LaTeX TITLE HERE

First Author
Institution1
Institution1 address
firstauthor@i1.org

Second Author
h Institution2
First line of institution2 address
secondauthor@i2.org

December 4, 2019

Abstract

1 Introduction

Plant architecture is a complex geometrical object. Due to both genetic and environmental variations, organs evolve in patterns which present great diversity both between species and among individuals of the same species. Automated reconstruction of plant structure from lab or in-field acquisitions remains a challenge in computer vision (ref). The most common method for complex measurements of plant structure remains handmade measurements and image analysis is a major bottleneck in plant phenotyping [5]. The application of automated processing of plant structure are many: precise quantifying of biomass and yield for agricultural crops, estimation of environmental response of crops [6], mapping genotypes to phenotypes, estimating growth parameters of species, among other. (Find many refs)

Deep convolutional networks have shown – see for example [3] – to be the state of the art method for many classification tasks. Several networks ([8], [4]) leverage huge classification datasets (ImageNet) to provide pixel by pixel semantic segmentation. They have been successfully used for plant organ segmentation [9], but are still limited to simple case with a very small diversity in test datasets. Using simulated data to augment training datasets is a method which have shown to improve neural network performance in many fields, e.g [7], [1].

Plant architecture is a well studied research topic, and many generative model of plant architecture have been developped in the last decades. Lindenmayer systems are rewriting systems specifically developped to model plant growth. They can be used to model arbitrary complex model of plants [2].

In this work, we propose to use plant models to train convolutional neural network for identification of different plant organs. The specific target application of our method is the identification of organs of the model plant Arabidopsis thaliana. We describe a data augmentation model using plant models and HDRI pictures to produce ground truth images, as well as a simple method for fine-tuning on real word data. We then present qualitative results in various acquisition condition to assert the robustness of our method.

2 Related works

3 Material and Methods

Virtual plants. The virtual plants were designed with the Python library OpelAlea developped by french research institutes of biology and mathematics to provide tools for plant architecture modelling. The ROMI partners from Virtual Plants team in INRIA Grenoble provided virtual 3D meshes of Arabidopsis Thaliana designed with the L-Py library from OpenAlea, which is a python implementation of L-system. L-systems were developped in 1968 by Aristid Lindenmayer [] to model plant growth. It is a generative grammar that allows to grow a virtual plant using symbols, shapes and constraints derived for plant growth observation.

Formally it is called a rewriting system, or formal grammar. It comprises:

- A vocabulary V containing the *modules* of the system. For plant generation it will represent an architectural element of the plant (apex, internode, leaf) and associated parameters (age, length, etc)
- An initial *axiom* or state s_0 corresponding to the virtual plant at t_0. It is a string of elements from the vocabulary.
- A set of *production* rules to iterate in order to model the growth. They will be applied in parallel to each variable element from the string of the previous state. They are composed of a *predecessor*, to identify the elements that will be replaced by a *successor*.

The strings can then be represented graphically in 3D using OpenAlea's PlantGL (Figure1).

"figures/""blank".png

Figure 1: Left: Example of an L-Py system with two modules, right: Visual representation of the first five steps of the L-system comprising apex (green dots) and internodes (brown sticks) in the vocabulary. Figure from [2]

The A. Thaliana provided by INRIA comprised 5 different types of organs: fruit, stem, peduncle, leaf and flower, and geometrical data such as organ lengths and angles between organs, in order to produce a ground truth for every step of the pipeline. They were provided as meshes, with a material associated to each organ.

Virtual scanner. An API was implemented to generate and visualise the virtual plants in Blender. The 3D models are loaded in a virtual environment and virtual cameras are used to take pictures of the model (Figure 3). The API was used to simulate a virtual scanner and generate images for the training of neural networks. The position, pan and tilt of the cameras is tunable and allows to generate camera paths to take images from different views. It is also possible to move the plant in the scanner. First the plants were placed on a highy contrasted scene. After attending the Image Analysis Methods in Plant Science (IAMPS) workshop we improved the virtual scanner by adding a 3D background [] which reproduces realistic lightening to increase the representation space of the virtual plants.

"figures/""blank".png

Figure 2: Illustration of 2D overlapping due to perspective projection

The virtual labels are acquired by rendering the plant material by material, with one material per organ (Figure 3). The organs considered for *A. Thaliana* are leaf, stem, flower, fruit and peduncle. In the pinhole model [] we use for 3D to 2D projection, a whole line in 3D projects onto a single pixel (Figure 2). As a consequence, a single pixel can belong to several classes depending on the organs crossed by the line. We decided that the labelling method should take this plurality into account rather then keeping only the class of the organ closer to the virtual camera. Therefore, for each pixel and 6 classes (including background), there are $2^6 = 64$ possible labels. To save the labels in a binary image, it was natural to encode the labels as binary numbers and convert them to decimal numbers. If from one viewpoint the background, a leaf and the stem project onto the same pixel due to perspective projection, the label will be $L_2 = 100110$, and the value of the pixel will therefore be $L_{10} = 38$.

"figures/""blank".png

Figure 3: Example of virtual training images and all labels

4 Results and Discussions

5 Conclusion and perspectives

References

[1] Hassan Abu Alhaija, Siva Karthik Mustikovela, Lars Mescheder, Andreas Geiger, and Carsten Rother. Aug-

mented reality meets computer vision: Efficient data generation for urban driving scenes. *International Journal of Computer Vision*, 126(9):961–972, 2018. 1

[2] Frederic Boudon, Christophe Pradal, Thomas Cokelaer, Przemyslaw Prusinkiewicz, and Christophe Godin. L-Py: An L-System Simulation Framework for Modeling Plant Architecture Development Based on a Dynamic Language. *Frontiers in Plant Science*, 3, 2012. 1, 2

[3] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 1

[4] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 1

[5] Massimo Minervini, Hanno Scharr, and Sotirios A Tsaftaris. Image analysis: the new bottleneck in plant phenotyping [applications corner]. *IEEE signal processing magazine*, 32(4):126–131, 2015. 1

[6] Laura Soledad Peirone, Gustavo Pereyra Irujo, Alejandro Bolton, Ignacio Erreguerena, and Luis AN Aguirrezabal. Assessing the efficiency of phenotyping early traits in a greenhouse automated platform for predicting drought tolerance of soybean in the field. *Frontiers in plant science*, 9:587, 2018. 1

[7] Weichao Qiu and Alan Yuille. Unrealcv: Connecting computer vision to unreal engine. In *European Conference on Computer Vision*, pages 909–916. Springer, 2016. 1

[8] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 1

[9] Weinan Shi, Rick van de Zedde, Huanyu Jiang, and Gert Kootstra. Plant-part segmentation using deep learning and multi-view vision. *Biosystems Engineering*, 187:81–95, 2019. 1