. *IEEE Trans. Autom. Sci. Eng.* 16 (1), 241–252.
- Lafarge, F., Mallet, C., 2012. Creating large-scale city models from 3d-point clouds: A robust approach with hybrid representation. *Int. J. Comput. Vis.* 99 (1), 69–85.
- Lang, I., Manor, A., Avidan, S., 2020. Samplenet: Differentiable point cloud sampling. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7578–7588.

- Le, T., Duan, Y., 2018. Pointgrid: A deep network for 3d shape understanding. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 9204–9214.
- Lee, J. H., Yu, S. J., Park, S. C., 2001. Design of intelligent data sampling methodology based on data mining. *IEEE Transactions on Robotics and Automation*, Oct, 17 (5), 637–649.
- Levy, R.B., Reyes, A.D., 2011. Coexistence of lateral and co-tuned inhibitory configurations in cortical networks. *PLoS Comput. Biol.* 7 (10), e1002161.
- Li, Y., Bu, R., Sun, M., Wu, W., Di, X., Chen, B., 2018b. PointCNN: Convolution on X-transformed points. *Adv. Neural Inform. Process. Syst. (NIPS)* 828–838.
- Li, J., Chen, B.M., Lee, G.H., 2018a. So-Net: Self-organizing network for point cloud analysis. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 9397–9406.
- Li, B., Guo, C., 2022a. MASPC\_Transform: A plant point cloud segmentation network based on multi-head attention separation and position code. *Sensors* 22 (23). <https://doi.org/10.3390/S22239225>.
- Li, D., Shi, G., Wu, Y., Yang, Y., Zhao, M., 2021a. Multi-scale neighborhood feature extraction and aggregation for point cloud segmentation. *IEEE Trans. Circuits Syst. Video Technol.* 31 (6), 2175–2191.
- Li, D., Li, J., Xiang, S., Pan, A., 2022b. PSegNet: Simultaneous semantic and instance segmentation for point clouds of plants. *Plant Phenom.* <https://doi.org/10.34133/2022/9787643>.
- Li, Y., Ma, L., Zhong, Z., Liu, F., Chapman, M.A., Cao, D., Li, J., 2021b. Deep learning for lidar point clouds in autonomous driving: A review. *IEEE Trans. Neural Networks Learn. Syst.* 32 (8), 3412–3432.
- Li, D., Shi, G., Li, J., Chen, Y., Zhang, S., Xiang, S., Jin, S., 2022c. PlantNet: A dual-function point cloud segmentation network for multiple plant species. *ISPRS J. Photogramm. Remote Sens.* 184, 243–263.
- Li, D., Wei, Y., Zhu, R., 2023. A comparative study on point cloud down-sampling strategies for deep learning-based crop organ segmentation. *Plant Methods* 19, 124. <https://doi.org/10.1186/s13007-023-01099-7>.
- Li, Y., Wen, W., Miao, T., Wu, S., Yu, Z., Wang, X., Guo, X., Zhao, C., 2022d. Automatic organ-level point cloud segmentation of maize shoots by integrating high-throughput data acquisition and deep learning. *Comput. Electron. Agric.* 193. <https://doi.org/10.1016/J.Compag.2022.106702>.
- Lin, Q., Zong, Y., Xue, C., Wang, S., Jin, S., Zhu, Z., Wang, Y., Anzalone, A.V., Raguram, A., Doman, D.R., 2020. Prime genome editing in rice and wheat. *Nat. Biotechnol.* 38 (5), 582–585.
- López-Rubio, E., Ramos, A.D., 2014. Grid topologies for the self-organizing map. *Neural Netw.* 56, 35–48.
- Lu, D., Xie, Q., Gao, K., Xu, L., Li, J., 2022. 3DCTN: 3D convolution-transformer network for point cloud classification. *IEEE Trans. Intell. Transp. Syst.* 23 (12), 24854–24865.
- Lugmayr, A., Danelljan, M., Timofte, R., 2019. Unsupervised learning for real-world super-resolution. In: 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), pp. 3408–3416.
- Ma, Y., Guo, Y., Lei, Y., Lu, M., Zhang, J., 2018. 3DMAX-Net: A multi-scale spatial contextual network for 3D point cloud semantic segmentation. In: 24th International Conference on Pattern Recognition (ICPR), pp. 1560–1566.
- Magistri, F., Chebrolu, N., Stachniss, C., 2020. Segmentation-based 4D registration of plants point clouds for phenotyping. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2433–2439.
- Masuda, T., 2021. Leaf area estimation by semantic segmentation of point cloud of tomato plants. In: 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), pp. 1381–1389.
- Mirande, K., Godin, C., Tisserand, M., Besnard, F., Hétry-Wheeler, F., 2022. A graph-based approach for simultaneous semantic and instance segmentation of plant 3D point clouds. *Front. Plant Sci.* 13 <https://doi.org/10.3389/fpls.2022.1012669>.
- Mo, K., Zhu, S., Chang, A. X., Yi, L., Tripathi, S., Guibas, L. J., Su, H., 2019. Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 909–918.
- Nezhadarya, E., Taghavi, E., Razani, R., Liu, B., Luo, J., 2020. Adaptive hierarchical down-sampling for point cloud classification. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 12956–12964.
- Protti, D., Di Marco, S., Huang, J., Vonhoff, C., Nguyen, V., Solomon, S., 2014. Inner retinal inhibition shapes the receptive field of retinal ganglion cells in primate. *J. Physiol.-London* 592 (1), 49–65.
- Qi, C. R., Su, H., Mo, K., Guibas, L. J. 2017a. Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 652–660.
- Qi, X., Liao, R., Jia, J., Fidler, S., Urtasun, R., 2017c. 3D graph neural networks for rgbd semantic segmentation. In: Proceedings of the IEEE international conference on computer vision, pp. 5199–5208.
- Qi, H., Feng, C., Cao, Z., Zhao, F., Xiao, Y., 2020. P2b: Point-to-box network for 3d object tracking in point clouds. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 6329–6338.
- Qi, C.R., Yi, L., Su, H., Guibas, L., 2017b. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Adv. Neural Inform. Process. Syst. (NIPS)* 5099–5108.
- RAMOS, A. D., LÓPEZ-RUBIO, E., PALOMO, E. J., 2020. The forbidden region self-organizing map neural network. *IEEE Transactions on Neural Networks and Learning Systems*, Jan, 31(1), 201–211.
- Rego, R., Araújo, A.F., De Lima Neto, F.B., 2010. Growing self-reconstruction maps. *IEEE Trans. Neural Netw.* 21 (2), 211–223.
- Riera, L.G., Carroll, M.E., Zhang, Z., Shook, J.M., Ghosal, S., Gao, T., Singh, A., Bhattacharya, S., Ganapathysubramanian, B., Singh, A.K., 2021. Deep multiview image fusion for soybean yield estimation in breeding applications. *Plant Phenomics*. <https://doi.org/10.34133/2021/9846470>.
- Simonovsky, M., Komodakis, N., 2017. Dynamic edge-conditioned filters in convolutional neural networks on graphs. *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3693–3702.
- Sindelar, A. J., Schmer, M. R., Papanikolopoulos, N., 2000. Self-organizing maps for the skeletonization of sparse shapes. *IEEE Transactions on Neural Networks*, Jan, 11(1), pp. 241–248.
- Su H., Maji S., Kalogerakis E., 2015. Learned-Miller, Multi-view convolutional neural networks for 3d shape recognition. In: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 945–953.
- Tchapmi, L., Choy, C., Armeni, I., Gwak, J., Savarese, S., 2017. Segcloud: Semantic segmentation of 3d point clouds. 2017 international conference on 3D vision (3DV), pp. 537–547.
- Vitter, J.S., 1984. Faster methods for random sampling. *Commun. ACM* 27 (7), 703–718.
- Wang, X., Jin, Y., Cen, Y., Lang, C., Li, Y., 2021. Pst-net: Point cloud sampling via point-based transformer. *PST-Net: Point cloud sampling via point-based transformer*, In: 11th International Conference on Image and Graphics (ICIG), pp. 57–69.
- Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M., 2019. Dynamic graph cnn for learning on point clouds. *ACM Trans. Graph.* 38 (5) <https://doi.org/10.1145/3326362>.
- Wu, W., Qi, Z., Fuxin, L., 2019. PointConv: Deep convolutional networks on 3d point clouds. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 9621–9630.
- Xiang, Y., Wang, F., You, H., 2018. OS-SIFT: A robust SIFT-like algorithm for high-resolution optical-to-SAR image registration in suburban areas. *IEEE Trans. Geosci. Remote Sens.* 56 (6), 3078–3090.
- Xu, Y., Fan, T., Xu, M., Zeng, L., Qiao, Y. SpiderCNN: Deep learning on point sets with parameterized convolutional filters. In: Proceedings of the European conference on computer vision (ECCV), pp. 87–102.
- Xue, F., Elnashef, B., Jin, W., Filin, S., 2023. Polarimetric monocular leaf normal estimation model for plant phenotyping. *ISPRS J. Photogramm. Remote Sens.* 202, 142–157.
- Ye, X., Li, J., Huang, H., Du, L., Zhang, X. 3D recurrent neural networks with context fusion for point cloud semantic segmentation. In: Proceedings of the European conference on computer vision (ECCV), pp. 403–417.
- Yin, H., Allinson, N. M., 2001. Self-organizing mixture networks for probability density estimation. *IEEE Transactions on Neural Networks*, Mar, 12,(2), 405–411.
- Yin, H. 2002. ViSOM-a novel method for multivariate data projection and structure visualization. *IEEE Transactions on Neural Networks*, Jan, 13(1), 237–243.
- Ying, W., Dong, T., Ding, Z., Zhang, X., 2021. PointCNN-based individual tree detection using LiDAR point clouds. *Adv. Comput. Graph. CGI* 2021, 89–100.
- Zandalinas, S.I., Mittler, R., Balfagón, D., Arbona, V., Gómez-Cadenas, A., 2018. Plant adaptations to the combination of drought and high temperatures. *Physiol. Plant.* 162 (1), 2–12.
- Zheng, G., Moskal, L.M., 2012. Computational-geometry-based retrieval of effective leaf area index using terrestrial laser scanning. *IEEE Trans. Geosci. Remote Sens.* 50 (10), 3958–3969.
- Zhou, Y., Tuzel, O., 2018. Voxelnet: End-to-end learning for point cloud based 3d object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4490–4499.
- Zhu, X., Liu, J., Skidmore, A.K., Premier, J., Heurich, M., 2020. A voxel matching method for effective leaf area index estimation in temperate deciduous forests from leaf-on and leaf-off airborne LiDAR data. *Remote Sens. Environ.* 240, 111696.
- Zou, X., Cheng, M., Wang, C., Xia, Y., Li, J., 2017. Tree classification in complex forest point clouds based on deep learning. *IEEE Geosci. Remote Sens. Lett.* 14 (12), 2360–2364.