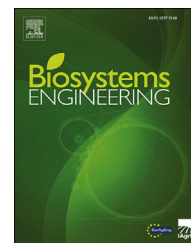


Available online at [www.sciencedirect.com](http://www.sciencedirect.com)

ScienceDirect

journal homepage: [www.elsevier.com/locate/issn/15375110](http://www.elsevier.com/locate/issn/15375110)

## Research Paper

# RoseSegNet: An attention-based deep learning architecture for organ segmentation of plants

Kaya Turgut<sup>a</sup>, Helin Dutagaci<sup>a</sup>, David Rousseau<sup>b,\*</sup><sup>a</sup> Eskisehir Osmangazi University, Eskisehir, Turkey<sup>b</sup> Université d'Angers, LARIS, UMR INRAe, IRHS, Angers, France

## ARTICLE INFO

## Article history:

Received 20 March 2022

Received in revised form  
22 June 2022

Accepted 28 June 2022

Published online 13 July 2022

## Keywords:

3D phenotyping

Plant segmentation

Point-based deep learning

Self-attention

Surface descriptor

An important component for the advancement of plant breeding, genetics, and genomics research is the rapid and accurate measurement of phenotypic traits of large plant populations. The phenotypic data that are of interest can be at multiple levels of plant organization including organ-level geometric characteristics as well as the spatial organization of the organs. 3D computer vision enabling 3D geometry acquisition and processing promises to supply fast, automated phenotypic data collection. One important component of the processing pipeline is the segmentation of the plant into its structural components, such as leaves, stems, and flowers. In this paper, a novel 3D point-based deep learning network, namely **RoseSegNet**, is proposed for segmentation of point clouds of rosebush plants to their organs. The network is equipped with two attention-based modules, one for extracting contextual features at the encoder phase, another for feature propagation at the decoder phase. The network processes regions of points in a hierarchical manner, where at each level, point features are aggregated using attention-based operators. The aggregation is performed by incorporating point relations both within and between the receptive fields, defined by the hierarchical organization of points. RoseSegNet outperforms the widely-used architecture **PointNet++** by 4% in terms of *MIoU* on the publicly available ROSE-X data set. Also, it is demonstrated that introducing local surface features together with the spatial coordinates of each 3D point at the input level boosts the segmentation performance of both networks by 9% in terms of *MIoU*.

© 2022 IAGRE. Published by Elsevier Ltd. All rights reserved.

## 1. Introduction

Component phenotypes of plants refer to measurements of individual components of plants such as leaves, branches and flowers (Choudhury et al., 2019). Leaf area, branching angle, stem length are examples to such measurements, which are traditionally obtained through manual methods. Manual

phenotyping, being highly labor-intensive and error-prone, is far from meeting the demand for rapid phenotyping of large populations of plants to analyse complex interactions between genotypes and the environment (Minervini et al., 2015). Automated phenotyping through computer vision and machine learning techniques has been intensively pursued in the last decade in order to break this phenotyping bottleneck (Mochida et al., 2018).

\* Corresponding author.

E-mail address: [david.rousseau@univ-angers.fr](mailto:david.rousseau@univ-angers.fr) (D. Rousseau).<https://doi.org/10.1016/j.biosystemseng.2022.06.016>

1537-5110/© 2022 IAGRE. Published by Elsevier Ltd. All rights reserved.

## Nomenclature

$(\lambda_1, \lambda_2, \lambda_3)$	Eigenvalues of the covariance matrix $\Sigma$	$f_{att}$	The attention-based interpolated features in the Att-Prop module
$(F_1, F_2, F_3, F_4)$	Local surface features	$f_{prop}$	The propagated features
$(x_i, y_i, z_i)$	Coordinates of point $p_i$	FN	Number of false negatives
$\alpha$	The weight function in the <i>Intra-Emb</i> operator	FP	Number of false positives
$\beta$	The transformation function to obtain the query vector in the <i>Intra-Emb</i> operator	$g$	Features of point $c_i$
$\Delta c_u$	The relative position vector between $p$ and $c_u$	$g_u$	Features of point $c_u$
$\delta_k$	The position vector of $k$ th point in the <i>Intra-Emb</i> operator	$g_{att}^{inter}$	Attention-based aggregated features in the <i>Inter-Emb</i> operator
$\gamma$	The non-linear transformation of $\alpha$ in the <i>Intra-Emb</i> operator	$g_{att}^{intra}$	Attention-based aggregated features in the <i>Intra-Emb</i> operator
$\hat{\alpha}$	The weight function in the Att-Prop module	$g_{context}$	Concatenated features of <i>Intra-Emb</i> and <i>Inter-Emb</i> operators
$\hat{\beta}$	The transformation function to obtain the query vector in the Att-Prop module	$g_{inter}$	The output features of the <i>Inter-Emb</i> operator
$\hat{\gamma}$	The non-linear transformation of $\hat{\alpha}$ in the Att-Prop module	$g_{intra}$	The output features of the <i>Intra-Emb</i> operator
$\hat{\phi}$	The non-linear transformation in the Att-Prop module	$g_{res}^{inter}$	The residual features in the <i>Inter-Emb</i> operator
$\hat{\psi}$	The transformation function to obtain value vectors in the Att-Prop module	$g_{res}^{intra}$	The residual features in the <i>Intra-Emb</i> operator
$\hat{\phi}$	The transformation function to obtain key vectors in the Att-Prop module	IoU	Intersection over Union
$\hat{W}_{ke}$	The weights of the function $\hat{\phi}$	K	Number of points randomly selected from $\mathcal{R}_i^{intra}$
$\hat{W}_{qu}$	The weights of the function $\hat{\beta}$	L	Number of centre points in $\mathcal{R}_i^{inter}$
$\hat{W}_{va}$	The weights of the function $\hat{\psi}$	M	Number of points in set $\mathcal{C}$
$\mathcal{C}$	Set of centre points	MIoU	Mean Intersection over Union
$\mathcal{F}$	Set of point features	N	Number of points in $\mathcal{P}$
$\mathcal{N}_{\S}$	Neighbourhood of point $p_i$	$N_0$	Number of points in the input point cloud
$\mathcal{P}$	The set of points representing a point cloud	P	Precision
$\mathcal{R}_i^{inter}$	The set of nearest L centre points to $c_i$	$p_i$	$i$ th 3D point in $\mathcal{P}$
$\mathcal{R}_i^{intra}$	The point set falling inside the region centred at $c_i$	$p_j$	$j$ th neighbour of point $p_i$
$\bar{p}$	Mean of the coordinates of the points in $\mathcal{N}_{\S}$	$p_k$	$k$ th 3D point in $\mathcal{R}_i^{intra}$
$\varphi$	The non-linear transformation in the <i>Intra-Emb</i> operator	R	Recall
$\psi$	The transformation function to obtain value vectors in the <i>Intra-Emb</i> operator	$r_F$	Radius of the neighbourhood for the extraction of local surface features
$\rho$	Softmax function	$r_T$	Radius of the region centred at $c_i$
$\Sigma$	Covariance matrix	s	Factor of dimension reduction for input features to attention-based modules
$\Theta$	The linear transformation of $\delta_k$ in the <i>Intra-Emb</i> operator	TP	Number of true positives
$\phi$	The transformation function to obtain key vectors in the <i>Intra-Emb</i> operator	U	The number of closest centre points in the Att-Prop module
$\vartheta$	A one-layer MLP with leaky ReLU to transform $f_k$ for $g_{res}^{intra}$	$W_{ke}$	The weights of the function $\phi$
C	Number of semantic categories of plant organs	$W_{qu}$	The weights of the function $\beta$
$c_i$	$i$ th centre point in set $\mathcal{C}$	$W_{va}$	The weights of the function $\psi$
$c_u$	$u$ th nearest centre point to $p$ in the Att-Prop module	Att-Abs	Attention-based abstraction
D	Dimension of features in $\mathcal{F}$	Att-Prop	Attention-based propagation
$f_i$	Features of point $p_i$	Inter-Emb	Inter-region Embedding
$f_k$	Features of point $p_k$	Intra-Emb	Intra-region Embedding
		CNN	Convolutional Neural Networks
		FPFH	Fast Point Feature Histograms
		FPS	Farthest point sampling
		LFPC-s	Local Features on Point Cloud - supervised
		LFPC-u	Local Features on Point Cloud - unsupervised
		NLP	Natural Language Processing
		PCA	Principal Component Analysis
		SVM	Support Vector Machines
		VCNN	Voxel-based convolutional neural networks

The visual plant data can be digitised through 2D photographic imaging (Feldmann & Tabb, 2022; Xu et al., 2019; Zhang et al., 2021) or through 3D sensing (Bao et al., 2019; Liu, Zhang, et al., 2020). In either case, component phenotyping requires segmentation of the visual plant data into individual organs, such as leaves, stems, nodes, fruits, and flowers. The accurate segmentation of plants to the organs is also critical to extract morphological and architectural traits for automated high-throughput phenotyping.

2D image-based systems are widely used for segmentation of plant parts and estimation of organ-level traits. However, 2D photographic imaging poses challenges such as self-occlusion, missing data, and the variability due to illumination conditions. The lack of 3D depth information complicates the accurate evaluation of many traits such as component size, shape, orientation, and location. The geometrical data of plants in the form of point clouds, depth images, etc, acquired through 3D sensors, supplies direct access to such measurements, provided that individual organs are accurately segmented.

The general practise for segmentation of organs from 3D plant models has been the extraction of local surface features that describe the local geometric information around each 3D point. Local features capture distinguishing properties of organ classes and model within class variability (Ziamtsov & Navlakha, 2019). Examples to such local surface features are first and second tensor features (Elnashef et al., 2019), Fast Point Feature Histograms (FPFH) (Sodhi et al., 2017; Wahabzada et al., 2015) and eigenvalues of local covariance matrix (Dey et al., 2012; Dutagaci et al., 2020). Once these point-based features are extracted, the segmentation is performed through classifying each point with traditional machine learning approaches, such as Support Vector Machines (SVM) or Random Forests.

Classification of handcrafted surface features at the local point level is effective to some degree; however, depends heavily on the design of the features and is blind to the contextual information at larger scales. Deep neural networks are capable of simultaneously extracting and aggregating features at multiple scales providing context information and weighting relevant features according to a loss function evaluated on training data.

Point cloud data can be obtained through multi-view stereo or RGB-D image acquisition. In these cases, it is possible to apply standard convolutional neural networks (CNN) to individual colour or depth images (Liu, Hu, & Li, 2020; Majeed et al., 2020; Shi et al., 2019). Point clouds can be the raw output of other acquisition devices such as 3D LiDARs. The irregular structure of point clouds poses a challenge for direct application of standard CNNs, which require regular data grids of fixed-size as input. To overcome this limitation, rendering 3D point clouds onto 2D images has been proposed (Japes et al., 2018; Jin et al., 2018; Wang et al., 2019). Segmentation is performed on the rendered 2D images through convolutional neural networks such as U-net, Mask R-CNN, or Fast-RCNN. Then, segmented pixels are associated with the original 3D point cloud data in accordance with the transformation relationship established between the images and the point cloud. Although these methods enable the application of CNNs to data derived from 3D plant models, they do not operate directly on the point clouds in 3D space. Another issue is the

computational cost of the rendering and projection phases. As another strategy, the point cloud can be converted to a volumetric form that preserves the spatial relationships in 3D and enable the application of 3D CNNs. Jin et al., 2020 and Le Louédéc and Cielniak (2021) proposed voxel-based convolutional neural networks (VCNN) for maize stem and leaf segmentation and segmentation of strawberry fruit, respectively. The disadvantage with volumetric approaches is the trade-off between computational cost and resolution due to quantization.

Recent advances on extension of deep neural networks for direct application on 3D point cloud data are key to exploring their capabilities for 3D plant analysis. Such advances on 3D point-based deep neural networks started with the introduction of PointNet (Qi, Su, et al., 2017) and its local variant PointNet++ (Qi, Yi, et al., 2017), and exploded in the last decade (Guo, Wang, et al., 2021). Despite this proliferation, the application of 3D point-based networks on plant phenotyping is limited to a few studies, mainly due to the scarcity of annotated 3D plant data sets (Chaudhury et al., 2020). In work (Turgut et al., 2022), point-based deep learning architectures were compared for organ segmentation on ROSE-X data set (Dutagaci et al., 2020) and their performances were enhanced with the incorporation of synthetic point cloud data. Schunck et al. (2021) released a multi-temporal data set and provided baseline results of point-based deep learning networks such as PointNet, PointNet++ and LatticeNet (Rosu et al., 2020). Chaudhury et al. (2021) explored the performance of PointNet++ model trained with virtual plants on real plants and obtained promising results. Boogaard et al. (2021) demonstrated the ability of PointNet++ of segmenting incomplete point clouds of cucumber plants and also showed that spectral information boosted the performance. Morel et al. (2020) proposed a network based on PointNet and PointNet++ to segment virtual trees into woody and leaf parts. After the point cloud was partitioned into overlapping sub-clouds, a variant of PointNet++ was used to extract the global information of each sub-cloud. The global feature and local features extracted by Principal Component Analysis (PCA) were concatenated and PointNet was applied recursively to predict labels. Ghahremani et al. (2021) introduced Pattern-Net to segment wheat models into organs. The point cloud was decomposed into multiple and different subsets via a random downsampling operator and a feature extraction pattern was applied across all subsets to extract the stationary patterns.

The problems associated with sampling and organizing 3D points within the structure of neural networks for effective surface characterization are still not solved. Morel et al. (2020) showed that introducing geometric local descriptors as input to 3D deep learning networks provides additional information on the distribution of points in the local neighbourhood and enhances the performance of the classifier significantly for segmentation of trees. Boogaard et al. (2021) used the spectral information as additional feature channels for improvement of the classification performance of the deep learning architecture. The addition of prior information of plant organs formulated as local geometric and spectral characteristics provides a significant benefit to the deep learning architectures to learn the underlying latent surface information.

Recently, due to the success of transformer networks in the Natural Language Processing (NLP) domain (Vaswani et al., 2017), the concept of self-attention was adapted to different domains to reveal contextual information present across longer ranges (Khan et al., 2021; Wang et al., 2021). The self-attention mechanism is well-suited to extract latent relationships of points in the 3D model analysis as it is inherently permutation-invariant for processing point cloud data (Guo, Cai, et al., 2021). Zhao et al. (2021) showed that self-attention can extract descriptive features considering the correlation between points. To the best of our knowledge, the potential of attention mechanism integrated into a point-based deep learning architecture has not been previously explored for 3D plant analysis.

In this work, a novel point-based deep network architecture, which is called RoseSegNet, is proposed to segment 3D plant models into structural parts. The network is equipped with self-attention mechanisms. It is composed of encoder and decoder parts, each designed in a hierarchical manner. The attention-based modules embedded into the layers of the encoder structure, abstract point features by relating them both within local regions (intra-region), and between representatives of local regions (inter-region). Residual connections are inserted to both the attention-based intra-region and inter-region embedding operators. In the decoder part, attention-based propagation modules are employed to hierarchically interpolate abstracted point features back to the input point cloud. The network is trained and tested on the fully annotated 3D rosebush models in the ROSE-X data set (Dutagaci et al., 2020) to segment them into flower, leaf, and stem parts. The results demonstrate that RoseSegNet outperforms the widely-used deep learning architecture, PointNet++. Moreover, introducing local surface descriptors as input to both networks results in a boost in segmentation performance.

The contributions of this work can be summarised as follows:

- A novel 3D deep learning network for efficient and accurate segmentation of 3D plant models is introduced. The network is equipped with self-attention mechanisms and residual connections to model interactions among local structures.
- It is demonstrated that, in their current state, 3D point-based deep neural networks benefit from augmenting the point coordinates with local surface features at the input stage.

## 2. Material and methods

In Section 2.1 the data set is described and details on the steps of data pre-processing are given. In Section 2.2, the local surface descriptors that are provided as input features to the networks are defined. In Section 2.3, the operations of the proposed attention-based abstraction and propagation modules are explained. Lastly, RoseSegNet is introduced as the full attention-based hierarchical point processing network in Section 2.4.

### 2.1. Data set

The publicly available **ROSE-X data set** (Dutagaci et al., 2020) is used to test the proposed deep learning architecture and to analyse the effect of incorporating local surface features. ROSE-X data set consists of 11 complete 3D point cloud models of real rosebush plants. The 3D data was acquired through **X-ray tomography**, and initially modeled in volumetric form. The models were fully annotated at point level, into three classes as flower, leaf, and stem. The stem class includes the main stem, the branches, and the petioles. Two point cloud samples from the ROSE-X data set are given in Fig. 1.

The models in the ROSE-X data set are provided in the following forms: (1) raw X-ray image stacks, (2) labeled binary volume masks, (3) labeled binary volume masks indicating the voxels only on the surfaces of the plant shoots, (4) labeled point clouds, (5) labeled point clouds composed of the points on the surfaces of the plant shoots. In this work, point clouds that represent the surfaces of the plant shoots are used. More details related to the data set are given in (Dutagaci et al., 2020). The information on file formats are also explained in the supplementary material available at the publisher site (Dutagaci et al., 2020).

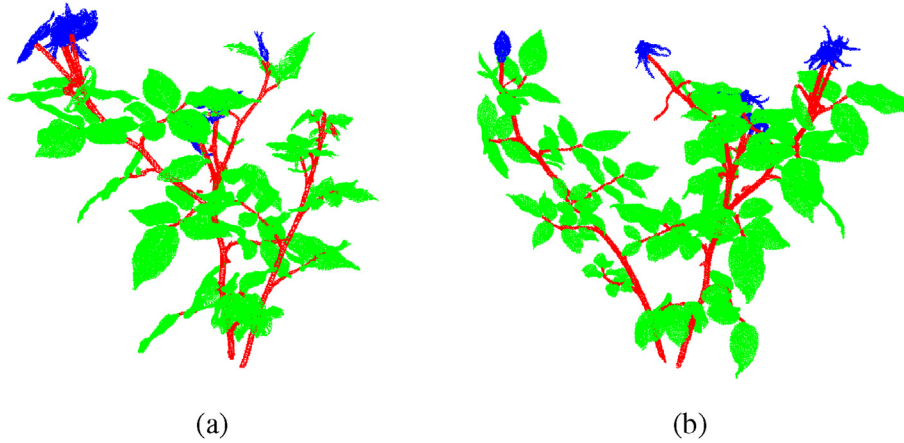
#### 2.1.1. Data pre-processing

3D point-based deep learning networks accept a fixed number of points as input. This fixed number is denoted as  $N_0$ , the number of points in the point cloud provided as input to the network. For large point clouds, subsampling the entire data to the required size is not an option since it would result in a significant loss of geometric information. The practise is to partition large point clouds into blocks of predetermined size. The set of points in a block is then processed as an independent point cloud by the network, both at the training and test phases. The off-line data preparation procedure described in the work of Li et al. (2018) is followed. Each point cloud representing a complete rosebush model is divided into non-overlapping blocks.

The first stage is the partitioning of the point cloud into fixed-size non-overlapping blocks. If the number of points in a block is less than 10% of the predetermined number of points ( $N_0$ ), the block is merged into a neighbouring block. The second stage forces the point distribution of each block to be homogeneous. Each block is divided into voxels with predefined size and the average number of points over all the voxels is calculated. For voxels that have points below the average value, the number of points is duplicated to obtain regularly distributed blocks. The last stage ensures that the number of points in each block is equal to  $N_0$ . If the number of points in a block are higher than  $N_0$ , the block is separated into new blocks that define the same region with differently sampled points. If the number of points in a block are less than  $N_0$ , random points are duplicated such that the number of points is raised to  $N_0$ .

At the training phase, block partitioning over a full rosebush model is performed with two different offset values (0 and 5 cm), thus two sets of blocks are extracted from each model. This operation both provides augmented data and pushes the discontinuities at the blocks in the first set towards





**Fig. 1** – Two samples from ROSE-X data set. ROSE-X data set consists of 11 complete 3D point cloud models of real rosebush plants (Dutagaci et al., 2020). The models were fully annotated at point level, into three classes as flower, leaf, and stem. The stem class includes the main stem, the branches, and the petioles.

the centres of the blocks in the second set. At the inference phase, the same offset values are applied to obtain two sets of blocks from a rosebush model. Each block is processed independently by the network, which produces the class probabilities of the points in the block as the output. For a single point in the rosebush model, class probabilities of the points are calculated by the network separately for each block. The final class of a point is determined according to the highest value among the probability scores of the corresponding points contained in two overlapping blocks.

The choice of the block size depends on the resolution of the point cloud. A large block size results in a significant loss of geometric detail due to subsampling, while small blocks lack contextual information among neighbouring plant organs. In the experiments, the block size is set as 10 cm, which provides a good compromise between point resolution and context range. Examples to extracted blocks are given in Fig. 2. The grid size, and the number of final points in each block are set as 0.2 cm, and 8192, respectively.

## 2.2. Local surface features

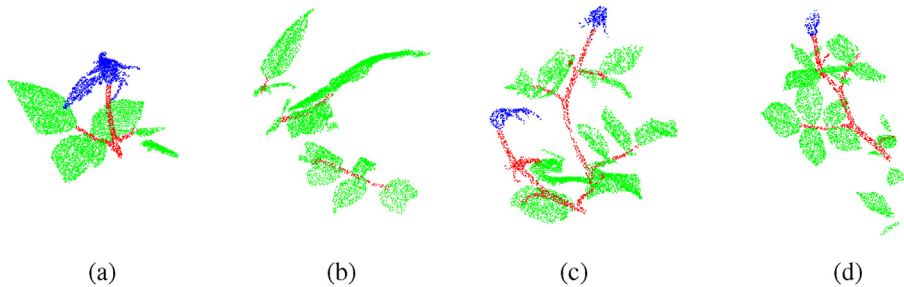
3D point-based networks are expected to take as input the raw 3D coordinates of points, occasionally together with surface

normals, and to probe and encode class-specific properties out of this raw data. However, at their current stage, the 3D point-based networks are still progressing in their manner of organizing the geometric data at the local level. To enhance the informative power of the raw point coordinates, hand-crafted point-based local surface features can be incorporated as input attributes. Then, the networks can exploit this additional information for encoding and aggregation of the interactions of local structures at various scales.

The eigenvalues of the local point covariance matrix are used to define the surface features. Let the input point cloud be denoted as the set  $\mathcal{P} = \{p_1, p_2, \dots, p_N\}$  where the each point  $p_i = (x_i, y_i, z_i)$  is represented in 3D coordinates. The neighbourhood of a point  $p_i$  can be defined as  $\mathcal{N}_i = \{p_j : \|p_i - p_j\| < r_F\}$ , where  $r_F$  is the radius of the neighbourhood. The covariance matrix of the points in the neighbourhood is calculated as:

$$\Sigma = \frac{1}{\mathcal{N}_i - 1} \sum_{p_j \in \mathcal{N}_i} (p_j - \bar{p}_i)(p_j - \bar{p}_i)^T \quad (1)$$

where  $\bar{p}_i$  denotes the mean of the coordinate vectors of the points in the neighbourhood.



**Fig. 2** – Examples of blocks from ROSE-X data set. Each point cloud representing a complete rosebush model is divided into non-overlapping blocks. The set of points in a block is then processed as an independent point cloud by the network, both at the training and test phases. In the experiments, the block size is set as 10 cm, which provides a good compromise between point resolution and context range.

The eigenvalues  $\lambda_1 < \lambda_2 < \lambda_3$  of the covariance matrix  $\Sigma$  represent the amount of the variation of the points in the neighbourhood along three principal axes. They carry information about the local shape around the point  $p_i$ . For example, when  $\lambda_1$  and  $\lambda_2$  are close to zero and  $\lambda_3$  is relatively large, that is indicative of an elongated, line-like structure. On a locally planar region, both  $\lambda_2$  and  $\lambda_3$  are expected to be larger than  $\lambda_1$ . The relations between eigenvalues of the covariance matrix around each point can be used to distinguish line-like, plane-like, and spherical local structures, hence be used as local descriptors for classification of flower, leaf, and stem points. The following local features are used as given in the work of Dutagaci et al. (2020):

$$F_1 = \frac{\lambda_1}{\sqrt{\lambda_2 \lambda_3}}, F_2 = \frac{\lambda_2}{\lambda_3}, F_3 = \frac{\lambda_1}{\sqrt{\lambda_1 \lambda_2 \lambda_3}}, F_4 = \frac{\lambda_1}{\lambda_2} \quad (2)$$

Local regions of different sizes instead of a fixed-size neighbourhood are processed to provide information for multiple scales. The local features,  $\{F_1, F_2, F_3, F_4\}$ , are extracted from local neighbourhoods of six different radii around each point, amounting to 24 local features. The radii are selected as  $r_F$ , as 2, 3, 4, 5, 6, and 7 mm.

### 2.3. The attention-based modules of the network

Before providing the full architecture for RoseSegNet, two core modules are described: Attention-based abstraction module (Att-Abs) at the encoder and Attention-based propagation module (Att-Prop) at the decoder. The Attention-based abstraction module (Att-Abs) is responsible for extracting contextual information for each local region, by considering the relations of points within the region and the relations of the representative point of the region with representative points of neighbouring local regions. Residual connections are present in Att-Abs module in order to capture dominant features as well as contextual features. The Attention-based propagation module (Att-Prop) allows the contextual information to impact the rate of feature propagation at the decoder.

#### 2.3.1. Attention-based abstraction module (Att-Abs)

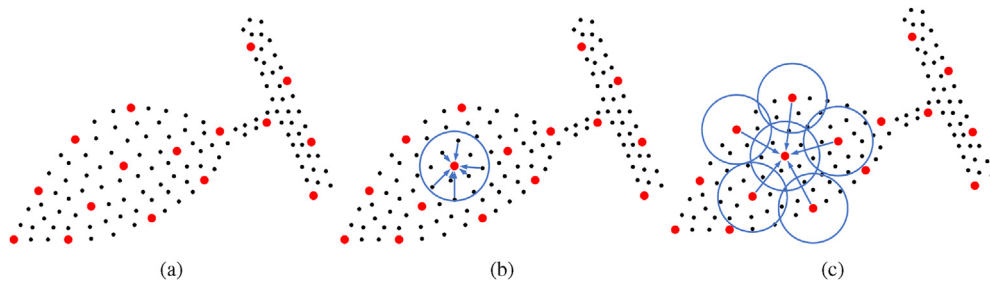
The input to the Att-Abs module is composed of the spatial coordinates  $\mathcal{P} = \{p_1, p_2, \dots, p_N\}$  with  $p_i \in \mathbb{R}^3$  and features

$\mathcal{F} = \{f_1, f_2, \dots, f_N\}$  with  $f_i \in \mathbb{R}^D$  of the unordered point set, where  $N$  denotes the cardinality, and  $D$  denotes the dimension of input point features. The module extracts contextual features from the point set by relating points within regions around representative (centre) points and between these representative points. To this end, the Att-Abs module engages two operators: 1-) Intra-region Embedding (*Intra-Emb*) Operator and 2-) Inter-region Embedding (*Inter-Emb*) Operator.

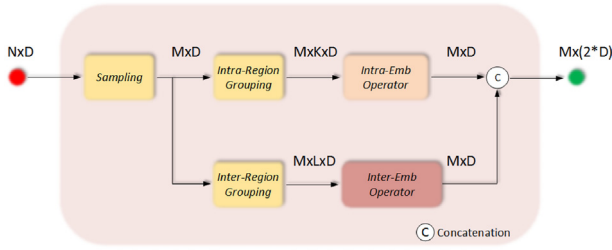
The input point set is sampled with iterative farthest point sampling (FPS) algorithm to determine representative points  $\mathcal{C} = \{c_1, c_2, \dots, c_M\}$ , where  $M < N$ . In the illustration given in Fig. 3, the blue points represent the input points, which correspond to the aggregated points from the previous layer. The representative points at the current layer onto which features will be aggregated are depicted in red colour. Around each representative point, a region whose radius is fixed at the particular layer is defined. The *Intra-Emb* operator relates each centre point to the points within its region through an attention-based approach (Fig. 3b). The *Inter-Emb* operator embeds longer range interactions by relating each centre point to other centre points of neighbouring regions (Fig. 3c). The Att-Abs module combines within-region and between-region contextual features extracted via the two operators.

The diagram of the Att-Abs module is given in Fig. 4. After centre points are determined by the FPS algorithm, the grouping stage of the intra-region points and neighbourhoods of centre points for *Intra-Emb* and *Inter-Emb* operators are carried out in parallel. The grouped points together with their features are fed to *Intra-Emb* and *Inter-Emb* operators. Finally, context-aware features returned by the two operators are concatenated. In the following subsections, details of the *Intra-Emb* and *Inter-Emb* operators are given.

***Intra-Emb* Operator:** The grouping stage prior to *Intra-Emb* operation corresponds to determining the points in the region centred at the representative point  $c_i$ . This region of radius  $r_T$  is called the receptive field centred at  $c_i$  at the current layer.  $K$  points are randomly selected among the points falling inside this region to form the set  $\mathcal{R}_i^{\text{intra}} = \{p_{i,1}, p_{i,2}, \dots, p_{i,K}\}$  (Fig. 3b). For simplicity, the index  $i$  is dropped, and the selected points within the region are denoted as  $p_k$ , and the corresponding input features as  $f_k$ , with  $k = 1, \dots, K$ . The input feature vector of the centre point is denoted as  $g$ .



**Fig. 3 – Point sampling and feature aggregation:** a-) Red points are sampled using farthest point sampling (FPS) algorithm from the aggregated points (blue points) of the previous layer. Each red point defines a region of fixed radius at the particular layer. b-) Intra-region interactions.  $K$  points are randomly selected. c-) Inter-region interactions.  $L$  nearest centre points are selected. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)



**Fig. 4 – The attention-based abstraction (Att-Abs) module. The module is responsible for extracting contextual information for each local region.**

With the attention-based approaches, query, key, and value vectors are obtained via transformations of the features of entities whose relationships are to be revealed (Vaswani et al., 2017). The goal of the *Intra-Emb* operation is to encode the relationship between the centre point and the points in the region defined by the centre point. A point feature aggregation approach similar to the point transformer described in the work of Zhao et al. (2021) is followed. The query vector is set to be a transformation of the features of the centre point, as ‘query’:  $\beta(g, W_{qu})$ . The key and value vectors are transformations of the features of  $K$  points within the region, and are calculated as ‘key’:  $\phi(f_k, W_{ke})$  and ‘value’:  $\psi(f_k, W_{va})$ . The functions  $\beta, \phi, \psi: \mathbb{R}^D \rightarrow \mathbb{R}^{D/s}$  map the input features linearly to lower dimensions through the transformations  $W_{qu}, W_{ke}$ , and  $W_{va}$ , respectively, which are to be learned through training. For all attention-based modules in the network,  $s$  is set to 2. For the sake of simplicity, the transformation parameters from the arguments are dropped, and the transformed features are denoted as ‘query’:  $\beta(g)$ , ‘key’:  $\phi(f_k)$ , and ‘value’:  $\psi(f_k)$ .

The block diagram of the *Intra-Emb* operation is given in Fig. 5. The features aggregated on the centre point through attention mechanism are calculated as:

$$g_{att}^{intra} = \phi \left( \sum_{k=1}^K \alpha(g, f_k, \delta_k) \odot (\psi(f_k) + \delta_k) \right) \quad (3)$$

where  $\odot$  is the Hamadard product,  $\alpha$  is the weight function,  $\delta$  is the position encoded vector,  $\phi$  is a non-linear transformation function and  $K$  is the number of points sampled from the local region. The transformation function  $\phi$  is used to increase the feature dimension of the aggregated contextual feature back to the original feature size. The weight function  $\alpha$  measures and transforms the dissimilarity between the transformed features (‘query’ vector) of the centre point and transformed features (‘key’ vectors) of the  $K$  points within the region. The function also incorporates a transformation of Euclidean distance vectors between the centre point and the  $K$  points, as positional encoding. The aggregation is then performed by weighing the transformed features (corresponding to ‘value’ vectors) of  $K$  points. The weight function is defined as:

$$\alpha(g, f_k, \delta_k) = \rho(\gamma(\beta(g) - \phi(f_k) + \delta_k)). \quad (4)$$

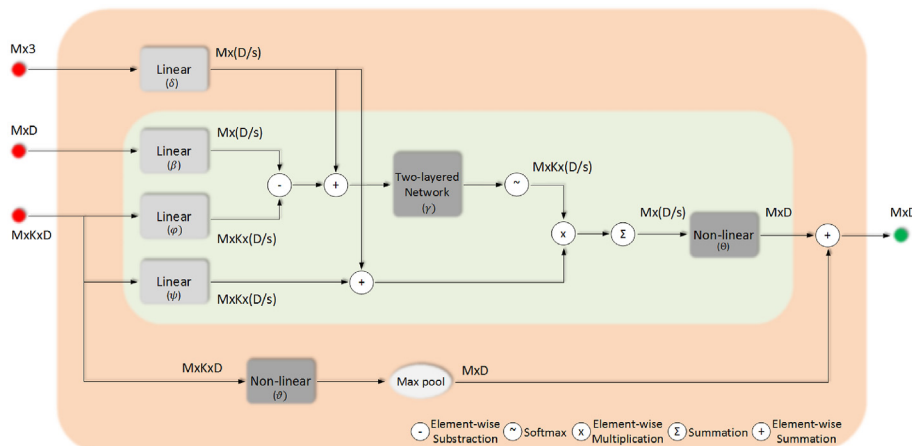
The relation between ‘query’  $\beta(g)$  and ‘key’  $\phi(f_k)$  vectors is represented by the subtraction operation. The position encoded vector  $\delta_k$  is a linear function of the relative position of the centre point to the  $k$ th point in the region:

$$\delta_k = \Theta(c - p_k). \quad (5)$$

The parameters of the linear transformation function  $\Theta$  is learned through training. The position encoded vector is added both to the difference of query  $\beta(g)$  and key  $\phi(f_k)$  vectors in the weight function and the value vectors  $\psi(f_k)$  in the aggregation function.

The function  $\gamma$  is a two-layered network, where the first layer is nonlinear and the second layer is linear. It is used to learn the embedding that will effectively represent the relative dissimilarity measures between points. The softmax function  $\rho$  is used to normalise the weights across  $K$  ‘value’ vectors.

Inspired by the effectiveness of residual networks (He et al., 2016), the max-pooled version ( $g_{res}^{intra}$ ) of the transformed features  $\{\psi(f_k)\}$  of  $K$  points are added to the attention-based aggregated features ( $g_{att}^{intra}$ ). The objective here is to let the



**Fig. 5 – Intra-Emb operator. The operator relates each centre point to the points within its region through an attention-based approach.**

dominant feature among the transformed input features of the  $K$  points contribute to the aggregated output:

$$g_{\text{intra}} = g_{\text{att}}^{\text{intra}} + g_{\text{res}}^{\text{intra}} \quad (6)$$

$$g_{\text{res}}^{\text{intra}} = \max_{k=1, \dots, K} \vartheta(f_k) \quad (7)$$

where  $\vartheta$  is a one-layer MLP with leaky ReLU.

**Inter-Emb Operator:** While the *Intra-Emb* operator aggregates features of  $K$  points within the receptive field defined around each centre point (Fig. 3b), the *Inter-Emb* operator explores the interactions between receptive fields through relating features of each centre point to its neighbouring centre points (Fig. 3c). The two operators run in parallel.

The *Inter-Emb* operator is designed in the same manner as the *Intra-Emb*. In *Inter-Emb*, the attention mechanism operates on the input features of a target centre point ('query') and other centre points ('keys' and 'values') in its vicinity. The *Inter-Emb* operator aggregates point features in longer ranges as compared to the *Intra-Emb* operator.

In *Inter-Emb*, for each centre point  $c_i$ , the nearest neighbour search algorithm is applied to find  $L$  closest centre points to form the group  $R_{i,L}^{\text{inter}} = \{c_{i,1}, c_{i,2}, \dots, c_{i,L}\}$ . The 'query' vector is a transformation of the features of the centre point. The key and value vectors are calculated as the transformed features of the neighbouring  $L$  centre points.

Through attention-based feature aggregation, the inter-region features  $g_{\text{att}}^{\text{inter}}$  are obtained. The output of the *Inter-Emb* operator is

$$g_{\text{inter}} = g_{\text{att}}^{\text{inter}} + g_{\text{res}}^{\text{inter}} \quad (8)$$

where  $g_{\text{res}}^{\text{inter}}$  represents residual max-pooled features. The details for the computation of  $g_{\text{att}}^{\text{inter}}$  and  $g_{\text{res}}^{\text{inter}}$  can be found in the Supplementary Material.

As stated before, the features extracted by the intra-region and inter-region operators are concatenated to form the output contextual feature of the Attention-based Abstraction (Att-Abs) Module:

$$g_{\text{context}} = g_{\text{intra}} \oplus g_{\text{inter}} \quad (9)$$

where  $\oplus$  denotes concatenation operation.

### 2.3.2. Attention-based propagation module (Att-Prop)

The Attention-based propagation (Att-Prop) module operates at the decoder layers, which propagate the aggregated

features back to the original points. The Att-Prop module allows the contextual information to impact the rate of feature propagation. The block diagram of the Att-Prop module is given in Fig. 6.

At the encoder, each successive Att-Abs module yields more descriptive and long-ranged features, but the features are aggregated at fewer points. The general practise for distributing the features aggregated at various layers back to the original point cloud is to perform distance-based interpolation and to introduce skip links from the abstraction layers to the propagation layers (Qi, Yi, et al., 2017). In this work, the use of self-attention is proposed to learn the interpolation weights according to the relation of point coordinates and features.

Recall that the Att-Abs module at the encoder accepts a point set ( $\mathcal{P}$ ) of size  $N$  with 3D coordinates and point-features and returns a subset ( $\mathcal{C}$ ) of size  $M$ , with output features aggregated through attention mechanisms. The representative (centre) points in  $\mathcal{C}$  are determined by Furthest Point Sampling. The propagation at the decoder stage aims to distribute the aggregated features of the set  $\mathcal{C}$  to the points in set  $\mathcal{P}$ . The Att-Prop module relates each point in  $\mathcal{P}$  to its neighbours in the set  $\mathcal{C}$  through an attention mechanism. Given a point coordinate  $p \in \mathcal{P}$ , its  $U$  nearest neighbours among the centre points are determined as  $\{c_1, c_2, \dots, c_U\} \subset \mathcal{C}$ , with corresponding features as  $g_u \in \mathbb{R}^{D_1}$ ,  $u = 1, \dots, U$ . The query vector is a linear transformation of the relative position vector between point  $p$  and centre point  $c_u$ , defined as 'query':  $\hat{\beta}(\Delta c_u, \hat{W}_{qu})$ , where  $\Delta c_u = p - c_u$ . The key and value vectors are computed through linear transformations of the features  $g_u$  as 'key':  $\hat{\phi}(g_u, \hat{W}_{ke})$  and 'value':  $\hat{\psi}(g_u, \hat{W}_{va})$ .

The feature for point  $p$  is interpolated from the  $U$  points through the following attention-based weighting scheme:

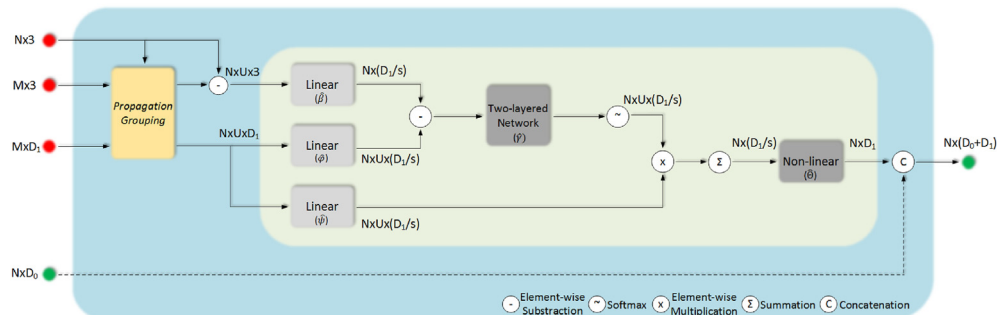
$$f_{\text{att}} = \hat{\phi} \left( \sum_{u=1}^U \hat{\alpha}(\Delta c_u, g_u) \odot \hat{\psi}(g_u) \right) \quad (10)$$

where  $\odot$  is the Hamadard product,  $\hat{\alpha}$  is the weight function, and  $\hat{\phi}$  is a non-linear function that transforms the features into their original dimensionality.

The weight vector is determined as:

$$\hat{\alpha}(\Delta c_u, g_u) = \rho(\hat{\gamma}(\hat{\beta}(\Delta c_u) - \hat{\phi}(g_u))) \quad (11)$$

where  $\hat{\gamma}$  is a two-layered network and  $\rho$  is the softmax function. Weight vector  $\hat{\alpha}(\Delta c_u, g_u)$  is learned according to the



**Fig. 6 – The attention-based propagation (Att-Prop) module.** The module allows the contextual information to impact the rate of feature propagation at decoder layers.



relation function which is a subtraction of linear transformations of relative point coordinates and features.

Given that the features of point  $p$  were determined as  $f \in \mathbb{R}^{D_0}$  at the encoder stage, the final propagated point features  $f_{prop} \in \mathbb{R}^{D_0+D_1}$  at the decoder stage is set as the concatenation of  $f$  and the interpolated features  $f_{att} \in \mathbb{R}^{D_1}$ :

$$f_{prop} = f_{att} \oplus f \quad (12)$$

The features  $f$  are provided to the decoder through skip links from the encoder.

#### 2.4. The network architecture

The proposed deep learning architecture, RoseSegNet, for organ segmentation of plants is given in Fig. 7. The network is built on an encoder-decoder structure. The encoder follows a local-to-global strategy, employing the Att-Abs module at each layer to extract semantic affinities between points at the given scale of spatial interaction range. By this strategy, the receptive fields are gradually expanded, and the spatial interaction ranges between aggregated points become longer. Also, the number of regions processed by successive abstraction modules is decreased to mimic convolution neural networks. The decoder is responsible to propagate aggregated features at successive layers of the encoder back to the original points in an hierarchical manner through Att-Prop modules. This allows that each point in the original set is enriched by the features carrying context information from various scales. The semantic labels of the points are then inferred through these informative point features.

Unit embedding operators are used before each Att-Abs module at the encoder stage, and after each Att-Prop module at the decoder stage. These operators consist of weight-shared MLPs which uplift the input features to higher dimensions to enrich their representation power at the encoder stage, and decrease the dimensionality at the decoder stage.

In the encoder part, four layers, each equipped with Att-Abs modules are used to aggregate features with a local-to-global strategy. The input point set with  $N_0 = 8192$  points is downsampled to 1024, 256, 64, and 16 points through these four layers, and the receptive fields are expanded to 5 mm, 10 mm, 20 mm, and 40 mm, respectively. The number of points randomly sampled from intra-regions is set to  $K = 32$

for the Intra-Emb operator. The number of nearest centre points is set to  $L = 8$  for the Inter-Emb operator. The unit embedding operators at the four successive layers map feature dimensions to 64, 128, 256, and 512, respectively. The output dimension of each Att-Abs module is doubled since the embedded features on intra-region and inter-region are concatenated.

At the decoder stage, the contextual and long-range features of the down-sampled points are propagated to the original points. The decoder has four layers consisting of unit embedding operators and Att-Prop modules. Features of centre points are propagated to the point sets of the previous layers by using  $U = 3$  nearest neighbours in the Att-Prop module. The output of the Att-Prop is the concatenation of the propagated features and the features of Att-Abs provided by skip links. The unit embedding operators, inserted after the Att-Prop modules, map feature dimensions to 512, 256, 128, and 64 at the four successive layers.

After the decoder stage, two fully connected layers with feature dimensions, 64 and  $C$  are engaged to extract scores of  $C$  categories for each point in the input point cloud.

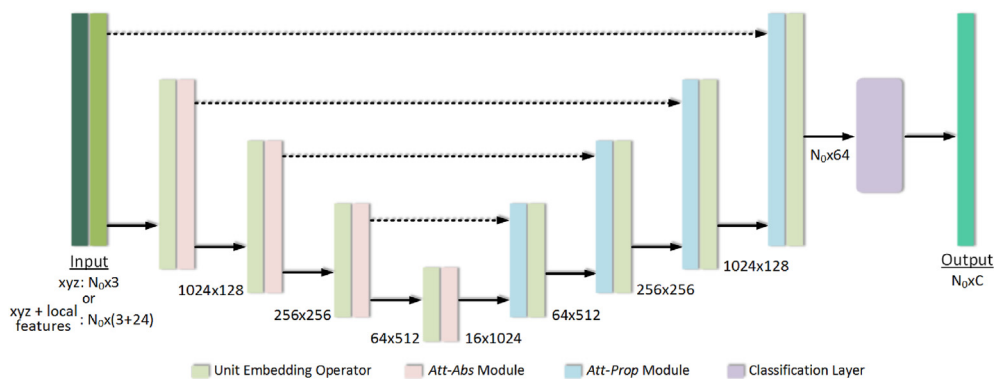
All non-linear layers in the architecture include leaky ReLU activation function and batch normalization. Drop-out with keep ratio 0.5 is used on the last fully connected layer.

#### 2.5. Implementation details

The network is implemented in Tensorflow, and trained on an NVIDIA Quadro P5000. The size of the input point cloud is set to  $N_0 = 8192$  points. The training is performed using Adam optimiser, to minimise a weighted cross-entropy loss function. The batch size is 16, and training is completed after 200 epochs. The initial learning rate is set as 0.005. The learning rate is decayed by 0.7 for every 30 epochs. As data augmentation, rotated versions of the point clouds are used. Specifically, random rotations around the upright axis is applied to the point clouds.

#### 2.6. Experimental setup

To evaluate the segmentation performance of the proposed network, 5-fold cross-validation experiments were performed. For each fold, a single rosebush model from ROSE-X



**Fig. 7 – The architecture of RoseSegNet. The input is a point cloud of  $N_0$  points. The point features at the input level can either be the 3D coordinates only ( $xyz$ ) or the 3D coordinates and 24 local features together ( $xyz + \text{local features}$ ).**

data set was reserved for optimizing the weights of the network, and the remaining 10 models are used as test data. The data processing steps were followed to partition the point cloud into blocks as described in Section 2.1.1. In Table 1, the total number of training and test blocks that were extracted from training and test rosebush models are given for each fold. The number of blocks varied depending on the size of the corresponding rosebush.

### 2.7. Evaluation metrics

In this study, three metrics were used to compare the success of plant organ segmentation: 1-) Precision (P), 2-) Recall (R), and 3-) Intersection over Union (IoU). These metrics are defined as:

$$P_i = \frac{TP_i}{TP_i + FN_i} \quad (13)$$

$$R_i = \frac{TP_i}{TP_i + FP_i} \quad (14)$$

$$IoU_i = \frac{TP_i}{TP_i + FN_i + FP_i} \quad (15)$$

where the  $TP_i$ ,  $FP_i$  and  $FN_i$  represent the number of true positives, the number of false positives and the number of false negatives of each class  $i \in \{\text{flower}, \text{leaf}, \text{stem}\}$ . The categorization of a point as a true positive or otherwise is demonstrated in Fig. 8.

## 3. Results

The segmentation performance of RoseSegNet is evaluated on the publicly available ROSE-X data set (Dutagaci et al., 2020). The effect of employing the attention-based modules are analysed through ablation studies. Also, the positive contribution of feeding the network with hand-crafted local features is demonstrated.

### 3.1. Semantic segmentation of ROSE-X

The segmentation results of the proposed network, RoseSegNet, are given in Table 2, in comparison with three other segmentation methods. Each evaluation value in the Table 2 represents the average and standard deviation over the 5-fold experiments. The state-of-the-art point-based deep learning network PointNet++ was selected as a baseline for

		Prediction by the network	
		Classified as class $i$	Classified to a category other than $i$
Ground Truth	Belongs to class $i$	True Positive ( $TP_i$ )	False Negative ( $FN_i$ )
	Does not belong to class $i$	False Positive ( $FP_i$ )	True Negative ( $TN_i$ )

**Fig. 8 – Confusion matrix** ( $i \in \{\text{flower}, \text{leaf}, \text{stem}\}$ ). A point is a true positive if both its actual category and predicted category is  $i$ . It is a false negative if the point's actual category is  $i$  and the network classified it wrongly to another class. It is a false positive if the point does not belong to class  $i$ , but the network classified it as class  $i$ . If neither the actual class nor the predicted class of the point is  $i$ , then it is a true negative.

comparison. The default architecture of PointNet++ was used, but it was trained with the same hyperparameters (radii of local regions,  $K$ , number of epochs, batch size and learning rate) as those of RoseSegNet. For both PointNet++ and RoseSegNet, two different networks are trained. The first network accepts only the 3D coordinates as input point features. The second network is designed to accept the local surface features (see Section 2.2) together with the 3D coordinates as input point features.

The segmentation results on the same data set of two other methods described in (Dutagaci et al., 2020) are also given. These methods are the LFPC-u (Local Features on Point Cloud - unsupervised) and The LFPC-s (Local Features on Point Cloud - supervised). The supervised method uses **Support Vector Machines (SVM)** as the machine learning model. The segmentation results in Table 2 are given as reported in (Dutagaci et al., 2020). These two methods were used to classify points only to two categories as 'stem' and 'leaf'. The points annotated as 'flower' were ignored.

As observed from Table 2, in terms of IoU, RoseSegNet fed with local surface features outperforms the other methods over all categories. RoseSegNet fed with local surface features outperforms LFPC-s over all performance metrics. The 0.5% drop in terms of 'leaf' precision as compared to LFPC-u is compensated by 3%, 17%, and 5% increase in terms of 'leaf' recall, 'stem' precision, and 'stem' recall, respectively.

RoseSegNet outperforms PointNet++ by 4% in terms of MIOU, with or without the use of local features. When both networks are augmented with local features, PointNet++ performs with a 1.6% higher 'flower' precision, and a 0.1% higher 'leaf' recall. These are compensated by RoseSegNet with 14% higher 'flower' recall and 1% higher 'leaf' precision.

PointNet++, relying only on the spatial coordinates of the input points, returns lower performance figures for leaf and stem categories as compared to LFPC-u and LFPC-s. Notice that these two methods are representatives of the traditional unsupervised and supervised techniques for organ segmentation of plants. RoseSegNet, without the local

**Table 1 – Number of training and test blocks used in the experiments.**

	# Blocks for training	# Blocks for test
Fold 1	99	677
Fold 2	53	723
Fold 3	103	673
Fold 4	56	720
Fold 5	89	687

**Table 2 – The segmentation performance of RoseSegNet on ROSE-X data set in comparison with PointNet++, LFPC-u, and LFPC-s.**

		LFPC-u		LFPC-s		PointNet++		RoseSegNet	
						xyz	xyz + local features	xyz	xyz + local features
Precision	Flower	—	—	—	—	88.92 ± 5.31	<b>92.86 ± 2.03</b>	86.55 ± 8.18	91.20 ± 3.79
	Leaf	<b>98.23 ± 0.33</b>	97.19 ± 0.48	95.74 ± 0.84	96.76 ± 1.01	95.74 ± 0.84	96.76 ± 1.01	96.53 ± 0.25	97.78 ± 1.11
	Stem	75.01 ± 9.76	83.67 ± 4.88	77.25 ± 3.85	90.48 ± 1.93	77.25 ± 3.85	90.48 ± 1.93	79.90 ± 5.20	<b>91.96 ± 0.46</b>
Recall	Flower	—	—	61.11 ± 10.10	64.67 ± 10.39	61.11 ± 10.10	64.67 ± 10.39	73.52 ± 7.36	<b>79.05 ± 12.03</b>
	Leaf	95.74 ± 1.74	97.79 ± 0.46	97.11 ± 0.77	<b>98.77 ± 0.29</b>	97.11 ± 0.77	<b>98.77 ± 0.29</b>	97.04 ± 0.32	98.67 ± 0.34
	Stem	88.03 ± 1.82	80.50 ± 1.29	82.90 ± 2.47	92.78 ± 3.97	82.90 ± 2.47	92.78 ± 3.97	83.01 ± 3.98	<b>92.87 ± 1.37</b>
IoU	Flower	—	—	56.17 ± 6.71	61.51 ± 9.30	56.17 ± 6.71	61.51 ± 9.30	65.01 ± 2.41	<b>72.91 ± 7.59</b>
	Leaf	94.10 ± 1.54	95.10 ± 0.46	93.08 ± 0.68	95.60 ± 0.77	93.08 ± 0.68	95.60 ± 0.77	93.77 ± 0.29	<b>96.50 ± 0.78</b>
	Stem	67.96 ± 8.18	69.57 ± 3.87	66.63 ± 3.54	84.52 ± 3.68	66.63 ± 3.54	84.52 ± 3.68	68.41 ± 2.81	<b>85.90 ± 1.11</b>
MIoU		—	—	71.96 ± 2.16	80.55 ± 3.80	71.96 ± 2.16	80.55 ± 3.80	75.73 ± 1.02	<b>85.10 ± 2.85</b>

features, gives lower ‘leaf’ and ‘stem’ IoU measures compared to LFPC-s. Supplying local features as input to the networks boosts the performance significantly for both PointNet++ and RoseSegNet. The performance increase in terms of MIoU is 9% for both networks, when the input data is enriched by local features. These observations demonstrate the importance of augmenting input 3D point coordinates with corresponding local surface features while training the point-based networks.

In Fig. 9, visual segmentation results provided by PointNet++ and RoseSegNet on six sample rosebush blocks are given. The first column represents the ground truth. The second and third columns give the results obtained with PointNet++ and RoseSegNet, respectively, with the use of spatial coordinates only. The fourth and fifth columns depict the segmentation of the blocks by PointNet++ and RoseSegNet, respectively, when both networks are fed with local surface features.

In Figs. 9a, c, 9e and 9f, it can be observed that petioles are classified as leaf points by PointNet++ without the use of local features. The addition of local features alleviates this confusion for both PointNet++ and RoseSegNet. However, petiole-leaf distinction is best modeled by RoseSegNet augmented with local features. This success can be attributed to the attention-based mechanisms that extract contextual information at leaf-stem boundaries.

Another source of error is the misclassification of some flower regions as either leaf or stem points (Figs. 9a, c, 9d, 9e, and 9f). The misclassification is most pronounced with PointNet++ trained without the local surface features. With the exception of the block in Fig. 9f, RoseSegNet with local features captured the variations among the flower points most effectively. The gain with IoU for the flower class with RoseSegNet over PointNet++ is significant (11%) as can be observed from Table 2. For the case of the flower in Fig. 9f, the addition of local features to RoseSegNet lead to confusion of petals with leaves. Despite this example, the IoU value for the flower class is significantly higher, in average, for RoseSegNet operating with local features (Table 2).

An interesting result arises in Fig. 9a. The stipules in the rosebush models were originally labeled as ‘stem’ in the ground-truth annotation of ROSE-X data set. The attention-based RoseSegNet tends to classify those stipule points at

the extremities as flowers, which is coherent with the contextual relation that elongated and short flower parts tend to occur at the extremities, while stem class is long ranging and usually followed by leaf or flower points. The assignment of stipules to the flower class also contributes to the lower ‘flower’ precision yielded by RoseSegNet.

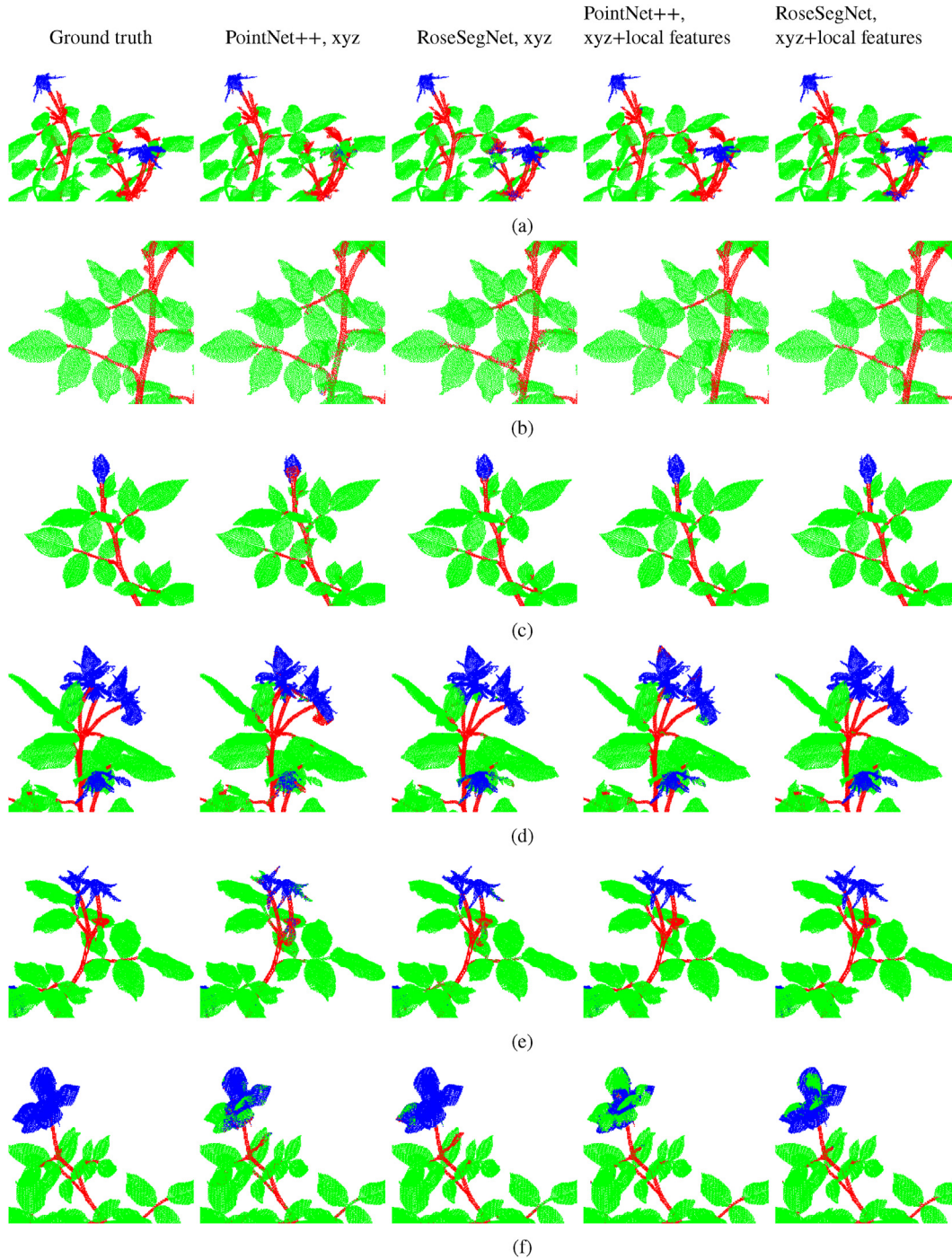
### 3.2. Ablation study

Ablation experiments on RoseSegNet were conducted to observe the influence of different settings. The version of RoseSegNet where local features were introduced to the network as input in addition to the spatial coordinates was studied. The results of these experiments are given in Table 3. The first row of the table indicates the classification performance of the default RoseSegNet. Each model in the remaining rows was trained with the same hyper-parameters and the same settings, except the one specified in the first column of the corresponding row.

Three experiments were performed to analyse the effect of exclusion of attention-based modules. When Att-Abs was removed, the feature aggregation operation was replaced by PointNet++’s default abstraction strategy (max-pooling of embedded features). When Att-Prop was excluded, the feature propagation was implemented according to the distance-based scheme of PointNet++. Using either Att-Abs or Att-Prop modules has individually increased MIoU by about 5% as compared to the case where neither was included. There was about 1% drop in IoU for the stem class with the inclusion of attention-based modules, however, the increase in IoU for the flower class was significant (Table 3). The attention-based modules helped increase the performance for the difficult and rare ‘flower’ class.

The Att-Abs module was assessed in terms of the contributions of the Intra-Emb and Inter-Emb operators. While the performance of the network without the features encoded by the Inter-Emb operator decreased slightly as compared to the default RoseSegNet, the performance drop without the Intra-Emb operator was significant (Table 3). The Intra-Emb operator is more essential since it aggregates features within receptive fields in the spirit of a convolution operation. The Inter-Emb operator provides longer range context information, which especially helps modelling the ‘flower’ class more effectively.





**Fig. 9 – Segmentation results provided by PointNet++ and RoseSegNet on six sample rosebush blocks. The first column represents the ground truth. The second and third columns give the results obtained with PointNet++ and RoseSegNet, respectively, with the use of spatial coordinates only. The fourth and fifth columns depict the segmentation of the blocks by PointNet++ and RoseSegNet, respectively, when both networks are fed with local surface features.**

The number of neighbouring centre points ( $L$ ) used in the *Inter-Emb* operator was also varied. Setting  $L = 8$  yielded the highest *MIoU* (Table 3).

Another experiment was conducted to assess the contribution of the residual connections. *Intra-Emb* and *Inter-Emb* operators were kept, but their residual connections were removed. The *IoU* for flower and leaf classes dropped without

the residual connections, while there was a slight increase in *IoU* for the stem class. However, in overall, it can be observed that the *MIoU* benefits from the inclusion of residual connections.

Finally, the number of closest points  $U$  selected for feature propagation at the *Att-Prop* module was varied. While increasing this number up to a certain point ( $U = 3$ ) had a



**Table 3 – Ablation study on RoseSegNet. The first row gives IoU values of the default RoseSegNet that includes Att-Abs and Att-Prop modules. Each Att-Abs module consists of Intra-Emb and Inter-Emb operators with residual connections. Number of neighbouring centre points is selected as  $L = 8$  in Inter-Emb operator. In Att-Prop module,  $U$  is set to 3. Remaining rows of the Table give IoU values with the setting changed as specified in the first column.**

Model	$\text{IoU}_{\text{flower}}$	$\text{IoU}_{\text{leaf}}$	$\text{IoU}_{\text{stem}}$	$\text{MIoU}$
RoseSegNet	<b>81.99</b>	<b>97.26</b>	85.26	<b>88.17</b>
Architecture				
w/o Att-Abs	78.48	96.71	86.31	87.17
w/o Att-Prop	79.43	96.90	86.58	87.64
w/o Att-Abs, Att-Prop	64.59	95.81	86.86	82.42
Att-Abs Module				
w/o Inter-Emb	79.41	97.01	86.40	87.61
w/o Intra-Emb	71.55	96.02	83.12	83.57
w/o Residual	77.19	96.87	85.57	86.54
$L = 4$	76.72	96.90	<b>87.14</b>	86.92
$L = 12$	81.23	97.22	85.95	88.13
$L = 16$	80.61	96.86	86.39	87.96
Att-Prop Module				
$U = 1$	77.97	96.72	85.77	86.82
$U = 5$	79.42	96.99	86.46	87.62
$U = 8$	78.45	96.49	84.39	86.44

positive effect on the performance, further increase lead to a drop in performance in addition to the increased computational cost.

#### 4. Discussion

The demand for increased productivity and quality of produce pushes for programs aiming to breed agricultural plants with high genetic potential. Automated solutions for trait measurements through 3D computer vision will enable experimentation with a high number of plants for breeding, genetics, and genomics research. 3D acquisition and modelling of plant geometry provides complete and accurate measurements of the plant shape. Organ segmentation of the 3D plant models is indispensable for extraction of organ-level traits in high-throughput phenotyping as well as for monitoring emergence and growth of individual organs in horticulture.

Examples to such organ-level traits are number of leaves, organization of leaves, leaf areas, leaf inclination angles, proportions among the organs, and shape of organs. One important application area where measurements of such traits over a large number of plants during their development are in demand is genetic and mechanical control of organ growth. Understanding the genetic regulation mechanisms that determine organ identity, growth, size, and shape is possible through phenotypic shape measurements at organ level (Johnson & Lenhard, 2011). The role the external mechanical processes and constraints play in organ morphogenesis along with genes is also an important research question (Trinh et al., 2021). There is considerable variation in the shapes of plant organs such as fruits, leaves, and stems

even within plants with identical genetic composition. The interplay of genetics with mechanical constraints leading to such variations is yet to be understood for regulating the shapes of harvestable plant organs (Lazzaro et al., 2018).

In the case of ornamental plants, the research question of correlating the subjective aesthetic quality with quantitative geometric and architectural attributes is of considerable interest (Boumaza et al., 2009; Demotes-Mainard et al., 2013; Garbez et al., 2018). Garbez et al. (2018) conducted a thorough study to investigate the relation between the architecture of rosebush plants and the visual perception of consumers. With a Fastrack® 3D digitiser (Polhemus, Colchester, VT, USA), they manually measured a large number of traits such as apparent plant axes, their topological relations (succession or branching), the number of branching, number of leaves, flowers, fruits, and carrier axes, size of the leaves, height of the flowers. Demotes-Mainard et al. (2013) also explored the elements of visual quality of rosebush plants by gathering quantitative attributes. They collected data of leaf dimensions via destructive measurements and manually observed the number of visible leaves, internodes and terminal leaflet lengths.

As opposed to such laborious manual data gathering over a small number of plants, digital processing of 3D plant models can provide automated trait measurements over large populations for expanding scientific knowledge on the development of the shape of the produce. Also, automatic plant monitoring and shape characterization via 3D vision enables accurate plant management, especially for plants where individual organ shapes as well as their spatial organizations are important agronomic traits. The architectural and morphological attributes of importance, such as organization of axes, leaf sizes, flower height, etc., can be estimated only through the decomposition of the acquired plant model into its individual organs. The accuracy of automatic plant segmentation methods operating on 3D models directly influences accurate trait estimation.

This work provides an organ segmentation method that brings an improvement on the accuracy of previous segmentation techniques measured on a publicly available data set. The application of technical innovations in 3D point cloud segmentation to plant models in the framework of deep learning is in its infancy. Deep learning techniques promise fast characterization of the vast amount of structural and geometrical variations among and within plant species via learning with training data, and without incorporation of much expert knowledge. The recent advances in 3D point-based deep learning methods in the field of computer vision, however, mainly target robotic applications other than those related to plant sciences and agriculture. The proposed RoseSegNet is a progress in the direction of designing deep neural network architectures suitable for 3D plant model analysis, specifically 3D plant organ segmentation. The inclusion of attention-based mechanisms modelling interactions of local structures at multiple scales, and within and among local regions, is the main contribution of this work in relation with previous applications of 3D point-based networks. Augmenting spatial features of local structures with hand-crafted surface features and letting the network

process, relate, and aggregate these features towards organ identity inference is another contribution.

Despite the considerable research in plant genetics, investment on provision of publicly available annotated 3D plant data sets for research purposes is alarmingly low. The need for large amount of annotated data for training deep learning techniques has the potential of pushing for development of fast acquisition and labelling protocols, that will eventually lead to common use of 3D robot vision both in plant sciences and in agriculture. In the context of this work, however, the amount of annotated data is limited to eleven rosebush models, and only one model is used for training. One important research question in the framework of deep learning is the assessment of the impact of the amount of training data on various architectures. The limits of the performance improvement with respect to the amount of data, as well as the potential of substantial data availability for closing the gap between different deep learning approaches are yet to be explored. On the other hand, the ability of a network in achieving high performance with limited annotated data is of considerable importance due to the time-consuming process of manually annotating 3D models of target crops.

## 5. Conclusion

This paper demonstrates a progress in the improvement of organ segmentation accuracy through advanced deep learning techniques to contribute to automated and accurate estimation and monitoring of organ-level and architectural phenotypical traits that are crucial in plant sciences and horticultural processes. A novel point-based deep learning network, which is named as RoseSegNet, is proposed to segment 3D point clouds of rosebush plants into their structural parts. The network is designed to process the input point cloud in a hierarchical manner and to extract contextual features based on the relations between points. The contextual features are encoded and propagated by attention-based modules. Through ablation studies, the contribution of each of these modules was analysed. The attention-based contextual features improved the segmentation accuracy, especially for the flower class. Augmenting input point coordinates with local surface descriptors boosted the performance of RoseSegNet and PointNet++. RoseSegNet improved the segmentation performance as compared to the traditional classification methods based on local surface features. RoseSegNet also achieved significant improvement over the state-of-the-art 3D point-based deep learning framework PointNet++. These results suggest that deep learning methods devised to model 3D characteristics of plants are capable of surpassing traditional techniques that solely depend on hand-crafted features. The capacity of deep neural networks to simultaneously extract and evaluate relevant attributes from raw data and simple surface features without intervention of experts is of special importance for plant characterization. Deep neural networks trained on one plant species also have the potential of applicability to a large variety of other species through domain adaptation with the use of few training data.

## 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.

## Acknowledgements

The authors acknowledge the support of The Scientific and Technological Research Council of Turkey (TUBITAK), Project No: 121E088.

The authors also acknowledge the support of 2214/A International Doctoral Research Fellowship Program granted to Kaya Turgut by TUBITAK and PHENOTIC platform node of the french plant phenotyping infrastructure PHENOME.

## Appendix A. Supplementary data

Supplementary data to this article can be found online at <https://doi.org/10.1016/j.biosystemseng.2022.06.016>.

## REFERENCES

- Bao, Y., Tang, L., Srinivasan, S., & Schnable, P. S. (2019). Field-based architectural traits characterisation of maize plant using time-of-flight 3d imaging. *Biosystems Engineering*, 178, 86–101. <https://doi.org/10.1016/j.biosystemseng.2018.11.005>
- Boogaard, F., van Henten, E., & Kootstra, G. (2021). Boosting plant-part segmentation of cucumber plants by enriching incomplete 3D point clouds with spectral data. *Biosystems Engineering*, 211, 167–182. <https://doi.org/10.1016/j.biosystemseng.2021.09.004>
- Boumaza, R., Demotes-Mainard, S., Huche-Thelier, L., & Guerin, V. (2009). Visual characterization of the esthetic quality of the rosebush. *Journal of Sensory Studies*, 24, 774–796. <https://doi.org/10.1111/j.1745-459X.2009.00238.x>. <https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1745-459X.2009.00238.x>. arXiv.
- Chaudhury, A., Boudon, F., & Godin, C. (2020). 3D plant phenotyping: All you need is labelled point cloud data. In *CVPPP-ECCV 2020 - Workshop on Computer Vision Problems in Plant Phenotyping* (pp. 1–17). Glasgow, United Kingdom. [https://doi.org/10.1007/978-3-030-65414-6\\_18](https://doi.org/10.1007/978-3-030-65414-6_18)
- Chaudhury, A., Hanappe, P., Boudon, Azais, R., Godin, C., & Colliaux, D. (2021). Transferring PointNet++ segmentation from virtual to real plants. In *CVPPP-ICCV 2021 - 7th workshop on Computer Vision in Plant Phenotyping and Agriculture*.
- Choudhury, S. D., Samal, A., & Awada, T. (2019). Leveraging image analysis for high-throughput plant phenotyping. *Frontiers of Plant Science*, 10. <https://doi.org/10.3389/fpls.2019.00508>.
- Demotes-Mainard, S., Bertheloot, J., Boumaza, R., Huch-Th  lier, L., Gu  ritaine, G., Gu  rin, V., & Andrieu, B. (2013). Rose bush leaf and internode expansion dynamics: Analysis and development of a model capturing interplant variability. *Frontiers of Plant Science*, 4. <https://doi.org/10.3389/fpls.2013.00418>
- Dey, D., Mummert, L., & Sukthankar, R. (2012). Classification of plant structures from uncalibrated image sequences. In 2012

- IEEE Workshop on the Applications of Computer Vision (WACV) (pp. 329–336). <https://doi.org/10.1109/WACV.2012.6163017>
- Dutagaci, H., Rasti, P., Galopin, G., & Rousseau, D. (2020). ROSE-X: An annotated data set for evaluation of 3D plant organ segmentation methods. *Plant Methods*, 16. <https://doi.org/10.1186/s13007-020-00573-w>
- Elnashef, B., Filin, S., & Lati, R. N. (2019). Tensor-based classification and segmentation of three-dimensional point clouds for organ-level plant phenotyping and growth analysis. *Computers and Electronics in Agriculture*, 156, 51–61. <https://doi.org/10.1016/j.compag.2018.10.036>
- Feldmann, M. J., & Tabb, A. (2022). Cost-effective, high-throughput phenotyping system for 3D reconstruction of fruit form. *The Plant Phenome Journal*, 5. <https://doi.org/10.1002/ppj2.20029>
- Garbez, M., Symoneaux, R., Belin, E., Caraglio, y., Chéné, y., Dones, N., Durand, J.-B., Hunault, G., Relion, D., Sigogne, M., Rousseau, D., & Galopin, G. (2018). Ornamental plants architectural characteristics in relation to visual sensory attributes: A new approach on the rose bush for objective evaluation of the visual quality. *European Journal of Horticultural Science*, 83, 187–201. <https://doi.org/10.17660/eJHS.2018/83.3.8>
- Ghahremani, M., Williams, K., Corke, F., Tiddeman, B. P., Liu, Y., & Doonan, J. H. (2021). Deep segmentation of point clouds of wheat. *Frontiers of Plant Science*, 12. <https://doi.org/10.3389/fpls.2021.608732>
- Guo, M.-H., Cai, J.-X., Liu, Z.-N., Mu, T.-J., Martin, R. R., & Hu, S.-M. (2021a). PCT: Point cloud transformer (Vol. 7, pp. 187–199). *Computational Visual Media*. <https://doi.org/10.1007/s41095-021-0029-5>
- Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L., & Bennamoun, M. (2021b). Deep learning for 3D point clouds: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43, 4338–4364. <https://doi.org/10.1109/TPAMI.2020.3005434>
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27–30, 2016 (pp. 770–778). <https://doi.org/10.1109/CVPR.2016.90>
- Japes, B., Mack, J., Rist, F., Herzog, K., Töpfer, R., & Steinhage, V. (2018). Multi-view semantic labeling of 3D point clouds for automated plant phenotyping, Article 03994. *ArXiv*, abs/1805.
- Jin, S., Su, Y., Gao, S., Wu, F., Hu, T., Liu, J., Li, W., Wang, D., Chen, S., Jiang, Y., Pang, S., & Guo, Q. (2018). Deep learning: Individual maize segmentation from terrestrial lidar data using faster R-CNN and regional growth algorithms. *Frontiers of Plant Science*, 9, 866. <https://doi.org/10.3389/fpls.2018.00866>
- Jin, S., Su, Y., Gao, S., Wu, F., Ma, Q., Xu, K., Ma, Q., Hu, T., Liu, J., Pang, S., Guan, H., Zhang, J., & Guo, Q. (2020). Separating the structural components of maize for field phenotyping using terrestrial lidar data and deep convolutional neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, 58, 2644–2658. <https://doi.org/10.1109/TGRS.2019.2953092>
- Johnson, K., & Lenhard, M. (2011). Genetic control of plant organ growth. *New Phytologist*, 191, 319–333. <https://nph.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1469-8137.2011.03737.x>. arXiv.
- Khan, S., Naseer, M., Hayat, M., Zamir, S. W., Khan, F. S., & Shah, M. (2021). Transformers in vision: A survey. <https://doi.org/10.1145/3505244>
- Lazzaro, M. D., Wu, S., Snouffer, A., Wang, Y., & van der Knaap, E. (2018). Plant organ shapes are regulated by protein interactions and associations with microtubules. *Frontiers of Plant Science*, 9. <https://doi.org/10.3389/fpls.2018.01766>
- Le Louédec, J., & Cielniak, G. (2021). 3D shape sensing and deep learning-based segmentation of strawberries. *Computers and Electronics in Agriculture*, 190, Article 106374. <https://doi.org/10.1016/j.compag.2021.106374>
- Li, Y., Bu, R., Sun, M., Wu, W., Di, X., & Chen, B. (2018). PointCNN: Convolution on [X]-transformed points. In *Advances in Neural Information Processing Systems* (Vol. 31) Curran Associates, Inc.
- Liu, X., Hu, C., & Li, P. (2020a). Automatic segmentation of overlapped poplar seedling leaves combining mask R-CNN and DBSCAN. *Computers and Electronics in Agriculture*, 178, Article 105753. <https://doi.org/10.1016/j.compag.2020.105753>
- Liu, Z., Zhang, Q., Wang, P., Li, Z., & Wang, H. (2020b). Automated classification of stems and leaves of potted plants based on point cloud data. *Biosystems Engineering*, 200, 215–230. <https://doi.org/10.1016/j.biosystemseng.2020.10.006>
- Majeed, Y., Zhang, J., Zhang, X., Fu, L., Karkee, M., Zhang, Q., & Whiting, M. D. (2020). Deep learning based segmentation for automated training of apple trees on trellis wires. *Computers and Electronics in Agriculture*, 170, Article 105277. <https://doi.org/10.1016/j.compag.2020.105277>
- Minervini, M., Scharr, H., & Tsafaris, S. A. (2015). Image analysis: The new bottleneck in plant phenotyping [applications corner]. *IEEE Signal Processing Magazine*, 32, 126–131. <https://doi.org/10.1109/MSP.2015.2405111>
- Mochida, K., Koda, S., Inoue, K., Hirayama, T., Tanaka, S., Nishii, R., & Melgani, F. (2018). Computer vision-based phenotyping for improvement of plant productivity: A machine learning perspective. *GigaScience*, 8. arXiv <https://academic.oup.com/gigascience/article-pdf/8/1/gy153/27359533/gy153.pdf>. Gy153.
- Morel, J., Bac, A., & Kanai, T. (2020). Segmentation of unbalanced and in-homogeneous point clouds and its application to 3D scanned trees. *The Visual Computer*, 36, 2419–2431 (Special issue for CGI 2020).
- Qi, C. R., Su, H., Mo, K., & Guibas, L. J. (2017a). PointNet: Deep learning on point sets for 3D classification and segmentation. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 77–85). <https://doi.org/10.1109/CVPR.2017.16>
- Qi, C. R., Yi, L., Su, H., & Guibas, L. J. (2017b). PointNet++: Deep hierarchical feature learning on point sets in a metric space. In *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems*.
- Rosu, R. A., Schütt, P., Quenzel, J., & Behnke, S. (2020). Latticenet: Fast point cloud segmentation using permutohedral lattices. In *Proc. of Robotics: Science and Systems (RSS)*.
- Schunck, D., Magistri, F., Rosu, R. A., Cornelißen, A., Chebrolu, N., Paulus, S., Léon, J., Behnke, S., Stachniss, C., Kuhlmann, H., & Klingbeil, L. (2021). Pheno4D: A spatio-temporal dataset of maize and tomato plant point clouds for phenotyping and advanced plant analysis. *PLoS One*, 16, Article e0256340. <https://doi.org/10.1371/journal.pone.0256340>
- Shi, W., van de Zedde, R., Jiang, H., & Kootstra, G. (2019). Plant-part segmentation using deep learning and multi-view vision. *Biosystems Engineering*. <https://doi.org/10.1016/j.biosystemseng.2019.08.014>
- Sodhi, P., Vijayarangan, S., & Wettergreen, D. (2017). In-field segmentation and identification of plant structures using 3D imaging. In 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 5180–5187). <https://doi.org/10.1109/IROS.2017.8206407>
- Trinh, D.-C., Alonso-Serra, J., Asaoka, M., Colin, L., Cortes, M., Malivert, A., Takatani, S., Zhao, F., Traas, J., Trehin, C., & Hamant, O. (2021). How mechanical forces shape plant organs. *Current Biology*, 31, R143–R159. <https://doi.org/10.1016/j.cub.2020.12.001>
- Turgut, K., Dutagaci, H., Galopin, G., & Rousseau, D. (2022). Segmentation of structural parts of rosebush plants with 3D point-based deep learning methods. *Plant Methods*, 18. <https://doi.org/10.1186/s13007-022-00857-3>
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., & Polosukhin, I. (2017). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach,

- R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* (Vol. 30).
- Wahabzada, M., Paulus, S., Kersting, K., & Mahlein, A.-K. (2015). Automated interpretation of 3D laserscanned point clouds for plant organ segmentation. *BMC Bioinformatics*, 16. <https://doi.org/10.1186/s12859-015-0665-2>
- Wang, J., Chen, X., Cao, L., An, F., Chen, B., Xue, L., & Yun, T. (2019). Individual rubber tree segmentation based on ground-based lidar data and faster R-CNN of deep learning. *Forests*, 10. <https://doi.org/10.3390/f10090793>
- Wang, X., Jin, Y., Cen, Y., Wang, T., & Li, Y. (2021). *Attention models for point clouds in deep learning: A survey*. *ArXiv, abs/2102.10788*.
- Xu, L., He, K., & Lv, J. (2019). Bayberry image segmentation based on manifold ranking salient object detection method. *Biosystems Engineering*, 178, 264–274. <https://doi.org/10.1016/j.biosystemseng.2018.12.001>
- Zhang, L., Xia, C., Xiao, D., Weckler, P., Lan, Y., & Lee, J. M. (2021). A coarse-to-fine leaf detection approach based on leaf skeleton identification and joint segmentation. *Biosystems Engineering*, 206, 94–108. <https://doi.org/10.1016/j.biosystemseng.2021.03.017>
- Zhao, H., Jiang, L., Jia, J., Torr, P. H., & Koltun, V. (2021). *Point transformer*. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 16259–16268).
- Ziamtsov, I., & Navlakha, S. (2019). Machine learning approaches to improve three basic plant phenotyping tasks using three-dimensional point clouds. *Plant Physiology*, 181, 1425–1440. <https://doi.org/10.1104/pp.19.00524>