Wheat Diseases Classification and Localization Using Convolutional Neural Networks and GradCAM Visualization

Elias Ennadifi
Information, Signal and Artificial Intelligence Laboratory
Polytech Mons, University of Mons
Mons, Belgium
elias.ennadifi@umons.ac.be

Sohaib Laraba
Information, Signal and Artificial Intelligence Laboratory
Polytech Mons, University of Mons
Mons, Belgium
sohaib.laraba@umons.ac.be

Damien Vincke
Walloon Agricultural Research Centre
Knowledge and Valorization of Agricultural Products Department
Quality and Authentication of Agricultural Products
Chaussée de Namur, 24, 5030, Gembloux, Belgium
d.vincke@cra.wallonie.be

Benoît Mercatoris
Biosystems Dynamics and Exchanges
TERRA Teaching and Research Centre
Gembloux Agro-Bio Tech, University of Liège
Passage des Déportés, 2, 5030, Gembloux, Belgium
benoit.mercatoris@uliege.be

Bernard Gosselin
Information, Signal and Artificial Intelligence Laboratory
Polytech Mons, University of Mons
Mons, Belgium
bernard.gosselin@umons.ac.be

***Abstract*—The world has been witnessing a population boom that has several implications including food security. Wheat is one of the world's most important crops in terms of production and consumption, and demand for it is increasing. On the other hand, diseases can damage the abundance and the quality of the crop, so this needs to be revealed through advanced methods. In recent years, along with the various technological developments, using Convolutional Neural Networks (CNN) has proved to be showing great results in many image classification tasks. However, deep learning models are generally considered as black boxes and it is difficult to understand what the model has learned. The purpose of this article is to detect diseases from wheat images using CNNs and to use visualization methods to understand what these models have learned. For this reason, a wheat database has been collected by CRA-W (Walloon Agricultural Research Center), which contains 1163 images and is classified into two groups namely sick and healthy. Moreover, we propose to use the mask R-CNN for segmentation and extraction of wheat spikes from the background. Furthermore, a visualization and interpretation method, namely Gradient-weighted Class Activation Mapping (GradCAM), is used to locate the disease on the wheat spikes in a non-supervised way. GradCAM is actually used generally to highlight the most important regions from the CNN model's viewpoint that are used to perform the classification.**

This project is funded by Wallonie agriculture SPW.

978-1-7281-8041-0/20/$31.00

Keywords: Convolutional Neural Networks, GradCAM, Mask R-CNN, image processing, classification, deep learning, plant diseases detection

I. Introduction

Recently, the demand for grain has increased exceptionally. In the meantime, seasonal fluctuations, extreme weather events and climate change in various parts of the world increase the risk of plant pests and diseases so that plant diseases can cause severe damage to crops by significantly reducing production. A research from the Food and Agriculture Organization of the United Nations (FAO) showed that pests and diseases lead to the loss of 20-40% of global food production. This represents a threat to food security (Food and Agriculture Organization of the United Nation, International Plant Protection Convention, 2017) [1]. This triggers off the urge for fighting diseases and protecting plants, and particularly wheat, to ensure the quality and quantity of crops needed for consumption.

To prevent the spreading of these diseases, an early and effective detection plan necessary for a successful protection strategy in order to choose the appropriate treatment, and at the right time. Generally, the detection of plant diseases is achieved by experts with an academic background and a solid practical experience on symptoms and causes of diseases.

Moreover, these experts must surveil the plants regularly and directly to specify the affected areas. This continuous monitoring represents a tedious and time consuming task for humans. The automation of such a task can be of very helpful to limit the spreading of diseases and should prioritized over the human factor.

Detecting plant diseases through an image is, nonetheless, a very hard task [1]. Crops are complex and complex environments with a constant evolution throughout the season. In recent years, with development of deep learning, and particularly Convolutional Neural Networks (CNNs), image classification task has rapidly advanced in many fields including agriculture and plant diseases detection. Classification accuracy using such models has even surpassed that of humans in some cases [2]. Despite the good outcomes of deep learning classifiers, they are often considered as black boxes where the classification results are very satisfying but without any further explanations about the classification process. In fact, high accuracy alone is not enough for plant disease classification [3]. Users are also interested in other information such as how the classification is realized and which symptoms are shown. This knowledge is very important from a practical point of view [3].

To implement this process on our dataset, several steps must be followed:

Step 1: Preprocessing
Step 2: Segmentation
Step 3: Classification using state-of-the-art (CNN)
Step 4: understand the trained models
Step 5: put a bounding box on the sick part of wheat sick

Images of the collected dataset contains two wheat spikes on a black background. These images contain also the camera flash that we need to deal with. Examples of the collected images and conditions are shown in Fig. 1. After processing the data, the first step in the object detection process is to create another database to be able to put a single wheat in a single image. In our new dataset, the first class contains the "Sick" images and the second class englobes the "Healthy" images (Fig. 2) using mask R-CNN [4]. We have tested multiple state-of-the-art CNN architectures on our new database. Then, to choose a model, we measured accuracy to evaluate how strong our classification system is. After choosing a model we applied the GradCAM visualization technique [5], which aims to make CNN-based models more transparent by visualizing the most important regions on the input images for the prediction of corresponding classes.

In the next sections, we will describe our proposed approach to classify and detect sick wheat. Then, we will show different experiments on the proposed dataset.

II. RELATED WORKS

Deep Learning-based approaches require a huge amount of data in order to obtain satisfying results, and generally, using small datasets lead to overfitting. However, collecting huge datasets is a tedious and very costly task because it is done manually.

Fig. 1. Example of images from the collected dataset.

Fig. 2. Example of images from the generated dataset after segmentation.

Most of available research works on plant diseases detection are done on public datasets. 11 out of 19 studies realized by Boulent et al. [1] are performed on public datasets. The most known and widely used dataset for this task is called "PlantVillage" which contains today 87,848 images of healthy and diseased crop plants (Hughes and Salathé, 2015) (Ferentinos, 2018). Mohanty et al. [6] for example were one of the first authors to apply Convolutional Neural Networks for plant diseases detection using the PlantVillage dataset. Authors tested two standard architectures, namely AlexNet [7] and GoogleNet, and studied the effect of transfer learning for classification. Sladojevic et al. [8] proposed to automatically classify and detect plant diseases from leaf images. They have first gathered data from internet and formed a dataset of thirteen different classes corresponding to different diseases. Two additional classes were added for more accurate classification. The first one corresponds to healthy leaves and second one corresponds to background images taken from the Stanford background dataset [9]. In total, 3000 images were gathered and extended using data augmentation techniques to about 30000 images. CaffeNet CNN architecture [10] was used with the process of finetuning to achieve classification. Brahimi et al. [3] Used also CNNs to classify tomato diseases from leaf images. A dataset of almost 15000 images of leaves was extracted from the PlantVillage dataset and was divided into nine different classes corresponding to nine diseases. Another extended research by these authors compared multiple state-of-the-art CNN architecture with different training strategies for plant diseases classification [3]. They have formed a dataset of more than 55000 images by merging the PlantVillage with the Stanford background dataset.
However, despite all the advances and fruitfulness results of deep learning classifiers, they are still considered as black boxes because of the lack of their interpretability.
In order to "unbox" the learning process and make it more

transparent, several visualization and debugging approaches have been proposed. These approaches allow to interpret and picture what is happening in the deep learning model. Atabay H.A. [11] for example proposed to use the occlusion technique. This techniques consists of sliding a black windows on the input image and analyze the change of the output probability. This method allows to generate a heatmap that highlights the pixels that are most sensitive to a specific class. It also allowed to detect the biases in the data where, in some cases, the assigned class depends on pixels on the background, which means that the features learned are not totally dependent on the object of interest [1].

Brahimi et al. [3] have also used the occlusion technique in order to location diseases on tomato leaves and compared it with another technique of saliency maps visualization. They have concluded that the occlusion technique was very time-consuming and computationally expensive. The saliency maps were computed based on gradient values and allow to estimate the importance of pixels in the input image in the node corresponding to the ground truth. The authors followed two strategies and tested this method with and without guided backpropagation. The guided backpropagation approach propagates only positive gradients through the activation functions. This allows to obtain more precise visualizations. Saliency maps are computed in only one backpropagation pass, which makes this method very fast compared to the occlusion technique.

The insight brought by such visualization methods can help us understand the behavior of trained models in order to bring more improvements. It can also make deep learning models more reliable where the user can obtain more transparency.

Visualization techniques are also recommended in the medical image analysis that requires a lot of precision and transparency to ensure the correct diagnosis [1] [12].

III. THE PROPOSED APPROACH

In this part, an approach to locate the disease on wheat is presented in an unsupervised manner, to provide information that may help peasants in their diagnosis. Fig. 3 explains the given approach, which involves the following steps:

Step 1: Pre-processing
Step 2: Segmentation
Step 3: Classification
Step 4: visualisation method
Step 5: localisation

A. Segmentation

Among the problems in these RGB images are the camera flash and sometimes the use of a yellow background which makes it hard to segment these images. For our task, we use Mask R-CNN segmentation method [4] in order to extract wheat spikes from the background. We have first tested the direct use of mask R-CNN, trained on the MSCOCO object detection dataset [13] but the segmentation results were not satisfying. After several empirical results, we found that only

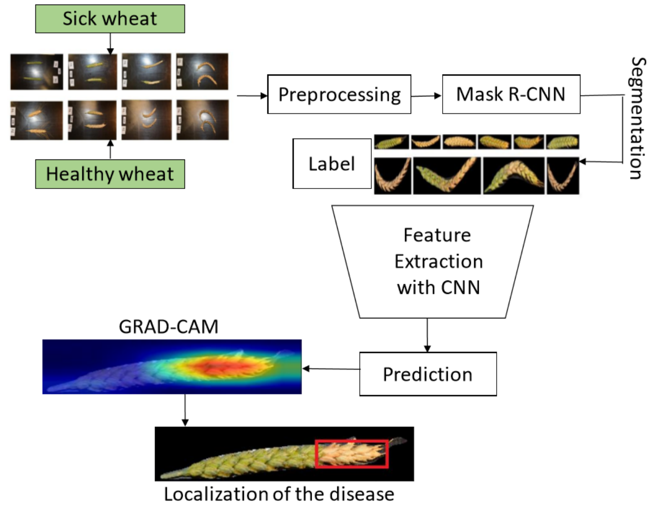

Fig. 3. Flowchart of the proposed approach.

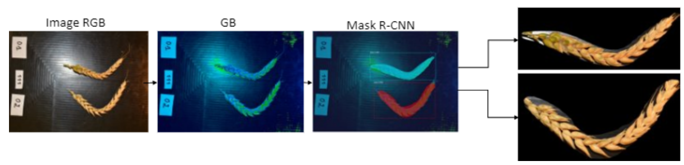

Fig. 4. Preprocessing and segmentation of wheat spikes.

removing the red channel gives good results and the segmentation as very smooth (as shown in Fig. 4). We note that we only need to benefit from the knowledge learn by the mask R-CNN of object segmentation and we do not re-train the model from scratch. After this transformation, we were able to segment more than 85% of our image database. We use a pre-trained Mask R-CNN model to detect wheat on our database. We used an implementation of Mask R-CNN on Python 3, Keras, and TensorFlow. The model generates bounding boxes and segmentation masks for each instance of an object in the image. It is based on the Feature Pyramid Network (FPN) [14] and a ResNet101 [2] backbone.

B. Classification

In recent years, the use Convolutional Researchers found themselves faced with a great opportunity to improve the accuracy of classification in all areas, including agriculture and the detection of plant diseases. Unlike classical machine learning techniques, convolutional neural networks automatically learn the characteristics of each image. The CNNs themselves do the tedious job of extracting and describing features: during the training phase, the classification error is minimized to optimize the classifier parameters and features, but these networks need a large database for training. As the collection of input images is small, it is strongly advised not to train the neural network from scratch: the number of parameters to be learned being much greater than the number of images, the risk of overfitting is enormous. Transfer Learning is a technique that is widely used in practice. It requires having a neural network that is already trained, preferably on a problem close to the one we want to solve, in our case, we will use the PlantVillage database. Besides, we replace the last fully-connected layer of the pre-trained network with a classifier adapted to the new problem and initialized randomly. The convolutional neural network chosen for this task is Densenet121 [15], this choice is based on a comparative study with other CNN models. Ultimately, we achieved 93.47% of accuracy.

C. Visualization method

Despite the good results of DL classifiers and those in many tasks of image classification including the classification of agricultural images and detection of plant diseases, one of the downsides associated with deep learning methods is the difficulty in understanding what the model has learned. Nevertheless, it is possible to understand the model's inner mechanism or at least get a glimpse of it through several techniques. By better understanding the trained models, we can not only ensure the relevance of generated results, but also improve their quality [1].
This type of method is very crucial from the practical view-point because it projects the features used by the network back to the input image. Therefore, the image can be examined to understand how the classifier behaves [2, 14, 15]. In our task of diseases detection, these visualizations can highlight the most important regions used by the network as features in order to perform the classification. If the classifier is correctly trained, these parts may represent the symptoms present on the input images. If any bias is present in the data that affects the training process, these visualization techniques allows as to debug the model and visualize these biases [16].
In this work, we use the GradCAM method [5] which allows to make CNN-based models more transparent by visualizing the input regions that are important for the predictions of these models. GradCAM uses the gradient information flowing into the last convolutional layer of the CNN to understand the importance of each neuron for a decision of interest [5].

IV. EXPERIMENTATION

The results of Convolutional Neural Network (CNN) and GradCAM are described in this part. The suggested approach was implemented and tested in a wheat database collected by CRA-W [17]. A total of 1163 images were collected over the entire 2019 season (in ”lab” configuration), they are classified into two groups, sick and healthy. All of these acquisitions are summarized in the table below (Table I).

TABLE I
METADATA OF THE COLLECTED DATA.

Harvest	Measured	Number	Scanned organs
11/06/2019	12/06/19 and 13/06/19	160	2 spikes per sample
24/06/2019	25/06/2019	160	2 spikes per sample
01/07/2019	02/07/2019	161	2 spikes per sample
09/07/2019	10/07/2019	162	2 spikes per sample
16/07/2019	17/07/2019	162	2 spikes per sample
22/07/2019	23/07/2019	162	2 spikes per sample
31/07/2019	01/08/2019	162	2 spikes per sample

The dataset is divided into 80% for training and 20% for evaluation. In the following, we elaborate a comparative study in terms of accuracy for different classification models (Table II), namely: Inceptionv3 [18], VGG19 [19], ResNet50 [2], ResNet50V2 [20], Xception [21], MobileNet [22], DenseNet121 [15] and DenseNet169 [15]. All these 8 training configurations use the values of the same hyperparameters (momentum 0.9, weight decay 0.0006, learning rate 0.001, batch sizes 10). Moreover, the last fully connected layer of the pre-trained network is replaced by a new layer compatible with the number of classes in our dataset and initialized randomly. All layers are then trained on the new images.

The results presented in the table 2 show that Densnet121 model gives the highest accuracy and is better than the other models. We choose this model for the next experiments. We calculate the sensitivity (TP rate) and specificity (TN rate) measures for the model chosen following these equations.

$$Sensitivity = \frac{TP}{TP + FN}$$

$$Specificity = \frac{TP}{FP + TN}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Where:
TP : number of true positives
TN : number of true negatives
FP : number of false positives
FN : number of false negatives

To analyze the proposed CNN models, we have used visualization methods to understand symptoms and to locate disease regions in wheat. More particularly, we used GradCAM to understand what these models learned, the DenseNet121 and DenseNet201 models have generally shown their ability to understand wheat images compared to the

TABLE II
EXPERIMENTAL RESULTS.

Model	Accuracy (%)	Training time (s)
DenseNet121	93.47	7013
DenseNet201	90.75	10676
ResNet152	87.3	8479
ResNet101V2	88.11	5869
Xception	91.16	9520
Mobilenet	88.11	1885
InceptionV3	90.62	6620
VGG19	65.57	5540

TABLE III
RESULTS PF SENSITIVITY AND SPECIFICITY ON THE CHOSEN MODEL.

Model	Sensitivity	Specificity
DenseNet121	92.8%	94.1%

other models. After selecting the DenseNet121 model, we applied GradCAM to highlight the areas that our model used to predict as shown in Fig. 5 - Middle. Based on a visual analysis of important areas, it should be noted that the model generally uses symptoms to classify diseased wheat, and no biasis are used. Finally, we add a frame on this important region for a practical view (Fig. 5 - Right).

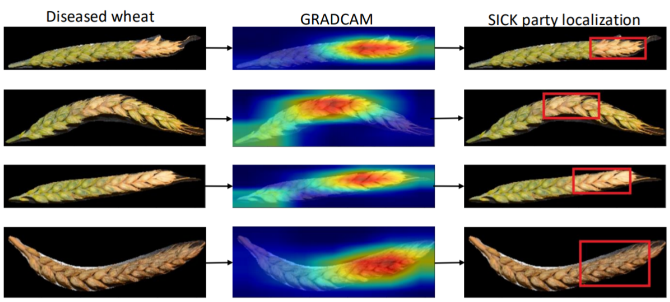

Fig. 5. Localization of symptoms using GradCAM visualization method. Left: input images. Middle: GradCAM visualization. Right: Boxes applied on the sick regions of wheat spikes.

V. CONCLUSION

Our goal in this paper is to find a model that can classify wheat into diseased and healthy categories. Therefore, we have tested multiple state-of-the-art Convolutional Neural Network (CNN) and we achieved an accuracy of 93% with DensNet121. To check the reliability of our model and that it correctly understood our database, we used visualization methods. In this context, we used the GradCAM method to get the important regions that DenseNet121 model used to predict the category of our input image.

Although our database is limited, and diseased wheat areas are not available, our model has generally demonstrated its ability to recognize these infected areas. Furthermore, the use of the visualization method shows a precise and sharp visualization which helps the inexperienced users to locate the diseases. As future work, we will focus on the preparation of a strongly labeled dataset, which makes it possible to measure numerically the performance of GradCAM in wheat diseases, we will start collecting in partnership with CRA-W in 2020 season.

REFERENCES

[1] J. Boulent, S. Foucher, J. Théau, and P.-L. St-Charles, "Convolutional neural networks for the automatic identification of plant diseases", *Frontiers in plant science*, vol. 10, 2019.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[3] M. Brahimi, M. Arsenovic, S. Laraba, S. Sladojevic, K. Boukhalfa, and A. Moussaoui, "Deep learning for plant diseases: detection and saliency map visualisation", in *Human and Machine Learning*. Springer, 2018, pp. 93–117.

[4] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn", in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[5] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization", in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.

[6] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection", *Frontiers in plant science*, vol. 7, p. 1419, 2016.

[7] S.-I. e. H. G. E. KRIZHEVSKY, Alex, "Imagenet classification with deep convolutional neural networks", *Advances in neural information processing systems*, pp. 1097–1105, 2017.

[8] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep neural networks based recognition of plant diseases by leaf image classification", *Computational intelligence and neuroscience*, vol. 2016, 2016.

[9] S. Gould, R. Fulton, and D. Koller, "Decomposing a scene into geometric and semantically consistent regions", in *2009 IEEE 12th international conference on computer vision*. IEEE, 2009, pp. 1–8.

[10] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding", in *Proceedings of the 22nd ACM international conference on Multimedia*, 2014, pp. 675–678.

[11] H. A. Atabay, "Deep residual learning for tomato plant leaf disease identification." *Journal of Theoretical & Applied Information Technology*, vol. 95, no. 24, 2017.

[12] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. Van Der Laak, B. Van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis", *Medical image analysis*, vol. 42, pp. 60–88, 2017.

[13] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context", in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[14] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[15] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[16] M. Brahimi, K. Boukhalfa, and A. Moussaoui, "Deep learning for tomato diseases: classification and symptoms visualization", *Applied Artificial Intelligence*, vol. 31, no. 4, pp. 299–315, 2017.

[17] CRA-W, *Walloon Agricultural Research Center, Belgium*, 2018 (accessed May 2020), http://www.cra.wallonie.be/fr.

[18] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[19] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", *arXiv preprint arXiv:1409.1556*, 2014.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks", in *European conference on computer vision*. Springer, 2016, pp. 630–645.

[21] F. Chollet, "Xception: Deep learning with depthwise separable convolutions", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.

[22] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications", *arXiv preprint arXiv:1704.04861*, 2017.