Computers and Electronics in Agriculture 206 (2023) 107619

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Original papers

Autoencoder-based 3D representation learning for industrial seedling abnormality detection

Hendrik A.C. de Villiers [a,*], Gerwoud Otten [a], Aneesh Chauhan [a], Lydia Meesters [a]

[a] Food and Biobased Research, Wageningen University and Research, Bornse Weilanden 9, 6708 WG Wageningen, The Netherlands

ARTICLE INFO

Keywords:
Deep learning
Autoencoder
Seedling
3D
PointNet
Usable transplant test

ABSTRACT

Industrial seedling quality assessment, such as attempting to find abnormal seedlings, is a challenging task where assessment methods must contend with the natural variability of seedlings, as well as the subjective nature of expert judgements. Furthermore, obtaining expert judgements is expensive and time-consuming, so machine learning approaches which rely on fewer judgements would be useful in practice. We investigate autoencoders, operating on 3D point clouds obtained from 6732 seedlings to address this challenge, exploiting such systems' ability to work with partially labelled data. Point clouds from tomato seedlings are recorded using a 3D data capture platform, MARVIN™, and the quality of each seedling is determined by expert judgement. An existing system is used to establish baseline performance scores using a rule-based expert system and machine learning with handcrafted features. Autoencoders are trained on the point clouds to learn representations for subsequent use in classification. We examine scenarios where large amounts of partially labelled data are available, and compare with the case where fully labelled data is available. To improve performance, we compare the architectural subcomponents based on PointNet and PointNet++, as well as the effect of different training strategies. We find, with 13.6% of training data labelled, our model has correct classification rates of 97.7% and 82.7% for normals and abnormals respectively. With further improvements and fully labelled data, we find that correct classification rates of 97.6% and 96.1% can be reached. The results demonstrate that semi-supervised learning supported by partially labelled data has the potential to greatly reduce the cost of data curation, with minimal impact on overall accuracy.

1. Introduction

In this paper, we address the challenge of seedling quality assessment for the purposes of implementing automated usable transplant (UT) tests for tomato seedlings. The UT test is a germination test where the quality of a seed lot is determined by the percentage of usable transplants. This test emerged from a demand from the plant propagation industry because the standard germination tests failed to predict the number of saleable plants accurately (Van Der Burg et al., 1994). The Usable Transplants are referred to as normal plants. Abnormal plants are plants that exhibit non-uniformity in development, in which the leaf surface and the morphology of the cotyledons and the first leaves differ from the normal plants. For plant propagators, in order to facilitate planning of plant production, it is critical to estimate the percentage of seedlings capable of growing normally.

Seedlings exhibit complex natural variation and the difference between a normal and an abnormal seedling may be subtle (see Fig. 1). A key challenge is that seed lots consist of relatively few abnormal plants where abnormality is perceived as a deviation from normal. Therefore, abnormal seedlings are not a closed set, and abnormal samples cannot be defined beforehand. This makes it difficult to frame seedling inspection as a purely supervised learning problem, since it is not possible to acquire large amounts of representative examples of abnormal plants. Furthermore, experts do not necessarily agree during the labelling of plants, meaning that machine learning models must be able to function despite the inherent subjectivity involved in the labelling process. In addition, expert labelling is a tedious and time consuming task and, as in many professions, finding skilled workers for seedling assessment is becoming increasingly difficult. An automated high-throughput seedling assessment system that provides repeatable objective seedling classification, in particular abnormality detection, could offer a solution.

In recent years, several plant phenotyping applications, including seedling inspections, have benefited from high-throughput imaging systems using computer vision and machine learning. Systems have been developed for seedling length measurements, automatic classification of their development and seedling sorting. The complexity and diversity of

* Corresponding author at: Food and Biobased Research, Wageningen University and Research, Bornse Weilanden 9, 6708 WG Wageningen, The Netherlands.
E-mail address: hendrik.devilliers@wur.nl (H.A.C. de Villiers).

https://doi.org/10.1016/j.compag.2023.107619
Received 14 February 2022; Received in revised form 28 November 2022; Accepted 1 January 2023
Available online 8 February 2023

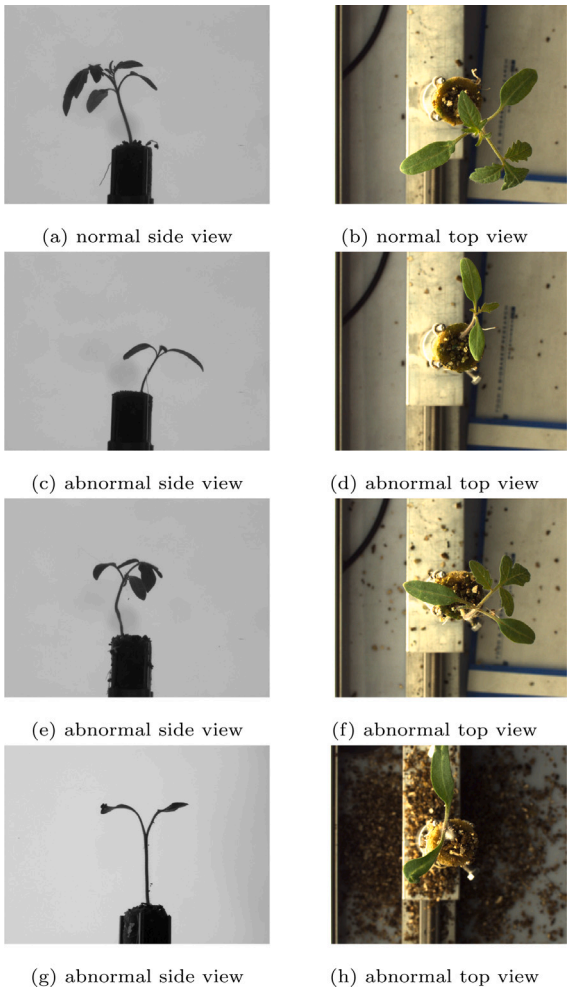

Fig. 1. The side view (left column) and the corresponding top view of each plant (right column) show examples of: a normal seedling (a) and (b); an abnormal seedling that was found to be too small (c) and (d); an abnormal seedling with a curve in the stem (e) and (f) which is only visible in the side-view; and an abnormal seedling with stopped or disturbed growth, the first leaves have not yet formed (g) and (h). The difference between a normal and abnormal seedling is subtle and often difficult to judge from the top view only.

seedling features remain a challenge, but most of the existing systems are demonstrated to be comparable to or better than human performance. We consider some examples from existing literature. Dobos et al. (2019) introduced a high-throughput plant phenotyping system based on deep learning to determine the hypocotyl length of seedlings. Image processing in combination with deep neural networks were used by Samiei et al. (2020) to classify seedlings into three growth stages. In Ashraf et al. (2011) a fully automated industrial grafting robot system was developed that deploys tomato seedling selection based on stem diameter differences. The MARVIN™ system (Golbach et al., 2016), used for seedling inspection and sorting, creates 3D point clouds of seedlings and extract multiple handcrafted features, such as total plant volume and number of leaves. Different classification techniques are then deployed based on these features to discriminate between normal and abnormal seedlings. MARVIN like approaches have multiple weaknesses, ranging from fidelity of point cloud construction, to segmentation of plant parts, to extraction of features and the eventual classification. In particular, the classification methods, which are optimized on the engineered features, are not capable of successfully generalizing to previously unknown abnormalities. With the current methods not all types of abnormalities can be detected, this is why a deep learning based classification approach is investigated.

Deep neural networks are known for their ability to learn rich sets of features with which input data is represented for further processing. In particular, autoencoders (Goodfellow et al., 2016) have become one of the more popular techniques for anomaly detection in an unsupervised manner (Chalapathy and Chawla, 2019; Ruff et al., 2019; Wang et al., 2019). An example of autoencoders employed for anomaly detection in the agrifood domain is provided by Strothmann et al. (2019) who operate on RGB images of grapes to locate abnormal instances. Autoencoders have also been investigated as unsupervised feature extractors where the learned encoded features are used as input to traditional classifiers for disease identification (Trang et al., 2020; Ani Brown Mary et al., 2020). Methods operating on 3D data have a potential advantage over such 2D methods in that the spatial structure of objects of interest are explicit in the input data. However, because of the unordered nature of 3D point clouds, autoencoders operating on them face unique challenges. The development of PointNet (Charles et al., 2017) enabled the direct processing of such point clouds by using symmetric functions like "max" to operate over points in an order-independent way. By extension, autoencoders adopting PointNet-like encoder stages could benefit from the same ability to directly process unordered point clouds, as demonstrated in Achlioptas et al. (2018).

A number of further autoencoder approaches have been proposed for point clouds. Certain systems consider improving the properties of the autoencoder's encoding directly. For example, the system of Zamorski et al. (2020) utilizes generative adversarial training to enforce a prior on the code. Other systems attempt to improve reconstruction fidelity by changing the output representation. This includes the approaches described by Chen and Zhang (2019) and Park et al. (2019) which learn to predict the reconstruction as implicit functions. Another example includes (Yang et al., 2018), which uses a decoder stage that represents the output shape as a 2D grid deformed in 3D space.

To the best of the authors' knowledge, this article presents the first introduction of PointNet-based autoencoders (such as in Achlioptas et al., 2018) to the agrifood domain. Because the objects being modelled from this domain, seedlings, can be substantially more complex and variable than the artificial 3D objects (such as cars and tables) often employed by previous studies, the reconstruction ability of such autoencoders is tested to new extremes. Furthermore, classification between normal and abnormal seedlings is typically a more subtle problem than categories from previous datasets, meaning that higher requirements are placed on the informativeness of the autoencoder code compared with earlier studies.

Previous studies typically employed a fully supervised training approach. We explore a semi-supervised approach, where limited data has been labelled by the experts and much of the data is unlabelled. This is of high interest in making the system more deployable in the industry, as expert labelling is expensive, time consuming and error-prone. Deep autoencoder architectures are explored to learn rich feature representations from the raw 3D data in an unsupervised manner. The encoder portion of the autoencoder is inspired by the PointNet (Charles et al., 2017) and PointNet++ architectures (Qi et al., 2017), while a novel decoding network is proposed which can be seen as Gaussian noise transformed using the encoder output (i.e. the bottleneck features) to a final output shaped like the input point cloud. Once the autoencoder is (pre)trained, three training scenarios were investigated: an "indirect" approach where classification layers are connected to the bottleneck layer and trained without modifying the encoder; a "refinement" approach where the classification layers and the pre-trained encoder networks are updated together; and a "direct" approach where the autoencoder is not pre-trained, and the encoder and the classifier are trained together. Indirect training was explored to take advantage of relatively plentiful unlabelled data in addition to limited labelled data.

In the subsequent sections, an outline is provided of the materials and methods involved in our experiments. Subsequently, the results obtained are given and discussed in Section 3. Here we examine the quality of the learned representations, the benefit of using additional

Table 1
Number of plants per class in each dataset (NRM = Normal, ABN = Abnormal, NG = Non-germinated). Except for Dataset 1, all non-germinated plant pots got excluded from the other datasets because their 3D point clouds contained no points.

Dataset	Total	NRM	ABN	NG
1	936	699	197	40
2	Too small plants/not used			
3	169	142	27	0
4	1038	21	1017	0
5	1004	0	1004	0
6	960	904	56	0
7	982	961	21	0
8	686	686	0	0
9	957	874	83	0

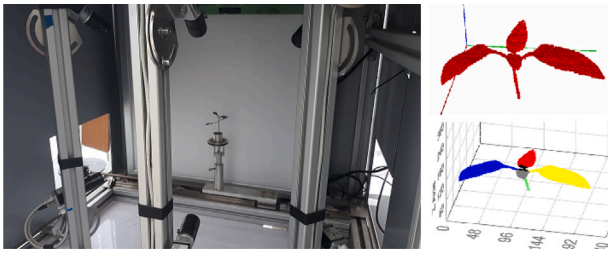

Fig. 2. Multi-view camera system used to capture the 2D images of a plant. The right top shows the reconstructed 3D plant model. The right bottom shows the same 3D plant model segmented in leafs and stem.

unlabelled data while training the system, as well as the effect on performance of various modifications to the architecture and learning procedures. Finally, conclusions are drawn from the work and presented.

2. Materials and methods

2.1. Data collection

2.1.1. Plant material

The seedlings were grown from commercial seeds, consisting of 9 different tomato cultivars. In total, 11 seed batches have been grown. At the time of recording, the seedlings were between 11 and 16 days old. Overall, there were 6732 scanned plants, of which 4287 normal and 2405 abnormal plants, are used in this paper. Table 1 contains a breakdown of these quantities per dataset. Datasets 1, 4 and 5 consist each of two different cultivars while Datasets 3, 6, 7, 8 and 9 contain plants from a single cultivar. The seeds used for Datasets 1, 3, 6, 7, 8 and 9 were known to result in a relatively high percentage of abnormals. Datasets 4 and 5 contain abnormal plants that were rejected by a propagator as UT. Plants in Datasets 8 and 9 are grown from the same seed batch, but plants in Dataset 8 contain only normals.

2.1.2. Imaging system

The datasets used in this paper are acquired using a MARVIN™ imaging system (Golbach et al., 2016) for seedling phenotyping or automated seedling sorting. Several of these systems have been developed by our group in recent years in cooperation with mechanical engineering companies. There are MARVIN™ systems for manual insertion of the seedling for laboratory use and automated systems for fast sorting using conveyor belts to pass the seedlings through the imaging system with automated rejection of abnormal seedlings. The latter are used in practice. In this research, a manual system has been used, consisting of 6 monochrome cameras and a colour camera to obtain images from different viewpoints.

The common principle of these imaging systems is a multi-camera setup to obtain images of the seedlings from different viewpoints. A multi-camera calibration procedure is applied to determine the mutual positions of the cameras and the mapping of 3D points in the measurement area to pixel coordinates of the cameras. In each camera image the seedling is segmented from the background by either using a combination of backlighting and thresholding for monochrome cameras or a front lighting and a colour segmentation method for colour cameras. A shape-from-silhouette (Koenderink et al., 2009) method is used to calculate a 3D reconstruction of the seedlings from the silhouettes, resulting in a list of 3D points occupied by the seedling, see Fig. 2.

2.1.3. Rule-based expert system and hand-crafted features

The 3D model obtained with a MARVIN system is further processed using traditional methods to calculate several features of the seedlings: the total plant volume, number of leaves, leaf length, leaf width, leaf area, stem length, stem width, stem angle and number of edge voxels (plant voxels that are also boundary voxels of the volume). The accuracy of the length and area features are evaluated by comparing with ground truth measurements obtained destructively by hand. The methods for calculating these features are described in Golbach et al. (2016). These features are combined in a manually adjusted classier to discriminate between normal and abnormal seedlings. In this paper we used these features for the rule-based expert system and the hand-crafted features.

For the rule-based expert system the classifier is based on the UT test and determined by plant volume and the number of leaves. A seedling is classified as non-germinated if the plant volume is less than a user defined minimum, set to 50 in this case. A seedling is abnormal if the plant has less than two leaves or the plant volume is less than 50% of the average plant volume of a particular batch. All other seedlings are considered normal. The average plant volume is based on all plants in the batch with more than two leaves and a minimum volume of 50. In this paper the classification is done per dataset. In practice the classification into normal and abnormal is carried out per tray and based on tray statistics. An important disadvantage of this rule-based classifier is that it mainly uses plant volume and it is not possible to discriminate between several types of morphological abnormality, this is one of the reasons why a deep learning based classification approach is investigated. For the hand-crafted features the plant volume, leaf area, stem length, stem width, stem angle, number of leaves and edge voxels are used in machine learning models as described in Section 3.3.

2.1.4. Seedling expert assessment

The plants were labelled by experts in three classes; normal, abnormal or not germinated. The experts were instructed to apply the Usable Transplant criteria, where all normal plants are those with a leaf surface of more than 25 percentage of the average leaf surface. Abnormal plants have a leaf area of less than 25 percentage of the average leaf surface or other morphological abnormalities. Dataset 1 was labelled by four experts (combined by rounded averaging of the numeric scale). The other datasets were labelled by a single expert.

2.2. Neural network models

2.2.1. Architecture

To test the potential for deep neural networks to classify seedlings, we adopt an autoencoder-based approach similar to that of Achlioptas et al. (2018). A high-level representation of the relationship amongst system components is shown in Fig. 3.

We learn representations of the 3D seedling point clouds in an unsupervised fashion using an autoencoder, consisting of an encoder

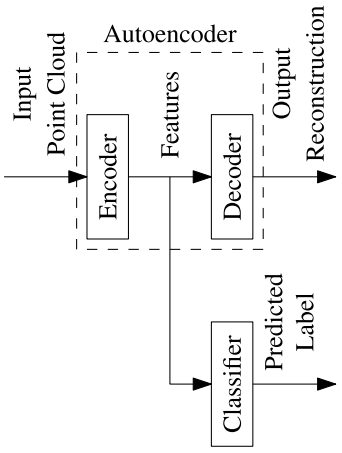

Fig. 3. High-level system architecture.

and decoder. Here, the autoencoder learns to reconstruct the input point cloud using the decoder from bottleneck features extracted by the encoder. Following this, a classifier can be built by supervised training of additional classifier layers that use the autoencoder features as input. Our choices for encoder and decoder are discussed within the following paragraphs. Two encoders were tested. One encoder was based on PointNet (Charles et al., 2017), with the "global feature" of 1024 values extracted by the classification network in that paper mapped by an MLP (multilayer perceptron) with hidden units 512, 256 and 256. The fully connected MLP layers are each followed by ReLU (rectified linear unit) activations and batch normalization. Batch normalization is omitted after the last fully connected layer. The other encoder tested was like PointNet++ (Qi et al., 2017) in that it consisted of a sequence of sampling, grouping and pointnet operations, thus aggregating information in a hierarchical fashion, as shown in Fig. 4. There were, however, four sample/group/pointnet stages reducing the number of points by a factor of 8 at each step. Because the point clouds are set to contain $8^4 = 4096$ points (achieved by resampling the original point clouds), this means that the last output is associated with a single point, and aggregation through a max operation is not necessary to produce the final bottleneck features output by the encoder to the decoder. Grouping was performed using k-nearest neighbours ($k = 16$) instead of by ball query, as this avoids the need for setting query ball radius parameters. MLPs forming part of the pointnet operations consisted of two linear units with the same number of hidden units, each followed by batch normalization and a ReLU activation. The number of hidden units for these layer pairs in the four pointnet MLPs are 32, 64, 128 and 256 respectively.

Instead of a fully connected decoder directly predicting N output points (Achlioptas et al., 2018; Zamorski et al., 2020), we formulate the decoding process as the transformation of one distribution (we choose a Gaussian distribution) to one shaped like the input point cloud (assuming the model is well-trained). This is performed by an MLP that has as input the 256 bottleneck values and 64 Gaussian noise values. The MLP then maps noise, conditioned on the bottleneck values, through hidden layers with sizes 256, 128, 64, 32 and 3 respectively (all but the last layer are followed by rectified linear activation and batch normalization). The result is a 3D point, which is a sample from the transformed Gaussian distribution. In practice, we parallelize the process such that 4096 points are generated simultaneously (each point derives from the same code but different Gaussian noise inputs). Together, these points form the reconstructed point cloud.

This decoder network, illustrated in Fig. 5, has some advantages over (Achlioptas et al., 2018; Zamorski et al., 2020) in that the number of output points are decoupled from the output network architecture itself, since the number of output neurons (and thus the number of output weights) is independent of the number of output points. This should make the architecture more scalable to higher numbers of output points. It also allows for changing the number of output points during training or testing to balance point cloud fidelity and speed of processing. This, for example, allows one to choose the number of output points for visualization purposes, where increasing the amount of output points may help clarify the shape of the distribution. Finally, when a classifier is required, a classifier stage is added to the network, taking the encoder's output as its input. For this, we employed an MLP with the number of hidden units in the layers given by 128, 64 and 32. The MLP input is the size of the code (256) from the encoder, and the MLP output size is 2 (the number of classes, normal/abnormal). Each layer (except for the final layer) is followed by ReLU activation and batch normalization. The final layer has a softmax activation.

2.2.2. Training

Autoencoders were trained using the Chamfer Distance (Achlioptas et al., 2018) as error metric. This is given by the sum of the square distances from each point in one of the two point clouds to its closest point in the other point cloud. The distance can be expressed as

$$d_{CH}(S_1, S_2) = \sum_{x \in S_1} \min_{y \in S_2} \|x - y\|_2^2 + \sum_{y \in S_2} \min_{x \in S_1} \|x - y\|_2^2 \quad (1)$$

where S_1 and S_2 are sets of points representing point clouds.

Autoencoders were trained using Adam (Kingma and Ba, 2015) with learning rate set initially to 10^{-3}, lowering by a factor of 0.5 until at lowest 10^{-5} each time the learning stagnates for 10 epochs as measured on the validation set. When combining an autoencoder with a classifier, three approaches to training were followed, as described below:

- Certain model encoder/decoders were trained in a standard unsupervised way, with subsequent supervised training of a separate classifier (without adjusting the encoder). We refer to this kind of training as "indirect", because of the unsupervised training step that precedes classifier training.

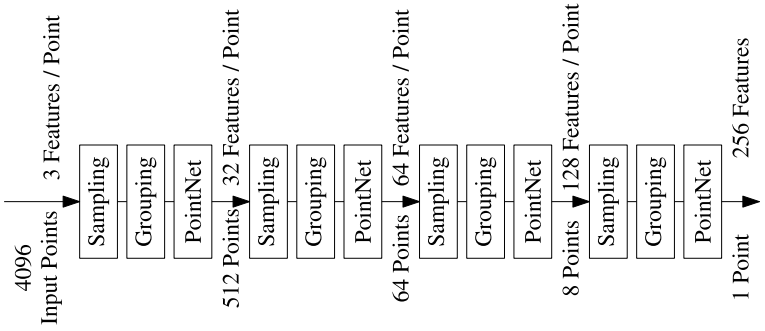

Fig. 4. PointNet++-based Encoder.

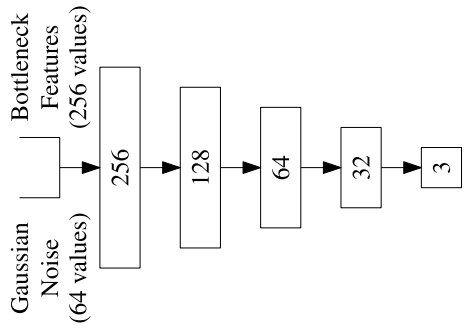

Fig. 5. Decoder architecture.

- In other models, the aforementioned training procedure is followed, except in the final step allowing the encoder and classifier to be refined together. We label this kind of training "refinement", because of the refinement step.
- Finally, in certain models we forego pretraining and train the encoder and classifier together in a supervised fashion. We call this kind of training "direct", because of the immediate training of the classifier.

In all three of these cases, the Adam-based training procedure described previously for the autoencoders is used when training components, except during refinement when the learning rate is preset to the minimum of 10^{-5} to discourage large changes.

3. Results and discussion

3.1. Expert labelling agreement

All plants in Dataset 1 were assessed by four experts. Cohen's κ (McHugh, 2012) is a metric to assess the agreement between raters. It takes into account the possibility that agreement could occur by chance. For example, Cohen's κ may assume the value of 0 if there is only agreement that can be attributed to chance (negative values representing worse disagreement are possible), or 1 for complete agreement.

Cohen's κ was run on both tomato cultivars separately to determine the agreement between the different experts on whether the plants were normal or abnormal. For tomato cultivar A and B, the agreement between the experts was substantial. For all expert couples the κ was between 0.71 and 0.82 for cultivar A and between 0.74 and 0.82 for cultivar B.

3.2. Existing expert system (rule-based)

Cohen κ can also be used as a performance measure or accuracy metric for automatic classification and can handle datasets with imbalanced classes. For Dataset 1, the rule-based expert system has been compared to the human experts. The expert system is included as an additional expert and the agreement between the expert system and the human experts is also substantial for both cultivars. The κ is between 0.67 and 0.71 and between 0.74 and 0.81 for cultivar A and B, respectively.

The rule-based classification is based on the UT criteria to predict whether a plant is normal or abnormal. The number of correct predicted normals and abnormals is high for datasets 3, 6, 7 and 9. Correct classification for normals in these datasets are 98.6%, 98.6%, 96.5% and 97.6%, respectively. For abnormals (including non-germinated predictions) the correct classification rates are 50%, 78.9%, 95.5% and 84.7%. The plants in these datasets are comparable to practice, where the majority of the plants are normal. The system performs significantly worse for datasets, such as Dataset 4–5 and 8, with an unusual distribution of normal and abnormal plants. The rules do not suffice in this case because plant morphological properties are neglected and tray statistics are used that are derived from these unusual distributions of plants.

3.3. Classification on handcrafted features

In addition to the rule-based expert system, we consider taking the features provided by the MARVIN™ system and use them in machine learning models. This provides another helpful set of performance baselines. For these experiments, we utilized seven features: the plant volume, leaf area, stem length, stem width, stem angle, number of leaves and edge voxels (the amount of volume edge voxels occupied by the seedling). Feature vectors are normalized by centring to the feature medians from the training set and divided by the per-feature difference between the 25th and 75th quantiles of features from the training set. We consider three kinds of machine learning models: Support Vector Machines, SVM (RBF) (SVM with Radial Basis Function kernels) (Burges, 1998), Random Forest (Breiman, 2001) and relatively shallow MLPs. On each dataset, model hyperparameters are first optimized using a grid search and five-fold cross validation on the training/validation data.

Table 2
Handcrafted feature classification results (Normal = NRM, Abnormal = ABN). The rule-based expert system uses batch statistics requiring whole batches/datasets, therefore only results for Dataset 6 are given. Extra metrics are provided in Table A.1.

Test drawn from:	All Datasets		Dataset 6	
	NRM	ABN	NRM	ABN
SVM (RBF)	90.9%	84.1%	89.6%	85.7%
Random Forest	93.4%	78.8%	93.6%	83.9%
MLP	93.6%	79.1%	92.0%	82.1%
Rule-based	–	–	98.6%	78.7%

The results from these classification experiments are given in Table 2. For each machine learning method, two sets of results are reported. One set derives from drawing the test set from all datasets. The other uses Dataset 6 exclusively as a test set. For comparison, these dataset choices correspond to subsequent testing on deep neural networks.

3.4. Deep neural networks

In this section, we describe the results obtained from experimentation with the deep neural network architectures described in Section 2.2.1. We first turn our attention to purely unsupervised learning and the quality of the learned representations. Following this, we explore the use of these models in classification tasks with different degrees of dataset labelling (partially labelled datasets).

3.4.1. Unsupervised learning

As discussed in Section 2.2.1, we employ an autoencoder architecture as the core of the system. Subsequent to training this autoencoder, a classifier can be built operating on the learned features. In this section, we focus first on the autoencoder itself and evaluate the quality of the learned features through visualization.

We train a PointNet-based model using the Chamfer Distance as described in Section 2.2.2. Dataset 6 is used exclusively for testing later classifiers, while 95% and 5% of the other datasets are each used for training and validation data respectively. One way of examining autoencoders is by looking at their output reconstructions, and how well they match that at the input. In Fig. 6, some input and reconstruction pairs of the PointNet-based model are shown. The reconstructions generally follow the shape of the input point cloud. However, they lack sharpness. Experimentation with architectural changes such as a larger bottleneck (1024 instead of 256) did not address this. It can be seen that, while the reconstructions are not high-fidelity replicas of the inputs, they do capture some important structural aspects of inputs. The question instead arises whether the learned representations within the autoencoder might be useful for later classification tasks.

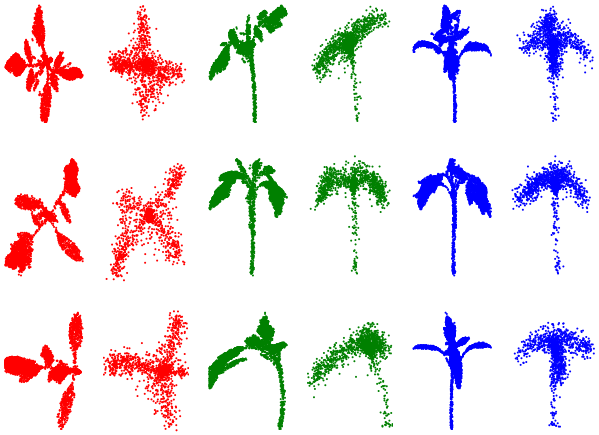

Fig. 6. Sample inputs and their autoencoder reconstructions. Each row shows three sides (red, green and blue) of a seedling, with each side being shown both as input (left) and reconstruction (right).

Fig. 7. t-SNE embeddings of the PointNet-based autoencoder features from each seedling. Each point is coloured by its class (green/normal or orange/abnormal).

To investigate this question, t-SNE embeddings (Van Der Maaten and Hinton, 2008) were calculated for the features generated by the PointNet-based network for each seedling. These embeddings are plotted in Fig. 7. Each point in the figure represents one seedling, coloured by the ground truth knowledge of whether or not it is abnormal.

It can be seen from the figure that the model appears to have learned something potentially useful for a subsequent classifier. We can see this from the relatively large clustered groups of green and orange, indicating the model has learned to group either normal or abnormal seedlings together in those regions of feature space. Relatively few orange points are scattered amongst predominantly green areas and vice versa, meaning this source of misclassifications is limited. While the autoencoder reconstructions do not have high fidelity, the features extracted by it show promise in normal/abnormal classification. We now consider adding a classifier to the model and evaluate its performance.

3.4.2. Partially labelled datasets

To what extent may unlabelled data help us attain higher performance? In this section, we explore classification scenarios illustrating how the autoencoder configuration can have an advantage over standard classifiers when unlabelled data is available. The results of this section are summarized in Table 3.

Table 3
Comparison of normal (NRM)/abnormal (ABN) classification results for fully and partially labelled cases. Extra metrics are provided in Table A.2.

Training Set		Test Set	NRM	ABN
Encoder (Unlabelled)	Classifier (Labelled)			
Dataset 1	Dataset 1	Dataset 1	98.0%	88.2%
Dataset 1	Dataset 1	Dataset 6	98.1%	76.0%
Dataset 1,3–5,7–9	Dataset 1	Dataset 6	97.7%	82.7%
Dataset 1,3–5,7–9	Dataset 1,3–5,7–9	Dataset 6	92.8%	87.5%

Typically, a core dataset with labels is available for the supervised part of training, as well as a larger unlabelled dataset. To emulate such a setup, we choose to use Dataset 1 and its labels for training (and some testing). We employ Dataset 6 and its labels purely for testing. All other datasets are, for this experiment, considered unlabelled and are used only for unsupervised training.

We first consider to what extent Dataset 1 is, by itself, sufficient for building predictive models. We split Dataset 1 as 80%, 5% and 15% training, validation and test data respectively. We then train autoencoders on the training data. Subsequently, we train classifiers using the labels. We find for the PointNet-based model that normals and abnormals are classified correctly 98.0% and 88.2% of the time. However, exposure to data from a different data source may affect these results. Therefore, we investigate what happens when the same model is applied to Dataset 6. We find that the correct classification rates for normals and abnormals are 98.1% and 76.0%. Noting the relatively poor performance in terms of the abnormals, it appears that Dataset 1 by itself did not present the model with enough variety to result in sufficient generalization.

We can attempt to remedy this situation by introducing the additional unlabelled data into the training process (while the autoencoder is training). Therefore, we add the unlabelled point clouds of Datasets 3 through 5, 7 through 9, and Dataset 1's former test set to the training data and rerun training and testing. This resulted in normals and abnormals being correct 97.7% and 82.7% of the time, respectively. Here, the model only saw 13.6% of the labels of the 5484 seedlings from the training set. Particularly apparent is the increase in the correct classification rate for abnormals by 6.7%, making a substantial contribution towards the deployability of the system.

What would the relative contribution of having fully labelled data have been? We can answer this question by introducing the expert labels from Datasets 3 through 5 and 7 through 9 into the training data. We find that the correct classification rates for normals and abnormals are 92.8% and 87.5% respectively. This result is interesting in that the model which has seen fewer labels actually has a performance advantage in terms of the normals (a difference of 4.9%), but is weaker in terms of identifying abnormals correctly (a difference of 4.8%). This may be because the fraction of abnormals in Dataset 1 (21%) is lower than when combining all datasets used for training (41%). This means the classifier may have shifted towards attempting better classification of abnormals, once the label data from the other datasets was introduced.

For perspective, we compare with representative results on handcrafted features from Table 2, which were obtained using a Random Forest classifier giving 93.6% and 83.9% for normal and abnormal correct classification rates. This is comparable to the just presented deep learning result. However, we show in a subsequent section that models based on deep learning can still substantially outperform those based on handcrafted features.

In this section, we have found that, for seedling quality inspection, unlabelled data can provide a substantial boost in the performance of resulting models. In the following section, we investigate variants of the system to maximize performance, in particular when full labels are available.

Table 4
Results for normal (NRM)/abnormal (ABN) classification with Dataset 6 as test set. Extra metrics are provided in Table A.3.

Training	Encoder			
	PointNet		PointNet++	
	NRM	ABN	NRM	ABN
Indirect	**92.8%**	**87.5%**	86.1%	82.1%
Refinement	92.4%	80.4%	90.4%	78.6%
Direct	92.7%	87.5%	92.7%	78.6%

Table 5
Results for normal (NRM)/abnormal (ABN) classification with test data from every dataset. Extra metrics are provided in Table A.4.

Training	Encoder			
	PointNet		PointNet++	
	NRM	ABN	NRM	ABN
Indirect	96.2%	94.4%	95.9%	92.0%
Refinement	96.4%	96.3%	96.1%	92.4%
Direct	95.8%	93.1%	**97.6%**	**96.1%**

3.4.3. Model variants

In this section, we evaluated a number of model variants to find potentially better choices for further development. Models varied in which encoder they used (PointNet- or PointNet++-based, see Section 2.2.1). They also varied in how they were trained, as described in Section 2.2.2. As in the preceding experiment, Dataset 6 was the test set, while all other datasets each contributed 95% and 5% to the training and validation sets respectively. We report the results of the different variations in Table 4 where the correct classification rates for normals and abnormals are reported for each variant.

What is immediately apparent from these results is that the original system (PointNet-based with indirect training) performed best (92.8% and 87.5% for correct classification of normals and abnormals respectively), with one close entry where direct training was performed. This result highlights the competitiveness of approaches with purely unsupervised feature learning (indirect training). Supervised refinement or training of the encoder does not necessarily lead to the best performance in all scenarios. However, the results may be influenced by aspects particular to Dataset 6. We therefore consider a different experiment where we divide each dataset into 80%, 5% and 15% training, validation and test data respectively. Each dataset then contributes the appropriate subsets to the total training, validation and test sets. As in the previous experiments, we report the variations in Table 5, similar to Table 4.

It is immediately evident that performance is greater overall for each of the model variants with this division of the datasets, with the best performance being obtained using the PointNet++-based model with direct training at correct classification rates of 97.6% and 96.1% for normals and abnormals respectively. Notably, this is substantially better than results of corresponding experiments with handcrafted features in Table 2 (“All Datasets” column). Furthermore, here we do observe performance improvements from using refinement. Direct training leads to the highest performance for the PointNet++-based model, but methods incorporating unsupervised training still lead to the best performance for the PointNet-based model. While no longer the highest performing variant, it should be noted that the indirect and refinement PointNet-based networks show comparable performance. Because they have the advantage of being able to make use of unlabelled data, they may still be preferred in situations where a large amount of such data is available.

4. Conclusion

Detection of abnormal seedlings can help forecast the number of saleable plants for plant propagators. With current methods, not all types abnormalities can be flexibly detected, which is why a deep learning-based approach is investigated in this paper. A partially supervised approach is taken where rich representations are learned from 3D data using autoencoders in an unsupervised manner. These representations are later refined with limited labelled data in a supervised setting. An autoencoder architecture with novel decoder stage is proposed and investigated on PointNet- and PointNet++-based encoder backbones. A model refinement strategy is also employed to take maximum benefit from expensive and limited expert labels.

The system described in this paper achieves performance approaching that of human experts in some cases, however some limitations to this study need to be taken into account. System performance may be affected by changes such as: (1) use of other imaging systems and 3D reconstruction settings (delivering different point cloud fidelity), (2) using other plant varieties or species, (3) variations in seedling growing conditions (for example, growing times that depart from those of the study’s), (4) abnormalities not present in the current dataset.

Taking these factors into account, the key conclusions from the present study are the following. In the scenarios investigated, we found that being able to utilize the unlabelled data in a partially labelled dataset improved the system’s classification of abnormals from 76.0% to 82.7%. This is substantial progress towards the abnormal classification rate of 87.5% when the system is provided fully labelled data.

When considering various improvements, we found that a system which can incorporate unlabelled data could achieve correct classification rates of 96.4% and 96.3% for normals and abnormals respectively. This was comparable to the best performing model, which used direct training and reached correct classification rates of 97.6% and 96.1% for normals and abnormals respectively.

So, while the best performing model did not use unsupervised training, methods incorporating unsupervised training were shown to be competitive. In addition, models incorporating unsupervised training would be particularly suited to the case where much unlabelled data is available, alleviating the labelling burden when creating new datasets.

Finally, while the reconstruction of input seedlings by the autoencoder with PointNet-based encoder does not have high fidelity, it is able to learn features potentially useful for normal/abnormal classification tasks. This was seen, in particular, in Fig. 7. However, this certainly suggests an important direction for future investigation. Can better combinations of encoder and decoder be found, perhaps improving the output reconstruction quality and, more importantly, perhaps the quality of the code as well? Furthermore, testing on other crops may help investigate the flexibility of the system.

We demonstrated that semi-supervised learning and partial labelling could greatly reduce the cost of data collection, with limited impact on accuracy. This enables the continuous data collection after initial model training. A key question remains: which partial data is most suitable for improving model performance? Future research should consider the role of active learning in selecting the right data elements to label.

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.

Data availability

The data that has been used is confidential.

Acknowledgements

This publication was produced as part of the public private partnership (PPS) project ‘Exploitation of high-tech plant phenotyping tools for breeding companies and growers’ with PPS number HT-17222 which is funded by the Dutch Topsector TKI (Topconsortium for Knowledge and Innovation) Agri&Food (Anon, 2021). The authors acknowledge also financial support from the European Commission through partnership in the STARGATE project (H2020/No. 952339): https://stargate-hub.eu/.

Appendix. Additional results

In this section, expanded versions of earlier result tables are presented. These provide extra evaluation metrics including confusion matrices for each experiment. Note that, because certain experiments were run multiple times to improve metric estimation, the confusion matrices provided here may sum to more than the number of plants in the relevant dataset.

Table A.1
Extended version of Table 2. Handcrafted feature classification results (Normal=NRM, Abnormal=ABN). The rule-based expert system uses batch statistics requiring whole batches/datasets, therefore only results for Dataset 6 are given.

Test drawn from dataset(s)	Classifier	Normal correct rate	Abnormal correct rate	F1 score	F1 score (Balanced)	Accuracy	Accuracy (Balanced)	Confusion matrix
All	SVM (RBF)	0.909	0.841	0.844	0.884	0.884	0.875	$\begin{bmatrix}579 & 58\\ 60 & 318\end{bmatrix}$
All	Rnd. Forest	0.934	0.788	0.830	0.878	0.880	0.861	$\begin{bmatrix}595 & 42\\ 80 & 298\end{bmatrix}$
All	MLP	0.936	0.791	0.833	0.880	0.882	0.863	$\begin{bmatrix}596 & 41\\ 79 & 299\end{bmatrix}$
6	SVM (RBF)	0.896	0.857	0.485	0.914	0.894	0.877	$\begin{bmatrix}810 & 94\\ 8 & 48\end{bmatrix}$
6	Rnd. Forest	0.936	0.839	0.584	0.940	0.930	0.888	$\begin{bmatrix}846 & 58\\ 9 & 47\end{bmatrix}$
6	MLP	0.920	0.821	0.529	0.928	0.915	0.871	$\begin{bmatrix}832 & 72\\ 10 & 46\end{bmatrix}$
6	Rule-based	0.986	0.787	0.787	0.973	0.973	0.886	$\begin{bmatrix}891 & 13\\ 13 & 48\end{bmatrix}$

Table A.2
Extended version of Table 3. Comparison of normal (NRM)/abnormal (ABN) classification results for fully and partially labelled cases. To improve metric estimates, the two middle rows were repeatedly evaluated 13 times by retraining the classifier, obtaining the joint confusion matrix on the right. These rows are critical, because they are used to quantify the improvement derived from semi-supervised learning.

Training set of encoder (Unlabelled)	Training set of classifier (Labelled)	Test set	Normal correct rate	Abnormal correct rate	F1 score	F1 score (Balanced)	Accuracy	Accuracy (Balanced)	Confusion matrix
1	1	1	0.980	0.882	0.909	0.955	0.956	0.931	$\begin{bmatrix}99 & 2\\ 4 & 30\end{bmatrix}$
1	1	6	0.981	0.760	0.734	0.968	0.968	0.870	$\begin{bmatrix}11526 & 226\\ 175 & 553\end{bmatrix}$
1,3–5,7–9	1	6	0.977	0.827	0.751	0.969	0.968	0.902	$\begin{bmatrix}11479 & 273\\ 126 & 602\end{bmatrix}$
1,3–5,7–9	1,3–5,7–9	6	0.928	0.875	0.576	0.937	0.925	0.902	$\begin{bmatrix}839 & 65\\ 7 & 49\end{bmatrix}$

Table A.3
Extended version of Table 4. Results for normal (NRM)/abnormal (ABN) classification with Dataset 6 as test set.

Encoder	Training approach	Normal correct rate	Abnormal correct rate	F1 score	F1 score (Balanced)	Accuracy	Accuracy (Balanced)	Confusion matrix
PointNet	Indirect	0.928	0.875	0.576	0.937	0.925	0.902	$\begin{bmatrix}839 & 65\\ 7 & 49\end{bmatrix}$
PointNet	Refinement	0.924	0.804	0.529	0.930	0.917	0.864	$\begin{bmatrix}835 & 69\\ 11 & 45\end{bmatrix}$
PointNet	Direct	0.927	0.875	0.573	0.936	0.924	0.901	$\begin{bmatrix}838 & 66\\ 7 & 49\end{bmatrix}$
PointNet++	Indirect	0.861	0.821	0.404	0.890	0.858	0.841	$\begin{bmatrix}778 & 126\\ 10 & 46\end{bmatrix}$
PointNet++	Refinement	0.904	0.786	0.471	0.915	0.897	0.845	$\begin{bmatrix}817 & 87\\ 12 & 44\end{bmatrix}$
PointNet++	Direct	0.927	0.786	0.530	0.931	0.919	0.856	$\begin{bmatrix}838 & 66\\ 12 & 44\end{bmatrix}$

Table A.4
Extended version of Table 5. Results for normal (NRM)/abnormal (ABN) classification with test data from every dataset. Because PointNet++ includes randomized processes at runtime, these tests were run 20 times, leading to a joint confusion matrix which sums to a higher value than the number of test samples.

Encoder	Training approach	Normal correct rate	Abnormal correct rate	F1 score	F1 score (Balanced)	Accuracy	Accuracy (Balanced)	Confusion matrix
PointNet	Indirect	0.962	0.944	0.941	0.956	0.956	0.953	$\begin{bmatrix}613 & 24\\ 21 & 357\end{bmatrix}$
PointNet	Refinement	0.964	0.963	0.952	0.964	0.964	0.963	$\begin{bmatrix}614 & 23\\ 14 & 364\end{bmatrix}$
PointNet	Direct	0.958	0.931	0.930	0.948	0.948	0.944	$\begin{bmatrix}610 & 27\\ 26 & 352\end{bmatrix}$
PointNet++	Indirect	0.959	0.920	0.925	0.945	0.945	0.940	$\begin{bmatrix}12221 & 519\\ 603 & 6957\end{bmatrix}$
PointNet++	Refinement	0.961	0.924	0.929	0.947	0.947	0.943	$\begin{bmatrix}12242 & 498\\ 571 & 6989\end{bmatrix}$
PointNet++	Direct	0.976	0.961	0.960	0.970	0.970	0.968	$\begin{bmatrix}12434 & 306\\ 295 & 7265\end{bmatrix}$

References

Achlioptas, P., Diamanti, O., Mitliagkas, I., Guibas, L., 2018. Learning representations and generative models for 3D point clouds. In: Proceedings of the 35th International Conference on Machine Learning. Vol. 80. PMLR, pp. 40–49, URL: http://proceedings.mlr.press/v80/achlioptas18a.

Ani Brown Mary, N., Robert Singh, A., Athisayamani, S., 2020. Banana leaf diseased image classification using novel HEAP auto encoder (HAE) deep learning. Multimedia Tools Appl. 79 (41), 30601–30613. http://dx.doi.org/10.1007/s11042-020-09521-1.

Anon, 2021. Automatic germination test provides insight into tomato seed quality. URL: https://www.wur.nl/en/newsarticle/Automatic-germination-test-provides-insight-into-tomato-seed-quality.htm.

Ashraf, M.A., Kondo, N., Shiigi, T., 2011. Use of machine vision to sort tomato seedlings for grafting robot. Eng. Agric. Environ. Food 4 (4), 119–125. http://dx.doi.org/10.1016/s1881-8366(11)80011-x.

Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32. http://dx.doi.org/10.1023/A:1010933404324.

Burges, C.J., 1998. A tutorial on support vector machines for pattern recognition. Data Min. Knowl. Discov. 2 (2), 121–167. http://dx.doi.org/10.1023/A:1009715923555.

Chalapathy, R., Chawla, S., 2019. Deep learning for anomaly detection: A survey. arXiv preprint arXiv:1901.03407.

Charles, R.Q., Su, H., Kaichun, M., Guibas, L.J., 2017. PointNet: Deep learning on point sets for 3D classification and segmentation. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition. CVPR, IEEE, pp. 77–85. http://dx.doi.org/10.1109/CVPR.2017.16, URL: http://ieeexplore.ieee.org/document/8099499/.

Chen, Z., Zhang, H., 2019. Learning implicit fields for generative shape modeling. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, IEEE, pp. 5932–5941. http://dx.doi.org/10.1109/CVPR.2019.00609, URL: https://ieeexplore.ieee.org/document/8953765/.

Dobos, O., Horvath, P., Nagy, F., Danka, T., Viczián, A., 2019. A deep learning-based approach for high-throughput hypocotyl phenotyping. Plant Physiol. 181 (4), 1415–1424. http://dx.doi.org/10.1104/pp.19.00728.

Golbach, F., Kootstra, G., Damjanovic, S., Otten, G., Van De Zedde, R., 2016. Validation of plant part measurements using a 3D reconstruction method suitable for high-throughput seedling phenotyping. Mach. Vis. Appl. 27 (5), 663–680. http://dx.doi.org/10.1007/s00138-015-0727-5.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press, URL: http://www.deeplearningbook.org.

Kingma, D.P., Ba, J.L., 2015. Adam: A method for stochastic optimization. In: Proceedings of the 3rd International Conference on Learning Representations. URL: http://arxiv.org/abs/1412.6980.

Koenderink, N.J., Wigham, M., Golbach, F., Otten, G., Gerlich, R., Van De Zedde, H.J., 2009. MARVIN: High speed 3D imaging for seedling classification. In: Precision Agriculture 2009 - Papers Presented At the 7th European Conference on Precision Agriculture. ECPA 2009, pp. 279–286, URL: https://edepot.wur.nl/177811.

McHugh, M.L., 2012. Interrater reliability: the kappa statistic. Biochemia Medica 22 (3), 276–282. http://dx.doi.org/10.11613/BM.2012.031, URL: http://www.biochemia-medica.com/en/journal/22/3/10.11613/BM.2012.031.

Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S., 2019. DeepSDF: Learning continuous signed distance functions for shape representation. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, IEEE, pp. 165–174. http://dx.doi.org/10.1109/CVPR.2019.00025, URL: https://ieeexplore.ieee.org/document/8954065/.

Qi, C.R., Yi, L., Su, H., Guibas, L.J., 2017. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In: Proceedings of the 31st International Conference on Neural Information Processing Systems. pp. 5105–5114, URL: https://papers.nips.cc/paper/2017/hash/d8bf84be3800d12f74d8b05e9b89836f-Abstract.html.

Ruff, L., Van Der Meulen, R.A., Görnitz, N., Binder, A., Müller, E., Müller, K.-R., Kloft, M., 2019. Deep semi-supervised anomaly detection. arXiv preprint arXiv:1906.02694.

Samiei, S., Rasti, P., Ly Vu, J., Buitink, J., Rousseau, D., 2020. Deep learning-based detection of seedling development. Plant Methods 16 (1), 103. http://dx.doi.org/10.1186/s13007-020-00647-9, URL: https://plantmethods.biomedcentral.com/articles/10.1186/s13007-020-00647-9.

Strothmann, L., Rascher, U., Roscher, R., 2019. Detection of anomalous grapevine berries using all-convolutional autoencoders. In: IGARSS 2019 - 2019 IEEE

International Geoscience and Remote Sensing Symposium. IEEE, pp. 3701–3704. http://dx.doi.org/10.1109/IGARSS.2019.8898366, URL: https://ieeexplore.ieee.org/document/8898366/.

Trang, K., TonThat, L., Thao, N.G.M., 2020. Plant leaf disease identification by deep convolutional autoencoder as a feature extraction approach. In: 2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology. ECTI-CON, IEEE, pp. 522–526. http://dx.doi.org/10.1109/ECTI-CON49241.2020.9158218.

Van Der Burg, W., Aartse, J., van Zwol, R., Jalink, H., Bino, R., 1994. Predicting tomato seedling morphology by X-ray analysis of seeds. J. Am. Soc. Hortic. Sci. 119 (2), 258–263. http://dx.doi.org/10.21273/JASHS.119.2.258.

Van Der Maaten, L., Hinton, G., 2008. Visualizing data using t-SNE. J. Mach. Learn. Res. 9 (86), 2579–2605, URL: https://www.jmlr.org/papers/v9/vandermaaten08a.html.

Wang, H., Bah, M.J., Hammad, M., 2019. Progress in outlier detection techniques: A survey. IEEE Access 7, 107964–108000. http://dx.doi.org/10.1109/ACCESS.2019.2932769.

Yang, Y., Feng, C., Shen, Y., Tian, D., 2018. FoldingNet: Point cloud auto-encoder via deep grid deformation. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. IEEE, pp. 206–215. http://dx.doi.org/10.1109/CVPR.2018.00029, URL: https://ieeexplore.ieee.org/document/8578127/.

Zamorski, M., Zięba, M., Klukowski, P., Nowak, R., Kurach, K., Stokowiec, W., Trzciński, T., 2020. Adversarial autoencoders for compact representations of 3D point clouds. Comput. Vis. Image Underst. 193 (April 2020), 102921. http://dx.doi.org/10.1016/j.cviu.2020.102921, URL: https://linkinghub.elsevier.com/retrieve/pii/S107731422030014X.