Machine Learning for Maize Plant Segmentation

Simon Donné, Hiep Quang Luong, Bart Goossens, Wilfried Philips SIMON.DONNE@UGENT.BE
iMinds - IPI - UGent

Stijn Dhondt, Nathalie Wuyts, Dirk Inzé
PSB - VIB, Plant Biotechnology and Bioinformatics - UGent

Keywords: big data, deep learning, convolutional networks, image segmentation

Abstract

High-throughput plant phenotyping platforms produce immense volumes of image data. Here, a binary segmentation of maize colour images is required for 3D reconstruction of plant structure and measurement of growth traits. To this end, we employ a convolutional neural network (CNN) to perform this segmentation successfully.

1. Input description

PHENOVISION is an automated plant phenotyping system for crops under greenhouse conditions. A conveyor belt system transports pots from a stationary growth area to irrigation stations and imaging cabins (capturing both RGB, thermal emittance and infra-red reflectance). Maize genotypes may differ in their response to specific abiotic stresses, such as drought or nutrient deficiency. A 3D reconstruction of maize plants can be used to measure the effects on growth. The cameras capture the maize plant from six separate side-view angles (30° apart) as well as a top-view, as shown in Figure 1. Segmented images are used to obtain a voxel cloud, representing plant structure in 3D (Figure 2).

The problem of segmentation is two-fold: an accurate segmentation is required for precise 3D reconstruction, whereas PHENOVISION captures an enormous amount of data on a daily basis Hence a fast technique is required that can process input images as fast as they are captured - on the order of 100 milliseconds per image segmentation, as the 3D reconstruction also requires computation time.

Appearing in *Proceedings of Benelearn 2016*.

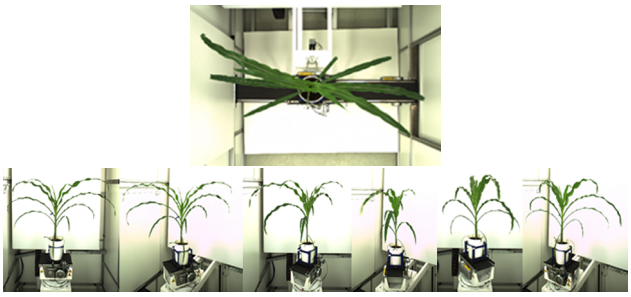

Figure 1. Example of the input images: six side views and one top view.

Figure 2. Example of the output: 3D reconstructions of a maize plant over time.

2. CNN design

Initial attempts at segmentation were based on linear combinations of colour channels and thresholds. In comparison, the application of Convolutional Neural Nets (CNN) is a generalization of this concept. CNNs consist of several layers of convolutions with pre-trained filter banks, interspersed with activation functions that are often simply taken to be ReLUs: rectified linear units (Glorot et al., 2011). They have been previously applied to segmentation of e.g. MRI images (Powell et al., 2008) with great success, outperforming alternative techniques. Here a CNN with four layers is used, consisting of a convolutional part followed by ReLU activation functions.

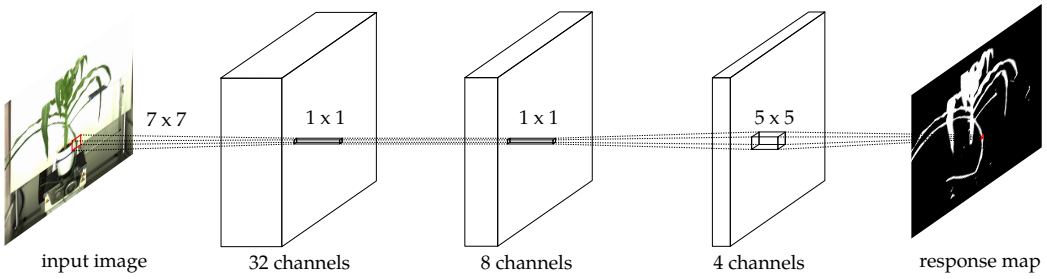

Figure 3. Overview of the neural network design.

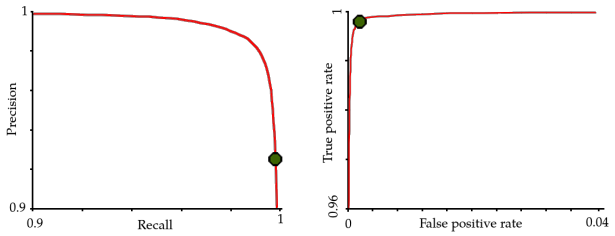

Figure 4. Precision-recall and ROC curves for the final response maps, as a function of the classification threshold. The highlighted point is a threshold of 0.5.

Figure 5. Example of the segmentation resulting from the trained CNN. Some false positives exist, but visual inspection shows only very few false negatives.

The first layer is responsible for transforming the RGB image into a 32-channel low-level feature image. The next two layers have no spatial influence, and simply serve to allow for a larger degree of non-linearity using the ReLU activation functions: they combine the 32 feature channels into four higher level channels. The final layer combines these four channels into a single response map, which can then be thresholded to arrive at the segmentation. The network is shown in Figure 3

3. Training the network

The machine is trained using stochastic gradient descent (Bottou, 2010), with additional momentum terms and a shrinkage rate introduced on the filters to mitigate the effect of local minima (Ngiam et al., 2011; Sutskever et al., 2013). The cost function used here is the penalization of poorly classified pixels, with an enforced gap of size 1. If the estimated response of a location (x, y) is given by $\tilde{g}(x, y)$ and the groundtruth response is $g(x, y)$ (either 0 or 1), then the cost corresponding $\Phi(x, y)$ to this location is given by

$$\Phi(x,y) = \begin{cases} \max(\tilde{g}(x,y),0)^2, & \text{where } g(x,y)=0 \\ \max(1-\tilde{g}(x,y),0)^2, & \text{where } g(x,y)=1 \\ 0, & \text{elsewhere} \end{cases} .$$

The third case is included because the regions near the image borders are denoted as *don't care*.

This was done in order to avoid boundary issues for pixels that are of little interest in any case. The input images were manually segmented: a series of 10 plants in various stages of growth were annotated.

4. Results

In the ideal case, the CNN response map contains values lower than 0 for all background pixels, and values higher than 1 for all maize plant pixels.

In practice, there are some misclassifications, and the threshold needs to be selected to perform the binary classification. This threshold allows the user to choose between type I and type II errors: either too many background pixels are classified as maize plant pixels (false positives) or maize plant pixels are misclassified as background (false negatives). Figure 4 shows a precision-recall and ROC curve for the threshold choice. It implies that the training of the machine optimizes the point on the ROC curve (the point for the midway of the separation zone, i.e. 0.5, is highlighted). Figure 5 shows an example output segmentation.

References

Bottou, L. (2010). Large-scale machine learning with stochastic gradient descent. In *Proceedings of compstat'2010*, 177–186. Springer.

Glorot, X., Bordes, A., & Bengio, Y. (2011). Deep sparse rectifier neural networks. *International Conference on Artificial Intelligence and Statistics* (pp. 315–323).

Ngiam, J., Coates, A., Lahiri, A., Prochnow, B., Le, Q. V., & Ng, A. Y. (2011). On optimization methods for deep learning. *Proceedings of the 28th International Conference on Machine Learning (ICML-11)* (pp. 265–272).

Powell, S., Magnotta, V. A., Johnson, H., Jammalamadaka, V. K., Pierson, R., & Andreasen, N. C. (2008). Registration and machine learning-based automated segmentation of subcortical and cerebellar brain structures. *NeuroImage*, *39*, 238 – 247.

Sutskever, I., Martens, J., Dahl, G., & Hinton, G. (2013). On the importance of initialization and momentum in deep learning. *Proceedings of the 30th international conference on machine learning (ICML-13)* (pp. 1139–1147).