New perspectives on plant disease characterization based on deep learning
Sue Han Lee, Hervé Goëau, Pierre Bonnet, Alexis Joly

To cite this version:

HAL Id: hal-02470280
https://hal.umontpellier.fr/hal-02470280
Submitted on 9 Mar 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives| 4.0 International License
New perspectives on plant disease characterization based on deep learning

Sue Han Leea,b, Hervé Goëaua,b, Pierre Bonneta,b, Alexis Jolyca

ARTICLE INFO

Keywords:
- Plant diseases
- Automated visual crops analysis
- Deep learning
- Transfer learning

ABSTRACT

The control of plant diseases is a major challenge to ensure global food security and sustainable agriculture. Several recent studies have proposed to improve existing procedures for early detection of plant diseases through modern automatic image recognition systems based on deep learning. In this article, we study these methods in detail, especially those based on convolutional neural networks. We first examine whether it is more relevant to fine-tune a pre-trained model on a plant identification task rather than a general object recognition task. In particular, we show, through visualization techniques, that the characteristics learned differ according to the approach adopted and that they do not necessarily focus on the part affected by the disease. Therefore, we introduce a more intuitive method that considers diseases independently of crops, and we show that it is more effective than the classic crop-disease pair approach, especially when dealing with disease involving crops that are not illustrated in the training database. This finding therefore encourages future research to rethink the current de facto paradigm of crop disease categorization.

1. Introduction

A plant disease is an alteration of the original state of the plant that affects or modifies its vital functions. It is mainly caused by bacteria, fungi, microscopic animals or viruses, and has a strong impact on agricultural yields and on farm budget. According to the Food and Agriculture Organization of the United Nations, transboundary plant diseases have increased significantly in recent years due to globalization, trade, climate change and the reduction in the resilience of production systems due to decades of agricultural intensification. The risk of transboundary epidemics is increasing and can cause huge losses in crops, threatening the livelihoods of vulnerable farmers and the food and nutritional security of millions of people. Early detection of disease symptoms is one of the main challenges in protecting crops and limiting epidemics. Initial disease identification is usually done by visual assessment (Barbedo, 2016) and the quality of the diagnosis depends heavily on the knowledge of human experts (Liu et al., 2017). However, human expertise is not easily acquired by all actors of the agricultural world, and is less accessible, especially in the case of small farms in developing countries.

Automatic recognition of plant diseases by image analysis represents a promising solution to overcome this problem and reduce the lack of expertise in this field (Ramcharan et al., 2017). Several studies have been carried out on isolated crops, such as maize (Wiesner-Hanks et al., 2018), cassava (Ramcharan et al., 2017), tomato (Fuentes et al., 2017; Durmus et al., 2017; Fuentes et al., 2018), apple (Liu et al., 2017), wheat (Johannes et al., 2017; Picon et al., 2018), citrus (Iqbal et al., 2018) or potato (Oppenheim and Shani, 2017). Deep Learning (DL) techniques, including Convolutional Neural Networks (CNN), have emerged as the most promising approaches given their ability to learn reliable and discriminative visual characteristics. Transfer learning, which is used in most cases (Slado Jevic et al., 2016; Too et al., 2019; Fuentes et al., 2017; Liu et al., 2017; Mohanty et al., 2016; Ramcharan et al., 2017), is not a single technique, but a whole family of methods, comprising, among others those commonly known as fine-tuning. Unlike learning from scratch where the weights of a model are learned from scratch, the weights of a pre-trained model on a large general dataset (in terms of number of images and classes, such as ImageNet) are fine-tuned. Transfer learning allows to build accurate models even on a specialized dataset such as those containing plant diseases where there are usually a small number of images and classes compared to ImageNet. Indeed, when used to training a CNN model based on images of diseased and healthy plant leaves captured under controlled conditions, it has been proven that fine-tuning is better than training a deep model from scratch (Mohanty et al., 2016). However, the ability of a deep learning model to transfer knowledge from one domain to
another, from one task to another, is not yet well understood and performances may vary according to the used datasets for pre-training a model (Torralba and Efros, 2011). In particular, (Zhou et al., 2014) showed that fine-tuning a CNN pre-trained on scene recognition datasets rather than the general Imagenet dataset can perform better in any urban scene recognition tasks. We can then assume that a model pre-trained on a dataset of images dedicated to plant identification could provide better features for plant disease identification rather than a pre-trained model on ImageNet which contains actually few visual botanical concepts. However, no previous study has reported such results. The first contribution of this article is to study the impact of transfer learning depending on whether the transfer learning is from a general domain or from the plant domain.

In recent years, more and more studies address the multi-crop disease problems where data modelling covers several varieties of crops and diseases at the same time (Mohanty et al., 2016; Too et al., 2019; Slado Jevic et al., 2016; Ferentinos, 2018). These investigations correspond to the needs of actors such as farm technicians, agricultural engineers, market gardeners or arboriculturists, who would potentially be confronted with different types of diseases on different types of crops, and where a more general approach (particularly through the use of smartphones) could provide a valuable help (Mohanty et al., 2016). From a technological point of view, this type of use is now possible to highlight irrelevant crop features like the shape of the leaf, and in particular, visually similar diseases with the same disease common name may be present in several classes. Most of the previous studies based on CNNs follow this data organization, where the last layer dedicated to the classification contains output of crop-disease pairs (Mohanty et al., 2016; Too et al., 2019; Slado Jevic et al., 2016; Ferentinos, 2018).

However, although previous studies have reported high identification performances, it is possible that the models learned are not optimal by extracting to many irrelevant features related more to crop than to disease. For example, as reported in (Mohanty et al., 2016), it has been proven that removing the background with image segmentation is less effective than using the ordinary colored background images to train a CNN, confirming a dependency of background features for disease identification. Furthermore, (Toda and Okura, 2019) showed that model which is supposed to learn plant disease visual appearance, tends to highlight irrelevant crop features like the shape of the leaf, and in fact, it is possible when a crop contains more visual discrimination characteristics than the disease itself such as a deeply grooved leaf of tomato. Therefore, the second contribution of this paper is to propose a new intuitive way of characterizing plant diseases in the context of multi-crop diseases, focusing on the common names of diseases regardless of the type of crop. We show that this approach is more generalizable and robust in identifying diseases of new data taken in domains different from those in the training set.

Next, we consider that it would be very difficult, if not impossible, in a real-world application to collect a complete set of images illustrating each crop-disease pair. Also, it may not be easy to establish a specific list of all diseases for each host species. We note that pathogens with a wide host range can infect many different host species. For example, as reported in (Prospero and Cleary, 2017), nearly 200 plant species can be infected with bacterial wilt (Ralstonia solanacearum). Therefore, the ability of a model to transfer knowledge from one disease context to another is crucial for practical application to reduce the time and cost of data collection and retraining a deep model. We show that by using our new strategy mentioned above to identify plant diseases, we could have the significant advantage of being able to identify a disease known to the classifier without the type of crop being known. Therefore, the third contribution of this paper is to study the generalization of a deep model to identify diseases of “unseen” crops. Unseen crops in this context refer to unknown crops that have never been seen by a deep model during the training.

To summarize the contributions of this paper:

1. We investigate the impact of two transfer learning strategies, one based on a pre-trained model on a plant dataset and the other using a general object dataset. We found that, it is sometimes more effective to fine-tune a model pre-trained on plant identification, or one which is pre-trained on general object recognition.

2. We qualitatively investigate the features learned from the models trained with crop-disease classes (Mohanty et al., 2016; Too et al., 2018). We highlight that the learned features might not necessarily be relevant to plant disease identification because they also focus on crop-specific features such as the leaf venation and lamina.

3. We demonstrate a simple and intuitive way of considering plant diseases alone, without any association with a crop. With this, it shown to enable a better generalization not only to new data taken in domains different from those in the training (different in data distribution), but also to unseen crops.

2. Datasets

2.1. Plant Village

Plant Village (PV) (Hughes and Salathé, 2015) is a popular dataset collected for evaluation of automatic plant disease identification systems. It contains healthy and infected leaves isolated on a uniform background. We used the initial version of the PV dataset2 that is still publicly available as a benchmark in this study. It has 38 crop-disease pairs, with 26 crop-disease categories for 14 crop plants. Note that, the data comes with predefined training and test subsets. In our study, we use the reported configuration as the one that produces the best identification performances in the original document (Mohanty et al., 2016), where 80% of the data is for the training set and the remaining 20% for the test (more precisely, 43,810 training images and 10,495 images for the test). Table 1 gives for the 38 pairs, the common names of the host plant and the disease, the scientific name of the disease and the number of training and testing images.

We can see in this table that different crops can be infected with pathogens associated with the same common disease name. It is mainly due to their similar ways of infecting the crops, which hence has led the agricultural community to call them by the same name. Table 2 illustrates the distribution of images based on a total of 14 crops and 20 common diseases3 with 1 healthy class. Some diseases such as Apple scab or Tomato mosaic virus can only be found in apple and tomato crops respectively, but other diseases such as Black rot or Bacterial spot are common in different crops.

---

2 https://github.com/spMohanty/PlantVillage-Dataset.
3 https://www.digipathos-rep.cnptia.embrapa.br/.

4 common names of plant diseases that have been recommended based on similar symptoms of potentially different pathogens.
Two complementary datasets named IPM and Bing, which were initially introduced in (Mohanty et al., 2016), are used in this work to evaluate the robustness of a model for predicting plant diseases on exterior images. Unlike in PV, the IPM test set allows to evaluate 19 pairs while Bing test images are no longer available online. Tables 3 and 4 show the distribution of the IPM and Bing images. Among the 38 crop-disease pairs shown in Fig. 1, shows that Bing images contain background noise (i.e. cluttered backgrounds) more often than IPM image, suggesting a more difficult recognition on Bing than on IPM.

### 3. Transfer learning and pre-trained models

Transfer learning is not a single technique, but a whole family of methods, comprising, among others those commonly known as fine-tuning. Using such approaches, a model trained on a task is re-purposed to a second related task (Goodfellow et al., 2016). A basic example of transfer learning is using a deep model just as fixed feature extractor (Karpathy, 2019), without tuning the weights of a pre-trained model and by removing the last fully connected layer related to the class names. This approach is more accurate than a “from scratch” approach where all the weights of a model are learned from scratch (Too et al., 2018; Mohanty et al., 2016; Slado Jevic et al., 2016; Huh et al., 2018).

Most of these previous works are based on models pre-trained on ImageNet (Russakovsky et al., 2015), a general image database containing about 1.2 million images associated to 1000 very diverse classes (animals, vehicles, buildings, landscapes, plants, etc.). Fine-tuning helps to prevent overfitting, leading to significantly better generalization especially when the number of examples for some source tasks but very few for some target tasks (Yosinski et al., 2014) showed that by transferring the features and fine-tuning all the layers of a network, one can boost the generalization capability of a model. Fine-tuning can be used to perform new classification. One of the most popular transfer learning approaches is to fine-tune the weights of a pre-trained model, with the last fully connected layer being replaced and randomly initialized on a new classification task. In fact, it is possible to fine-tune only some layers, which are generally the last layers corresponding to a higher level of abstraction. However, (Yosinski et al., 2014) showed that by transferring the features and fine-tuning all the layers of a network, one can boost the generalization capability of a model. Fine-tuning helps to prevent overfitting, leading to significantly better generalization especially when the number of labelled examples is scarce, or in a transfer setting where we have lots of examples for some source tasks but very few for some target tasks (LeCun et al., 2015). Additionally, it has been proven to be more effective if the pre-training and the target task are correlated (Yosinski et al., 2014; Huh et al., 2018).

Transfer learning is not a single technique, but a whole family of methods, comprising, among others those commonly known as fine-tuning. Using such approaches, a model trained on a task is re-purposed to a second related task (Goodfellow et al., 2016). A basic example of transfer learning is using a deep model just as fixed feature extractor (Karpathy, 2019), without tuning the weights of a pre-trained model and by removing the last fully connected layer related to the class names. This approach is more accurate than a “from scratch” approach where all the weights of a model are learned from scratch (Too et al., 2018; Mohanty et al., 2016; Slado Jevic et al., 2016; Ferentinos, 2018). Most of these previous works are based on models pre-trained on ImageNet (Russakovsky et al., 2015), a general image database containing about 1.2 million images associated to 1000 very diverse classes (animals, vehicles, buildings, landscapes, plants, etc.).
Table 2
Distribution of diseases for different types of crops. Note that, (%) is the percentage of images per row/column.

<table>
<thead>
<tr>
<th></th>
<th>Healthy</th>
<th>Apple scab</th>
<th>Black rot</th>
<th>Cedar apple rust</th>
<th>Powdery mildew</th>
<th>Cercospora leaf spot</th>
<th>Common rust</th>
<th>Northern leaf blight</th>
<th>Esca (Black Measles)</th>
<th>Leaf blight (Isariopsis Leaf Spot)</th>
<th>Haunglongbing (Citrus greening)</th>
<th>Bacterial spot</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apple</td>
<td>1645</td>
<td>630</td>
<td>621</td>
<td>275</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Blueberry</td>
<td>1502</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cherry</td>
<td>854</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Corn</td>
<td>3162</td>
<td></td>
<td></td>
<td></td>
<td>1052</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Grape</td>
<td>423</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Orange</td>
<td>1180</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Peach</td>
<td>360</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pepper bell</td>
<td>1278</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Potato</td>
<td>352</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Raspberry</td>
<td>371</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Soybean</td>
<td>5990</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Squash</td>
<td>1835</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Strawberry</td>
<td>456</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tomato</td>
<td>1591</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>15,084</td>
<td>630</td>
<td>1801</td>
<td>275</td>
<td>2887</td>
<td>513</td>
<td>1192</td>
<td>985</td>
<td>1383</td>
<td>1076</td>
<td>5507</td>
<td>2127</td>
</tr>
<tr>
<td>(%)</td>
<td>27.78</td>
<td>1.16</td>
<td>3.32</td>
<td>0.51</td>
<td>5.32</td>
<td>0.94</td>
<td>2.20</td>
<td>1.81</td>
<td>2.55</td>
<td>1.98</td>
<td>10.14</td>
<td>9.98</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Early blight</th>
<th>Late blight</th>
<th>Leaf scorch</th>
<th>Leaf Mold</th>
<th>Septoria leaf spot</th>
<th>Spider mites</th>
<th>Target Spot</th>
<th>Tomato Yellow Leaf Curl Virus</th>
<th>Tomato mosaic virus</th>
<th>Total</th>
<th>(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apple</td>
<td>3171</td>
<td></td>
<td>5.84</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Blueberry</td>
<td>1502</td>
<td></td>
<td>2.77</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cherry</td>
<td>1906</td>
<td></td>
<td>3.51</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Corn</td>
<td>3852</td>
<td></td>
<td>7.09</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Grape</td>
<td>4062</td>
<td></td>
<td>7.48</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Orange</td>
<td>5507</td>
<td></td>
<td>10.14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Peach</td>
<td>2475</td>
<td></td>
<td>4.56</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pepper bell</td>
<td>2512</td>
<td></td>
<td>3.96</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Potato</td>
<td>371</td>
<td></td>
<td>0.68</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Raspberry</td>
<td>5090</td>
<td></td>
<td>9.37</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Soybean</td>
<td>1835</td>
<td></td>
<td>3.38</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Squash</td>
<td>1109</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Strawberry</td>
<td>1565</td>
<td></td>
<td>2.88</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tomato</td>
<td>18,160</td>
<td></td>
<td>33.44</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>54,305</td>
<td></td>
<td>100</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(%)</td>
<td>3.64</td>
<td>5.36</td>
<td>2.04</td>
<td>1.75</td>
<td>3.26</td>
<td>3.09</td>
<td>2.59</td>
<td>9.86</td>
<td>0.69</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>
However, in the case of plant diseases, one can suppose that it would be appropriate to pre-train a model on domain with a dataset containing exclusively plant images. Therefore, in this study, we compare two fine-tuning strategies, one directly using a model pre-trained on ImageNet and another one using PlantCLEF2015, a dataset dedicated to plant identification (H. Goéau, P. Bonnet, A. Joly, LifeCLEF plant identification task, 2015). This dataset is related to the plant identification challenge in LifeCLEF (Joly et al., 2015) and addresses the problem of species identification based on multi-organ and multi-image observations of specimens. It has the same number of classes as ImageNet, but is two orders of magnitude smaller with precisely 91,759 training images (see Fig. 2 for some illustrations).

4. Identification based on common names of diseases and generalization to unseen crops

4.1. Identification based on visual features of common diseases

With regard to previous works on crop-disease pairs (Mohanty et al., 2016; Ferentinos, 2018), there remains a doubt that a deep model can be forced to discriminate similar diseases that have the same common name and most likely visually similar ways of infecting the crops. A potentially undesirable effect that may occur is that the model learns more from the content related to the crop than the visual content related to the disease (Toda and Okura, 2019; Mohanty et al., 2016). This adverse effect can potentially be amplified if a crop has a very pronounced characteristic such as toothed margins leaves, while the disease is not very visible, for example, with a slight bleaching at an early stage of infection. In this section, we study an alternative approach to the usual crop-disease pair modeling by considering only on common names of diseases regardless of crop categories. The hypothesis is that such intutive modelling will engage the model to learn more independent features from the host plants.

In order to train a disease classifier based on common diseases in-
into a single healthy class, so as to limit the amount to 2601 images in order to avoid imbalanced distributions (39,508 training images in total). During the inference, we calculate the classification result from the 21 probability outputs produced by the trained model. We call this disease modeling strategy $S_{c,d}$ which symbolizes the CNN modeling strategy based on the common name of diseases.

We compare $S_{c,d}$ with another disease modeling strategy called $S_{d,c}$ which instantiates the CNN modeling strategy based on the crop-disease pairs. In $S_{d,c}$, we first train a CNN based on 38 crop-disease pairs with a total of 43,810 training images. During the inference, from the 38 probability outputs produced by the trained model, we perform a late fusion method to integrate the probabilities corresponding to the individual common diseases. More precisely, we compute the average probability output with respect to each individual common name of disease for the conjoined crop-disease pairs. For example, the average probability of the Back rot common disease which is associated with apple and grape, is computed as:

$$P_{\text{Blackrot}} = \text{mean}(P_{\text{Apple Black rot}}, P_{\text{Grape Black rot}})$$

With this formulation, $S_{d,c}$ has the same number of accuracy outputs as $S_{c,d}$.

### 4.2. Generalization to unseen crops

There are several hundred common plant diseases in the world that can each of them potentially infect dozens or even hundreds of different types of crops. It is difficult, if not impossible, to collect a complete set of images illustrating each crop-disease combination for building an exhaustive training set. In addition, diseases can mutate and become transmissible to new cultivated species and then it is necessary to complete regularly the training set with new crop-disease combinations. On the other hand, an approach that considers diseases independently of crops has the advantage of making a classifier more easily extensible to new crops, by allowing a model to identify a known disease in the training dataset, even if the crop hosting the diseases on a test image is unknown in the training dataset. To evaluate the ability of a deep model to generalize to an unknown crop, in our methodology, all the images related to pepper crops such as those from the class Pepper_bell_Bacterial_spot and the class Pepper_bell_healthy are removed from the training set, and the model is trained only with those images which are not from pepper crop. It should be noted that, in doing so, the number of crop-disease pairs to be learned by a model is reduced from 38 to 36, giving a total of 41,762 and 37,727 images in the training set to train $S_{d,c}$ and $S_{c,d}$, respectively.

### 5. Approach

#### 5.1. Deep architectures

In our study, we employ three types of CNN architectures with different depth size: VGG16 (Simonyan and Zisserman, 20149) (16 layers), InceptionV3 (Szegedy et al., 2016) (48 layers) and our proposed architecture GoogLeNetBN7 (34 layers). This last architecture is inspired by the InceptionV2 and InceptionV3 architectures (Joffe and Szegedy, 2015; Szegedy et al., 2016) that bring several upgrades to the initial GoogLeNet architecture (Szegedy et al., 2014) for increasing the accuracy and reducing the computational complexity. It is a 34 layers deep CNN adding to each convolution a batch normalization (BN) operation. BN has been proven to speed up convergence and limit overfitting. Besides applying BN to GoogLeNet, we update the network by replacing the 5 × 5 convolutional layers with two consecutive layers of 3x3 convolutions like for the InceptionV3 architecture, so as to improve computational speed. The ImageNet dataset used for the ILSVRC challenge (Russakovsky et al., 2015), containing about 1.2 M images related to 1000 classes, was then used to train the model from scratch for a few days on a mini-cluster of GPU cards to produce a “pre-trained” model.

#### 5.2. Training from scratch versus transfer learning

All CNNs are trained by using stochastic gradient descent (SGD) optimization technique. The batch size was set to 45, and momentum set to 0.9. The 1.2 weight decay with penalty multiplier was set to 2 × 10⁻⁴. By employing the CNNs mentioned in Section 5.1 and training them based on the following training schemes, we will have 9 trained models based on 3 experimental configurations:

1. Transfer learning
   (a) Fine-tuning of ImageNet pre-trained model (FTIN)
   (b) Fine-tuning of PlantCLEF2015 pre-trained model (FTPC)

2. Training from scratch (FS)

For the FTIN configuration, a model is first pre-trained with ImageNet and then fine-tuned directly on the PV dataset. For the FTPC configuration, a model is first pre-trained with ImageNet, then fine-tuned on the PlantCLEF2015 dataset before being fine-tuned on the PV dataset.

For both configurations, all weights of the pre-trained models are fine-tuned simultaneously, with the last fully connected layer being replaced and initialized with target task classes. All the choices of training hyper-parameters were made based on empirical observation of the convergence of the network training as well as the training performance. For the two architectures VGG16 and GoogLeNetBN, the learning rate was initially set with a relatively low value of 10⁻⁴ in order to change the weights not too quickly, except for the last fully-connected layer where the learning rate was set to a higher value of 10⁻² to accelerate the convergence of the weights initialized with random values. For each model, the training are carried out for a total of 15 epochs, where one epoch is defined as the number of iterations required for the neural network to pass through the entire training set. The learning rate followed a “step” policy, meaning that it is decreased by a factor of 10 during the training, for every 15/3 epochs in our case. The architecture InceptionV3 with the configuration FTPC used the same hyper-parameters as the aforementioned two architectures, but when using the configuration FTIN on ImageNet the initial learning rate was set to 10⁻³ for all the layers because we noticed that the model converged too slowly with a low initial learning rate of 10⁻⁴. For the FS configuration, we set for all the three models an initial learning rate of 10⁻³ for all the layers. In this case, since all the weights are initialized with random values, we need more iterations, 30 epochs in these experiments, for the models to converge towards their best performances. Concerning the inputs, during the training, the image at the input of GoogLeNetBN and VGG16 is resized to 256 × 256 pixels and cropped to 224 × 224 pixels. As with InceptionV3, the image is resized to 300 × 300 pixels and cropped to 299 × 299 pixels. We augment our training image by random cropping and mirroring. For final model evaluations, each test image is cropped in its center and resized to the size imposed by the model architecture.

Fig. 3 shows the progression of the top-1 accuracy on the training set for all the different models and configurations limited to 15 epochs. We can see that the models learned with a fine-tuning approach all progress very quickly and tend to achieve a top-1 accuracy above 0.95 in less than 2 epochs, except for the InceptionV3 model pre-trained on ImageNet, which requires at least 7 epochs to reach an equivalent accuracy. We can also see that, in the FS configuration, the models converge more slowly and requires at least 11 epochs before reaching accuracies equivalent to the transfer learning configurations, except for the model with the VGG16 architecture which seems to stagnate around

---

7 https://github.com/AdelineMomo/CNN-plant-disease (Caffe).
an accuracy of 0.91.

6. Experiments

In this section, we first studied the impact of the use of transfer learning on the identification of plant diseases. Then, we evaluated a new classification problem considering common names of diseases as single classes, regardless of crop categories. Finally, we studied and analyzed the extent to which the deep models we learned can be generalizable to diseases despite the fact that the hosting crop is not illustrated in the training set.

6.1. Comparison of models with different learning configurations

Evaluation measures on the test sets. To measure model classification performance, the top-1 and top-3 accuracies were used to measure the performance of model classification in our experiments. Further, we also used the average accuracy per class to assess the performance of an individual class: $\text{AVE} = \frac{1}{C} \sum_{i=1}^{C} T_i$, where $T_i$ is the average accuracy for all test images related to the $C_i$ class and $C$ is the total number of classes. Next, to overcome the stochasticity of learning a single neural network, we used the same approach as described in (Mohanty et al., 2016), which combines the predictions of several trained models. Specifically, we aggregated and calculated the average of all the probabilities given by the models saved at the end of each epoch, then we calculated the average and overall accuracy of the entire test. By doing so, we are able to reduce the variance of the prediction results and obtain a fair comparison result with (Mohanty et al., 2016).

Generalization. Table 5 shows the classification performance of the CNNs trained with the different learning strategies. It can be seen that, although models trained with FS configuration, especially the deeper architectures, achieve relatively high performance in PV test set, they have lower classification performance in IPM and Bing than fine-tuned models. This result is consistent with the previous study (Mohanty et al., 2016), showing the advantage of fine-tuning in this task. The GoogLeNet model (Mohanty et al., 2016), which achieves the highest accuracy on the PV test set, shows lower performance on IPM images compared to GoogLeNetBN. This leads us to conclude that this model is particularly overfitted. Unlike what is generally observed in many other areas, the VGG16 model achieves the best performance on IPM and Bing test sets, 44.54% and 28.13% respectively, surpassing deeper architectures. This finding suggests that those models with deeper architectures are more likely to be overfitting for this task, probably because the PV dataset is not large or diverse enough.

Pre-trained models. It is difficult to conclude which pre-training task works best for the external IPM and Bing test data mainly because the size of these two set of images is quite small. However, by comparing the overall top-1 accuracy of the combined IPM and Bing images, the

<table>
<thead>
<tr>
<th>2^{\text{Architecture}}</th>
<th>2^{\text{Configuration}}</th>
<th>PV test set</th>
<th>IPM images</th>
<th>Bing images</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>top-1</td>
<td>top-3</td>
<td>AVE</td>
</tr>
<tr>
<td>GoogLeNetBN</td>
<td>FTIN</td>
<td>98.22</td>
<td>99.80</td>
<td>97.51</td>
</tr>
<tr>
<td></td>
<td>FTPC</td>
<td>99.09</td>
<td>99.95</td>
<td>98.83</td>
</tr>
<tr>
<td></td>
<td>FS</td>
<td>98.21</td>
<td>99.80</td>
<td>97.40</td>
</tr>
<tr>
<td>VGG16 (Simonyan and Zisserman, 2014)</td>
<td>FTIN</td>
<td>99.00</td>
<td>99.93</td>
<td>98.61</td>
</tr>
<tr>
<td></td>
<td>FTPC</td>
<td>98.56</td>
<td>99.90</td>
<td>97.84</td>
</tr>
<tr>
<td></td>
<td>FS</td>
<td>91.97</td>
<td>98.47</td>
<td>88.10</td>
</tr>
<tr>
<td>InceptionV3 (Szegedy et al., 2016)</td>
<td>FTIN</td>
<td>98.33</td>
<td>99.82</td>
<td>97.47</td>
</tr>
<tr>
<td></td>
<td>FTPC</td>
<td>98.46</td>
<td>99.89</td>
<td>97.71</td>
</tr>
<tr>
<td></td>
<td>FS</td>
<td>99.31</td>
<td>99.81</td>
<td>99.01</td>
</tr>
<tr>
<td>GoogLeNet (Mohanty et al., 2016)</td>
<td>FTIN</td>
<td>99.35</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

Fig. 3. Progression of top-1 accuracy on the training set for the 3 models and the 3 configurations.
VGG16 pre-trained with the ImageNet dataset shows a better result than the one pre-trained with PlantCLEF2015, which are 44.54% and 36.13% respectively. As for the GoogLeNetBN and InceptionV3, the model pre-trained on PlantCLEF2015 leads to better performance than the one pre-trained on the ImageNet dataset, unlike the VGG16. From this observation, it is unfortunately not possible to conclude on the usefulness or not of pre-training the models on a task closer to the target domain rather than the general object dataset, ImageNet.

**Confusions and misclassifications.** By analyzing the performance of each individual class of the PV test set (Supplementary Fig. 1), we found that most classification errors occur in the Tomato Early blight class. A closer analysis of the confusion matrix (Supplementary Table 1-6) shows that these errors generally result from confusion between the classes Tomato Early blight, Tomato Late blight, Tomato Septoria leaf spot and Tomato Target spot. Fig. 4 gives some examples of misclassified images related to these confusions and illustrates how they visually share similarities in leaf appearance and disease patterns. Although the disease Tomato Early blight may have specific visual characteristics, the intra- and inter-class variability makes their learning very difficult. This factor was discussed in (Barbedo, 2018), and one suggested solution was to extend the training dataset to a more exhaustive and complete set, covering all the visual variability of symptoms, which is in fact very difficult to collect. Next, we found that misclassification tends to occur within similar crops, particularly in maize and potatoes, and is also misleading due to visually similar leaf appearance and disease patterns.

Next, by analyzing the performance of the Bing images (Supplementary Table 7-12), we observed that with the exception of the VGG16 model pre-trained on PlantCLEF2015 (Supplementary Table 10), which has the correct prediction for all the images of Potato Late blight, the rest of the model configurations show major classification errors, confusing these images with Tomato Late blight, which is actually related to the same common name of disease but on different crops. This observation suggests that a common disease, although infecting different host types, produces very similar disease patterns.

6.1.1. Qualitative results

Attention map visualization. To better understand the visual features learned by the models, we used one of the visualization methods presented in (Yosinski et al., 2015), which consists of plotting the activation values of neurons in a CNN layer in response to an input image. Instead of individually visualizing the activation of a CNN layer as implemented in (Mohanty et al., 2016), we examined the highest global activation within all feature maps in a layer, so as to locate the regions most voted by the model and then exploit those that have the most impact in the final classification. More precisely, we started by plotting the position of the neuron with the highest activation for all the features maps extracted from the last convolutional layer, and from there we accumulated the first 30 dominant activations and matched them to the original image. It is possible to visualize any number of neural activations, but, based on our empirical observation, the accumulation of 30 dominant activations in this case is enough to visualize the difference in the focus of attention between different models. We chose to visualize the last convolutional layer because this layer corresponded to the highest level of abstraction and it is more specific to the target classes (Lee et al., 2017; Zeiler et al., 2011).

By comparing the visualization outputs of the GoogLeNetBN and VGG16, we found that their neuron activations were activated in different regions. For example, in Fig. 5a, both the GoogLeNetBN models pre-trained with ImageNet and PlantCLEF2015 are activated by areas of yellowish venation, while the VGG16 models are activated by areas infected with large brown spots with yellow tissues around. However, when comparing the visualization outputs of different pre-training tasks on the same network architecture, we found a certain degree of similarity in the activated areas. For example, in Fig. 5b, the GoogLeNetBN models pre-trained with ImageNet and PlantCLEF2015 concentrate mainly on the center of the leaf, while the VGG16 models mostly focus on infected secondary veins. On closer observation, we found that the activated regions do not necessarily focus on the largest infected area, but on regions with obvious leaf characteristics such as the venations.

6.1.2. Discussion

In connection to the above findings, we hereby make a few deductions. Firstly, we found that VGG achieves a better classification performance compared to other deeper network architectures which tend to overfit on the PV dataset. We believe that the lack of diversity in the PV data set, which could make the learned model sometimes misleading when used in real field data such as IPM and Bing. Secondly, it is not possible to conclude that a model pre-trained on a task closer to the final task is the best option to solve the PV dataset, as in our study we found that pre-training on plant domains improves performance in the case of deep networks, InceptionV3 and GoogLeNetBN, but not for the simpler and shallower networks such as VGG16. However, in our experiments, the generalization performances of the models were studied on IPM and Bing, which is currently the only field test set available online with verified field truth. It would be interesting to see if our results hold up on a larger number of images. If not, it might suggest that the size of the IPM and Bing is too small to generalize the results. Further investigation of the complementary field datasets will be required to reach a more robust conclusion on this point. We hope that this article will stimulate the curiosity of the scientific community in this field and facilitate further research on this topic related to the generalization of the deep model in cross domains to improve the identification of plant diseases.

Lastly, we noticed that the features learned might not necessarily look at the most obviously infected area but the region with the most distinctive leaf characteristics, such as the venation. This is believed to be due to the fact that models are trained with the terminology of crop-disease pairs, where models trained in this way could be biased towards crop-specific patterns, especially when crop features appear to be more discriminative than diseases. Therefore, in the next section, we explore an alternative approach to crop-disease modelling, attempting an
intuitive way to direct a deep model to learn feature representations based on common diseases, regardless of crop.

6.2. Disease identification based on common names of diseases and generalization to unseen crops

In this experiment we used the VGG16 model, which had been proven to be the best model in previous experiments, to experiment with the principle of learning based on common names of diseases. We compared two disease modeling strategies that are: (1) the formation of a model based on training solely on the common names of diseases (S_c), and (2) based on training on previously proposed crop-disease target classes and the subsequent implementation of a late fusion method to integrate probabilities corresponding to the individual common names of diseases (S_d). Details of these approaches can be found in section 4.

6.2.1. Evaluation on seen crops

Table 6 shows that the S_d approach is generally better than the S_c approach at predicting the PV (seen) set. The VGG16 model pre-trained with ImageNet gives the best result up to 98.98% top-1 accuracy and 98.20% average accuracy per class AVE. However, due to the nature of the PV data set, which seriously lacks diversity, both strategies were found achieving a relatively high accuracy (98% for top-1 accuracy). Through the analysis of the confusion matrices (Supplementary table 19–20) of the S_c and S_d approaches, we found that, compared to S_d, S_c has more classification errors between early blight and late blight. In fact, this misclassification occurs mainly in tomato crops where the number of disease classes is highest in PV. From this observation, we deduce that the lack of crop information may seem to affect S_c’s ability to recognize diseases with a very similar visual appearance. However, we note that although crop characteristics that could be extracted by S_d may provide additional information to help differentiate these classes, these crop characteristics do not necessarily correspond to the characteristics of plant diseases.

Next, we analyzed the performance of deep models to identify new data (IPM and Bing) from other domains, typically pictures in the field. Note that there are only two pepper crop images in IPM, and one pepper crop image in Bing, of which the number of samples is insufficient to infer the performance of deep models in recognizing the disease of an unseen crop. We hence tested the deep model using only the seen crop images. From Table 6, we can see that S_d approach in general works better than S_c on IPM and Bing. Next, using the S_d approach, in order to determine which pre-training task is the most effective, we compared the top-1 accuracy of the combined IPM and Bing images, and found

---

Fig. 5. Visualization of the attention maps of the GoogLeNetBN and VGG16 for the PV sample. The upper and lower row images are the output visualization of the model pre-trained with ImageNet and PlantCLEF2015 respectively.
Table 6: Performance of the VGG16 models on the disease classification problem based on $S_{un}$ and $S_{d}$. Note that the performance of IPM and Bing is based on only seen crop images while the unseen crop refers to Pepper_bell_Bacterial_spot and Pepper_bell_healthy.

<table>
<thead>
<tr>
<th>Pretrained dataset</th>
<th>PV (seen) (%)</th>
<th>PV (unseen) (%)</th>
<th>IPM images (%)</th>
<th>Bing images (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>top-1</td>
<td>top-3</td>
<td>AVE</td>
<td>top-1</td>
</tr>
<tr>
<td>$S_{un}$/$S_{d}$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ImageNet ($S_{un}$)</td>
<td>98.94</td>
<td>99.96</td>
<td>98.19</td>
<td>63.23</td>
</tr>
<tr>
<td>ImageNet ($S_{d}$)</td>
<td>98.96</td>
<td>99.96</td>
<td>98.20</td>
<td>34.89</td>
</tr>
<tr>
<td>PlantCLEF2015 ($S_{un}$)</td>
<td>98.42</td>
<td>99.91</td>
<td>97.31</td>
<td>65.11</td>
</tr>
<tr>
<td>PlantCLEF2015 ($S_{d}$)</td>
<td>98.52</td>
<td>99.86</td>
<td>97.70</td>
<td>37.47</td>
</tr>
</tbody>
</table>

that with ImageNet pre-training can achieve the highest top-1 classification performance, i.e. 45%, while the PlantCLEF2015 pre-training is 42.22%.

6.2.2. Generalization to unseen crops

From Table 6, we can see that, despite the fact that $S_{d}$ can achieve a better result in the PV (seen) set, $S_{un}$ generalizes better to PV (unseen) set, especially the PlantCLEF2015 model which reaches 65.11%, the highest top-1 accuracy. Note that the overall accuracy of the PV (unseen) in Table 6 was calculated from all PV (unseen) images, which means a combination of images from the Pepper_bell_Bacterial_spot and Pepper_bell_healthy classes. Next, we explored the classification performance of each individual PV (unseen) class in which, in this case, the overall accuracy is calculated according to each individual PV (unseen) class. The result of the classification is presented in Table 7.

We observed that different pre-training tasks can exert substantial influence on different categories of pepper classes. Specifically, it can be seen that, for the $S_{un}$, the ImageNet pre-trained model (83.89%) can distinguish better between healthy and unhealthy pepper leaf samples compared to that pre-trained with PlantCLEF2015 (65.40%). However, the class of pepper leaves with bacteria spots shows a contrary result where the model pre-trained with the PlantCLEF2015 (65.75%) is better than the one pre-trained with the ImageNet (43.06%).

Next, based on failure analysis of the PV (unseen), we observed that, with the ImageNet pre-trained models, a majority of Pepper_bell_Bacterial_spot are confused with target spot, Huanglongbing (Citrus greening) and early blight; with the PlantCLEF2015 pre-trained models, most of the Pepper_bell_Bacterial_spot images are confused with Haunglongbing (Citrus greening), Tomato Yellow Leaf Curl Virus and early blight. Through visual analysis, we noticed that the symptoms of those misclassified samples are very similar to the disease from the other class. For example, in Fig. 6a, the samples of Pepper_bell_Bacterial_spot that are confused with Haunglongbing (Citrus greening) have yellowish infected regions and also yellow veins, resembling the characteristics of Haunglongbing (Citrus greening). Also, those samples that are confused with the Target spot disease as shown in Fig. 6b have visually similar dark brown spots characteristics. Likewise, the healthy pepper samples are also found to be misled for the same reason. A majority of pepper healthy samples are wrongly identified as Haunglongbing (Citrus greening), most probably due to the characteristic of yellowing pepper bell leaves.

From this analysis, it is obvious that the recognition of deep models is highly affected by the symptom variations of one disease, analogous to the findings reported in (Barbedo, 2018). However, it is undeniable that, for the models formulated by both $S_{un}$ and $S_{d}$ that have never seen any pepper bell leaf samples, they are still able to capture very similar characteristics of pepper bell which could somehow help to improve the predictive modeling for unseen crops.

Lastly, by aggregating multi-class outputs into two classes, we analyzed the performance of the models to differentiate between healthy and unhealthy classes of the PV set. Note that, in this experiment, we verified the classification performance on the basis of the entire PV set, which includes the seen and unseen test set. From the ROC (Receiver Operating Characteristic) curves shown in Supplementary Fig. 2, it can be seen that the $S_{un}$ is more robust in differentiating between healthy and unhealthy leaf samples compared to the model trained with $S_{d}$. Indeed, through the AUC (Area Under the Curve) as shown in Table 8, $S_{un}$ as a whole performed better than $S_{d}$.

Table 7: Results of the individual classification of diseases related to the pepper bell crop for the PV (unseen). It presents the top-1 accuracy of the Pepper_bell_Bacterial_spot and Pepper_bell_healthy for the $S_{un}$ and $S_{d}$ with different pre-training tasks.

<table>
<thead>
<tr>
<th>Pepper class</th>
<th>ImageNet</th>
<th>PlantCLEF2015</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$S_{un}$</td>
<td>$S_{d}$</td>
</tr>
<tr>
<td>bacteria spot</td>
<td>43.06</td>
<td>18.98</td>
</tr>
<tr>
<td>healthy</td>
<td>83.89</td>
<td>51.18</td>
</tr>
</tbody>
</table>

Table 8: Performance of the VGG16 models on the healthy and unhealthy classification of the PV set.

<table>
<thead>
<tr>
<th>Pretraining task</th>
<th>Classifier</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImageNet</td>
<td>$S_{un}$</td>
<td>0.972</td>
</tr>
<tr>
<td></td>
<td>$S_{d}$</td>
<td>0.744</td>
</tr>
<tr>
<td>PlantCLEF2015</td>
<td>$S_{un}$</td>
<td>0.967</td>
</tr>
<tr>
<td></td>
<td>$S_{d}$</td>
<td>0.740</td>
</tr>
</tbody>
</table>
massive volumes of visual data that can’t be manually exhaustively analyzed. In such cases, an early detection of unhealthy leaf samples can contribute to quickly investigate their origins, and potentially implement appropriate solutions.

7. Conclusion

First, this paper has examined and compared the performance of different transfer learning mechanisms based on different (1) pre-training tasks: the plant specialized domain and general object domain, and the (2) network architectures: VGG16 (16 layers), GoogLeNetBN (34 layers) and InceptionV3 (48 layers). We have experimentally proven that pre-training with plant specialized tasks could reduce the impact of overfitting for the deeper Inception-based model but the VGG16 model with ImageNet pre-training has shown a better generalization in adapting to new data.

Second, the fact that the VGG16 perform better than the inception-based models would be due to the lack of diversity in the PV data set, which leads to a constraint in the deployment of deeper architectures. Nevertheless, with regard to the fact that ImageNet pre-training does not necessarily regularize or improve the accuracy of final target tasks, as indicated in (He et al., 1811), we suggest that the agricultural community move forward to create a broader and more diversified plant disease database, without having to use a pre-training model. In addition, for the model to better adapt to large-scale crops, a dataset of images captured under real cultivation conditions (Barbedo, 2018; Ferentinos, 2018) is needed, as demonstrated by (Ferentinos, 2018).

Third, by interpreting the underlying learned characteristics through the visualization of activations, we have showed that a CNN trained in this crop-disease terminology may not always focus on disease regions but crop-specific characteristics such as leaf venation or lamination to facilitate data discrimination. Therefore, in order to capture the visual symptoms of plant diseases, we demonstrated an intuitive way of leading a deep model to learn the representation of characteristics based on the common names of diseases instead of target classes of crop-disease pairs. We have shown experimentally that the model trained with common diseases, regardless of crop is more generalizable, especially for new data taken in different domains and also for unseen crops.

CRediT authorship contribution statement

Sue Han Lee: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Visualization, Resources, Data curation, Writing - original draft. Hervé Goëau: Formal analysis, Investigation, Conceptualization, Methodology, Writing - review & editing, Visualization. Pierre Bonnet: Formal analysis, Investigation, Supervision, Project administration, Funding acquisition, Resources, Data curation, Writing - review & editing. Alexis Joly: Formal analysis, Investigation, Conceptualization, Methodology, Supervision, Project administration, Funding acquisition, Writing - review & editing.

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.

Acknowledgement

This project is supported by Agropolis Fondation, Numev, Cemeb, #DigitAG, under the reference ID 1604-019 through the < < Investissements davenir > > programme (Labex Agro:ANR-10-LABX-0001-01).

Appendix A. Supplementary material

Supplementary data to this article can be found online at https://doi.org/10.1016/j.compag.2020.105220.

References


