Computers and Electronics in Agriculture 150 (2018) 196–204

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

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

Original papers

Seed-per-pod estimation for plant breeding using deep learning

L.C. Uzal[a,*], G.L. Grinblat[a], R. Namías[a], M.G. Larese[a], J.S. Bianchi[b], E.N. Morandi[b], P.M. Granitto[a]

[a] CIFASIS, French Argentine International Center for Information and Systems Sciences, CONICET - UNR, Argentina
[b] Laboratorio de Fisiología Vegetal, Facultad de Ciencias Agrarias, Universidad Nacional de Rosario & Instituto de Investigaciones en Ciencias Agrarias de Rosario (IICAR), CONICET - UNR, Argentina

ARTICLE INFO

Keywords:
Deep learning
Machine vision
Plant breeding programs
Plant phenotyping
Soybean

ABSTRACT

Commercial and scientific plant breeding programs require the phenotyping of large populations. Phenotyping is typically a manual task (costly, time-consuming and sometimes arbitrary). The use of computer vision techniques is a potential solution to some of these specific tasks. In the last years, Deep Learning, and in particular Convolutional Neural Networks (CNNs), have shown a number of advantages over traditional methods in the area. In this work we introduce a computer vision method that estimates the number of seeds into soybean pods, a difficult task that usually requires the intervention of human experts. To this end we developed a classic approach, based on tailored features extraction (FE) followed by a Support Vector Machines (SVM) classification model, and also the referred CNNs. We show how standard CNNs can be easily configured and how a simple method can be used to visualize the key features learned by the model in order to infer the correct class. We processed different seasons batches with both methods obtaining 50.4% (FE + SVM) and 86.2% (CNN) of accuracy in test, highlighting the particularly high increase in generalization capabilities of a deep learning approach over a classic machine vision approach in this task. Dataset and code are publicly available.

1. Introduction

Plant phenotyping can be defined as the identification and quantification of effects on the phenotype (i.e., the appearance and behavior of plants), using appropriate protocols and measurements, as result of both genotype differences and the interaction with the environment (Fiorani and Schurr, 2013). In their search for increased yields, plant breeding programs require the phenotyping of large populations, evaluating some or even several useful traits (Ghanem et al., 2015). In addition, the results must be validated through multiple environments and replicated trials, increasing the burden of the process. Unfortunately, phenotyping is typically a manual task, therefore laborious, costly, and time-consuming. Even more, visual evaluation over many segregating plants in the field is very difficult and error-prone due to the observer subjectivity. As a consequence, phenotyping has become a bottleneck for plant breeding programs (Singh et al., 2016). Thus, the possibility of developing quick, accurate and repeatable methods to characterize individual plant phenotypes will increase the quality of the selection process and will provide a useful tool to foster the incorporation of desirable traits into commercial germplasm.

In recent years, imaging-based automatic methods have been introduced to plant phenotyping (Fahlgren et al., 2015; Scharr et al., 2016), using diverse devices and multiple scales. For example, Bendig et al. (2014) used aerial images to estimate the biomass of entire crops. Giuffrida et al. (2015) developed an efficient method for counting leaves in rosette plants with images of individual plants in a controlled environment in the context of the Leaf Counting Challenge held in the CVPPP 2015 workshop. This challenge led to ongoing research which is using deep learning techniques, (Aich and Stavness, 2017; Dobrescu et al., 2017) for instance. Also (Pound et al., 2017) used deep learning for localising wheat spikes and spikelets.

In particular, the yield of a soybean crop depends on three major components: the number of pods per plant (PN), the number of seeds per pod (SPP) and the seed size (Fehr, 1987). Under cultivation conditions, yield is subject to strong genotype-environment interaction. However, of the three main components, SPP is the least subject to environmental influence (i.e. it is a characteristic of the genotype (Board and Harville, 1998)) thus offering the opportunity to genetically manipulate it to improve the yield potential of a cultivar.

The main goal of any breeding program is to develop varieties with high yield potential. The capability of selecting for traits linked to yield components such as high PN and SPP, early during the selection process, would increase the efficiency of the breeding program. This implies to cope with large genetic-engineering experiments consisting in the order of tens of thousands plants to be manually labeled pod by pod, among the other previous mentioned features, to select the best

* Corresponding author.
E-mail address: uzal@cifasis-conicet.gov.ar (L.C. Uzal).

https://doi.org/10.1016/j.compag.2018.04.024
Received 21 December 2017; Received in revised form 6 April 2018; Accepted 21 April 2018
0168-1699/ © 2018 Elsevier B.V. All rights reserved.

phenotype.

Counting the number of pods per plant is a simple but tedious task that can be easily automated. Furthermore, estimating the number of seeds per pod is laborious and difficult as well, requiring visual inspection of each pod by a human expert. The difficulty of the task relies on the wide range of maturing sizes of seeds within the pods as a result of environmental or genetic factors. All seeds must be counted even if they have suffered an abortion in the early stages of pod development. In such limit cases a trained expert can still visually distinguish subtle changes in pod shape evidencing the presence of an aborted seed. As a measure of the difficulty of the task, we point out that a group of trained operators achieve an accuracy of 84% (with an standard deviation of 2%).[1]

In this work we introduce an automated tool that could replace human experts in this counting task, allowing the increase in scale of breeding programs without losing accuracy and, in consequence, the speed up of the complete process. There are some precedents for automatic object counting from digital images in the context of agricultural applications (Dorj et al., 2017; Harmsen and Koenderink, 2009; Liu et al., 2017; Maldonado and Barbosa, 2016; Mussadiq et al., 2015; Aich and Stavness, 2017; Dobrescu et al., 2017; Pound et al., 2017). However, our case is different as we need to infer the number of seeds –that are hidden– from the pod shape.

Convolutional Neural Networks (CNNs) have proven to be very effective at solving vision problems in a wide range of fields and they have been one of key elements of the success of Deep Learning (LeCun et al., 2015). In the context of agricultural applications, recent years have witnessed a growing tendency to replace classic techniques with deep learning algorithms for a variety of vision tasks (Grinblat et al., 2016; Ding and Taylor, 2016; Sladojevic et al., 2016; Lu et al., 2017; Tang et al., 2017). Even a typical phenotyping problem, counting leaves, has been tackled with these techniques (Ubbens and Stavness, 2017). Recently, a complete survey of Deep Learning in agriculture has been published (Kamilaris and Prenafeta-Boldú, 2018). According to this survey our application is new and will fall into the area of plant phenology recognition where only two paper were surveyed (Yalcin, 2017; Namin et al., 2017).

The advantages of the Deep Learning approach for vision tasks are twofold: there is no need to carefully design handcrafted features extractors for the problem at hand, as CNNs can learn specialized features extractors from raw data, and when provided with enough data, CNNs generally reach higher accuracies than classic techniques (LeCun et al., 2015). An important drawback of neural networks is the high number of hyperparameters associated to the model architecture design (such as number of layer, units per layer, etc) and the training algorithm (learning rate, momentum) as well. Finding appropriate values may feel like a challenging task to an inexperienced user. However, current deep learning libraries offer default values for hyperparameters which are good starting points for optimal search.

The aim of this paper is to tackle the problem of counting the SPP number for soybean pods with a Deep Learning approach based on standard Convolutional Neural Networks and to compare the results with a classic approach based on a set of tailored features extracted specifically for this problem and a SVM classifier. We also offer an extensive study of how hyperparameters selection impacts on the CNN's performance on this task. This paper also aims to contribute to collecting evidence in favor of the widespread use of Deep Learning for agricultural applications even for users not specialized in these tools.

[1] In order to further illustrate the difficulty of the classification task, we have intentionally included in Fig. 2 some hard samples together with more common cases.

2. Background

2.1. Classic approach

The classic procedure in machine vision is to define and extract appropriate features for the problem at hand and then train a classifier (e.g. an SVM with Gaussian kernel) in the corresponding representation space (as done for example in (Wu et al., 2007; Pydipati et al., 2006; Golzarian and Frick, 2011)). Part of our team firstly tackled this problem with such a classic strategy.

In a preliminary stage, three state-of-the-art classic classification methods were implemented, namely SVM, Random Forest (RF) (Breiman, 2001) and Penalized Discriminant Analysis (PDA) (Hastie et al., 1995). The cross-validation error for the three methods were found to be comparable, with a slight difference in favor of SVM. For this reason, in this paper SVM was chosen as the classic classification method for comparison purposes against CNN.

Regarding the handcrafted features, many geometrical (Umbaugh, 2005) and shape (Hu, 1962) features were considered at first. Lately, the addition of a 25-bin histogram of the profile of the pod shown to improve classification. Alternatively, an ad hoc method of ellipse fitting of beans inside the pod was designed, but it did not show any improvement.

2.2. Convolutional neural networks

CNNs, introduced by LeCun et al. (1990), have an architecture specially designed to process images. Its topology leads to a huge reduction in the number of free trainable parameters in comparison to a standard (fully connected) artificial neural network. This is due to its sparse neural connectivity (restricted to small receptive fields) and to the sharing of filter values along image locations exploiting translational invariance. In the following we briefly describe CNN's architecture mainly with the purpose of defining the hyperparameters considered in this work. For a more detailed description of this kind of models, we refer the reader to (LeCun et al., 2010) and references therein.

Fig. 4 depicts a diagram of the considered CNN model, inspired on the VGG architecture (Simonyan and Zisserman, 2014). Each layer is composed of three transforms. First, there is a convolution operation between the input image and a filter bank. Each filter has a bounded size associated to a small receptive field in the input image, typically 3×3 or 5×5 filter sizes are considered. For each filter in the bank, the convolution produces a feature map. Together with the convolution operation a subsampling step may be introduced. This replaces the standard average or max pooling operation by simply setting an stride larger than 1 in the convolution transform (Springenberg et al., 2014). A subsampling of factor 2 in each dimension of the feature map retains 1/4 of the output values (those with even indexes). This subsampling is usually accompanied with a duplication of the number of feature maps with respect to previous convolutional layers.

Second, a Batch Normalization (BN) transform (Ioffe and Christian, 2015) is applied after each convolution. This transform standardizes convolution output by fixing maps mean and deviation to 0 and 1 respectively over small batches of samples. It then applies a learnable gain and bias to each feature map. This layer has a beneficial regularization effect that has been verified in numerous machine vision applications (Vinyals et al., 2015; Radford et al., 2015; He et al., 2016), making it a standard tool.

Finally, the third transform is an element-wise nonlinear function applied to all feature maps. We use in all cases the Leaky ReLU function[2] (Maas et al., 2013), which is widely used to enhance the back-propagation signal.

[2] $LReLU(x) = \max(0.01x, x)$.

Download English Version:

https://daneshyari.com/en/article/6539379

Download Persian Version:

https://daneshyari.com/article/6539379

Daneshyari.com