


RESEARCH

Open Access



AI-powered banana diseases and pest detection

Michael Gomez Selvaraj^{1*} , Alejandro Vergara¹, Henry Ruiz², Nancy Safari³, Sivalingam Elayabalan⁴, Walter Ocimati⁵ and Guy Blomme⁶

Abstract

Background: Banana (*Musa* spp.) is the most popular marketable fruit crop grown all over the world, and a dominant staple food in many developing countries. Worldwide, banana production is affected by numerous diseases and pests. Novel and rapid methods for the timely detection of pests and diseases will allow to surveil and develop control measures with greater efficiency. As deep convolutional neural networks (DCNN) and transfer learning has been successfully applied in various fields, it has freshly moved in the domain of just-in-time crop disease detection. The aim of this research is to develop an AI-based banana disease and pest detection system using a DCNN to support banana farmers.

Results: Large datasets of expert pre-screened banana disease and pest symptom/damage images were collected from various hotspots in Africa and Southern India. To build a detection model, we retrained three different convolutional neural network (CNN) architectures using a transfer learning approach. A total of six different models were developed from 18 different classes (disease by plant parts) using images collected from different parts of the banana plant. Our studies revealed ResNet50 and InceptionV2 based models performed better compared to MobileNetV1. These architectures represent the state-of-the-art results of banana diseases and pest detection with an accuracy of more than 90% in most of the models tested. These experimental results were comparable with other state-of-the-art models found in the literature. With a future view to run these detection capabilities on a mobile device, we evaluated the performance of SSD (single shot detector) MobileNetV1. Performance and validation metrics were also computed to measure the accuracy of different models in automated disease detection methods.

Conclusion: Our results showed that the DCNN was a robust and easily deployable strategy for digital banana disease and pest detection. Using a pre-trained disease recognition model, we were able to perform deep transfer learning (DTL) to produce a network that can make accurate predictions. This significant high success rate makes the model a useful early disease and pest detection tool, and this research could be further extended to develop a fully automated mobile app to help millions of banana farmers in developing countries.

Keywords: Artificial intelligence, Banana, Deep learning, Disease detection, Transfer learning, Convolutional neural networks, Mobile app

*Correspondence: m.selvaraj@cgjar.org

¹ International Center for Tropical Agriculture (CIAT), A.A. 6713, Cali, Colombia

Full list of author information is available at the end of the article



Background

Bananas (*Musa* spp.) are one of the world's most important fruit crops in terms of production volume and trade [1]. Though a major staple food in Africa, Asia, and Latin America, only 13% of bananas produced are globally traded [2], clearly indicating the fruit's importance in domestic markets and food security. In East and Central Africa, it is a substantial dietary component, accounting for over 50% of daily total food intake in parts of Uganda and Rwanda [3]. Smallholder farmers, representing 85% of the world's farms [4], face many abiotic and biotic constraints. Several banana pests and diseases have caused significant yield losses across production landscapes [5] and are a significant threat to global food security. Therefore, early detection of pests and diseases in the field is a first crucial step. Traditional pest and disease identification approaches rely on agricultural extension specialists, but these approaches are limited in developing countries with low human infrastructure capacity. Many smallholder farmers rely on empirical knowledge, which is less effective in overcoming farming challenges [6]. The early identification of a crop disease or pest can lead to faster interventions with resulting reduced impacts on food supply chains.

Artificial intelligence (AI) with deep learning models which help to identify plant diseases by the plant's appearance and visual symptoms that mimic human behavior should be considered [7]. Smartphone-based AI apps could alert farmers and expedite disease diagnosis, thus preventing the possible outbreak of pests and diseases [8]. Even though many farmers of developing countries do not have access to these advanced tools, internet infiltration and smartphone penetration offer new outfits for in-field crop disease detection. The Global System for Mobile Association (GMSA) predicted that global smartphone subscriptions would reach 5 billion by 2020, of which nearly one billion in Africa [9]. We do believe that cutting-edge technologies like AI, IoT (Internet of Things), robotics, satellites, cloud computing, and machine learning are transfiguring agriculture and helping farmers foresee their near future.

Deep learning is a novel method for image processing and object detection with greater accuracy in the classification of various crop diseases [10]. Transfer learning is one such popular approach in deep learning, where pre-trained models are adapted to do a new task. Deep transfer learning (DTL) generates a fresh framework for digital image processing and predictive analytics, with greater accuracy and has huge potential in crop disease detection. DTL approach also offers a promising avenue for in-field disease recognition using large trained image datasets and bids a shortcut to the developed models to meet the restrictions that are offered by mobile

application [11]. This would have a distinct practical value for real field environment.

Earlier investigations have validated AI-based recognition of crop diseases in wheat [12], cassava [11] and on datasets of healthy and diseased plants [8, 13]. Crop disease recognition based on a computerized image system through feature extraction has revealed promising results [14] but extracting features is computationally rigorous and involves expert knowledge for robust depiction. Only few restricted large, curated image datasets of crop disease library exists [10]. The PlantVillage platform holds over 50,000 images of different crops and diseases [15]. However, most of these images were taken with detached leaves on a plain background, and CNN trained on these images did not achieve well when using real field images [8]. To build robust and more practical detection models, plenty of healthy and diseased images taken from different infected parts of the plants, and growing under different environmental conditions are needed. These images subsequently need to be labeled and pre-screened by plant pathology experts. So far, existing crop disease detection models are mostly focusing on leaf symptoms. Unfortunately, numerous symptoms also appear in other parts of the plant and the best examples are banana pest and disease linked symptoms.

The objective of this study was to apply state-of-the-art deep learning techniques for the detection of visible banana disease and pest symptoms on different parts of the banana plant. We also considered the potential for adapting pre-trained deep learning CNN models to detect banana disease and pest symptoms using a large dataset of experts' pre-screened real field images collected from Africa and India.

Materials and methods

System description

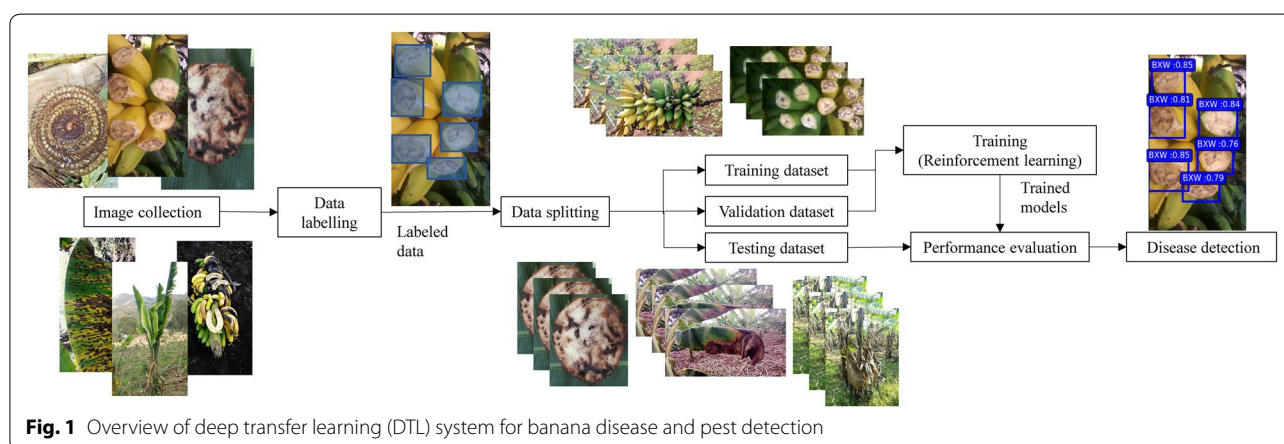
Our DTL system dataset consists of five major banana diseases along with their respective healthy classes; dried/old age leaves and banana corm weevil (*Cosmopolites sordidus*) damage symptom classes (Table 1). Since these major diseases and pest can affect different parts of the banana plant, we ended up with six different models (entire plant, leaves, pseudostem, fruit bunch, cut fruits and corm) and 18 different classes (Table 1) to achieve maximum accuracy. An overview of the DTL system is illustrated in Fig. 1.

Dataset collection

Our dataset comprises of about 18,000 field images of banana, collected by banana experts, from Bioversity International (Africa) and Tamil Nadu Agricultural University (TNAU, Southern India) (Additional file 2: Table S1). These field images were captured under

Table 1 Description of annotated banana datasets used in this study

Model/classes	Entire plant	Leaf	Pseudostem	Fruit bunch	Cut fruit	Corm	Total
Healthy plant (HP)	948	2583	702	716	8180	702	13,831
Xanthomonas wilt (BXW)	731	842	704	711	6196		9184
Bunchy top disease (BBTD)	902						902
Black sigatoka (BS)		980					980
Yellow sigatoka (YS)		1066					1066
Fusarium wilt (FWB)	759	967					1726
Dried/old leaves (DOL)		2562					2562
Corm weevil (BCW)						701	701
TOTAL	3340	9000	1406	1427	14,376	1403	30,952



different environmental conditions to build a robust model. For that purpose, various banana experts visited several banana farms located in disease/pest hotspots of Africa (Eastern Democratic Republic of Congo, Central Uganda, Burundi and Benin Republic) and Southern India (Tamil Nadu and Kerala). Our current dataset consists of various types of data, including images with various resolutions (cell phone, tablets, standard RGB camera); light conditions depending on time of image taking (e.g., illumination), season (e.g., temperature, humidity), and different environmental locations (e.g., Africa, India). We have collected the images at different growing phases of the crop (i.e., vegetative and reproductive). To prevent our model from being confused between dried/old leaves and diseased leaves, we also collected numerous images of dried and old age leaves at different plant growth stages. Images of a specific disease were collected from different varieties, at different plant growth stages and in different environments (Africa and India) in order to enrich the image library (Additional file 2: Table S1).

Our current CIAT banana image library consists of approximately 18,000 real field images. But in this present

study, our datasets cover healthy plants (HP), dried/old age leaves (DOL) and a balanced number of images (700 images) from five major diseases such as, Xanthomonas wilt of banana (BXW), Fusarium wilt of banana (FWB), black sigatoka (BS), yellow sigatoka (YS) and banana bunchy top disease (BBTV) along with the banana corm weevil (BCW) pest class. The major pest (corm weevil) and disease class symptoms and their control measures are presented in Additional file 2: Table S2. Since symptoms of different diseases and pests are seen at different parts of the banana plants, we captured images of all the plant parts (Fig. 2). Our current library was structured based on the disease and the affected plant parts so each part of the plant represents a model.

Data labeling

The image tagging process was done using LabelImg software [16]. Labels and coordinates of the boxes were saved as an XML file, in the same format (PASCAL VOC) used by ImageNet [17]. The number of annotated samples corresponded to the number of bounding boxes labeled in each image. Every image could contain more than one

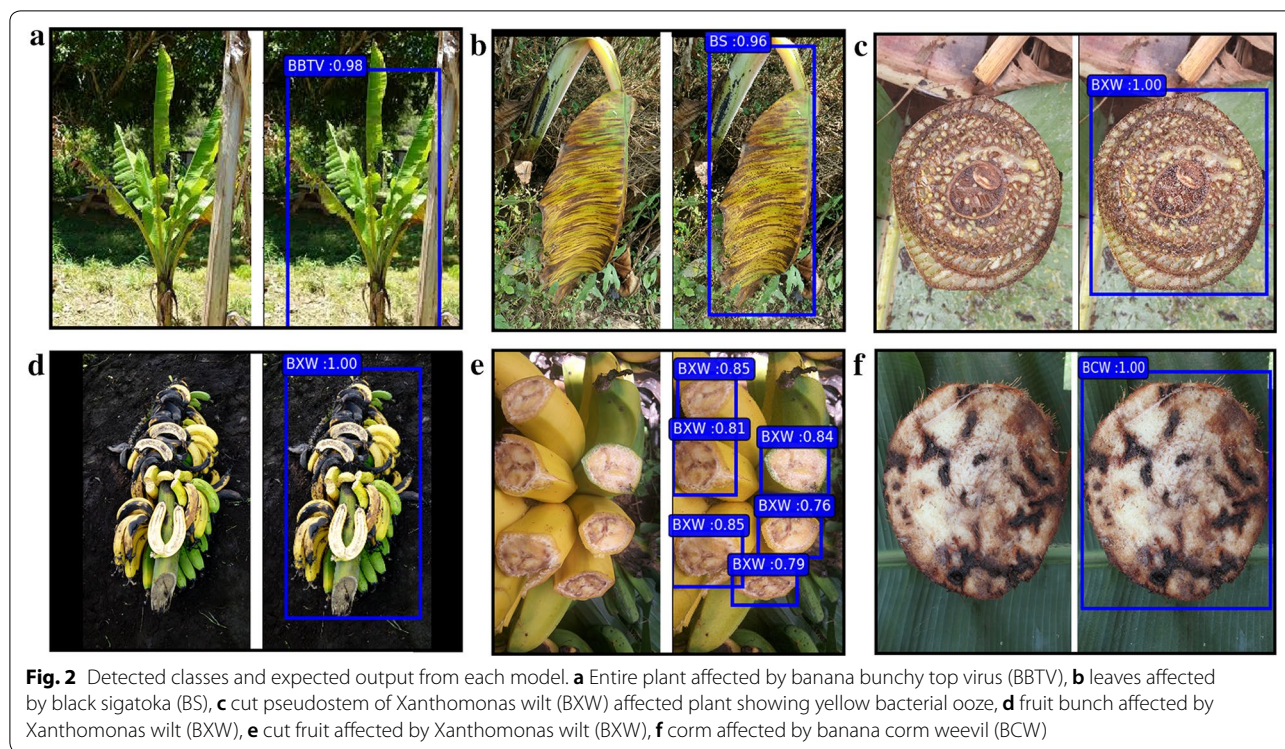


Fig. 2 Detected classes and expected output from each model. **a** Entire plant affected by banana bunchy top virus (BBTV), **b** leaves affected by black sigatoka (BS), **c** cut pseudostem of *Xanthomonas wilt* (BXW) affected plant showing yellow bacterial ooze, **d** fruit bunch affected by *Xanthomonas wilt* (BXW), **e** cut fruit affected by *Xanthomonas wilt* (BXW), **f** corm affected by banana corm weevil (BCW)

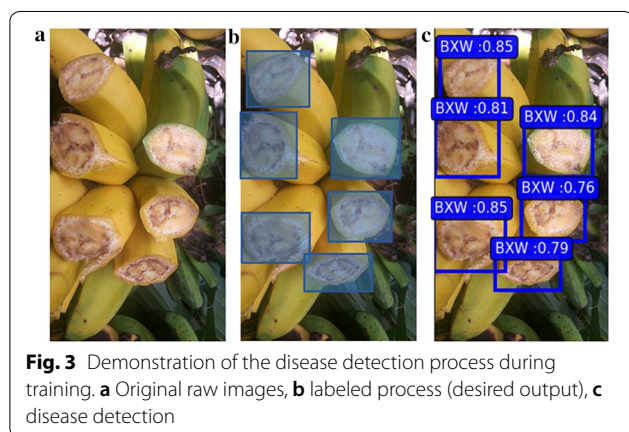


Fig. 3 Demonstration of the disease detection process during training. **a** Original raw images, **b** labeled process (desired output), **c** disease detection

annotation depending on the number of infected areas of the plant parts (Fig. 3).

CNN architectures

To train the models, we used three different architectures, such as ResNet50 [18], InceptionV2 [19] and MobileNetV1 [20]. For the object detector model architecture, we chose Faster RCNN with ResNet50 and InceptionV2 due to their accuracy. Single Shot Multi-box (SSD) model was selected with the MobileNetV1 since this was one of the fastest object detection models available in TensorFlow [21]. To train these models,

we used a python deep learning library called TensorFlow and its object detection Application Programming Interface (API) with the Graphics Process Unit (GPU) version [22]. Pre-trained models were trained with COCO (Common objects in context) data set [23], and it is openly available in the TensorFlow object detection API zoo models. These three architectures were re-trained using the transfer learning approach from the pre-trained versions. To finetune the original hyper-parameters, the following configuration changes were executed, batch size and epoch number. The batch size was changed only in the MobileNetV1 from 24 to 6, and the epoch number was kept 15,000 for all the architectures trained.

Training

One of the most challenging tasks in machine learning is splitting the data without suffering from overfitting, under fitting or generalization hitches. Nevertheless, there are several refined statistical sampling methods which provide a path to deal with these common disputes [24]. For developing banana model, our dataset was divided into the following proportions of 70%, 20%, 10%, for training (Ttr), validation (T_v) and testing (T_t), respectively. The simple random sampling (SRS) technique was selected, considering that it is efficient and simple to implement [24].

Performance metrics

Loss function

Classification loss is used to measure the model’s confidence by classifying the pixels region delimited by the bounding box [25] and the localization loss measures the geometric distance between the predicted bounding box and the ground truth annotation (validation bounding boxes). In this paper, we used the object detection API [26] to estimate the total loss function to measure model performance. The overall loss function or total loss was a weighted combination of the classification loss (classif) and the localization loss (loc).

MaP score

The mean average precision (mAP) was used as the validation metric for banana disease and pest detection. Precision refers to the accuracy. mAP score was calculated as follows: Average across the number of classes of the true positive divided by the true positives plus false positive as in the following equation

$$mAP = \frac{1}{\#classes} \sum_1^{\#classes} \frac{\#TP}{\#TP + \#FP}$$

Confusion matrix

In addition to mAP score, we also computed a confusion matrix (CM) for each selected model based on the object detection script [27]. Computation of CM protocol is described below. For each detection, the

algorithm mines all the ground-truth boxes and classes, along with the detected boxes, classes, and scores of Intersection over Union (IoU). Only detections with a score ≥ 0.5 were considered and anything under this threshold were excluded. For each ground-truth box, the algorithm creates the IoU with each detected box. A match was found if both boxes had an $\text{IoU} \geq 0.5$. The list of matches was trimmed to remove duplicates (ground-truth boxes that match with more than one detection box or vice versa). If there are duplicates, the best match (greater IoU) was continually selected. The CM was updated to reflect the resultant matches between ground-truth and detections. A detected box was reflected as correct where the intersection over union (IoU) of that box and the corresponding ground-truth box was ≥ 0.5 . The formula for calculating IoU is shown in Fig. 4. In the final step, the CM was normalized.

Software and hardware system

The list of hardware and software used in this study was depicted in Table 2. For algorithm implementation, and data wrangling scripts, python 3.6 was used. Then models were re-trained using the powerful library called TensorFlow object detection API [28] developed by Google, this library support control process unit (CPU) and GPU training and inference.

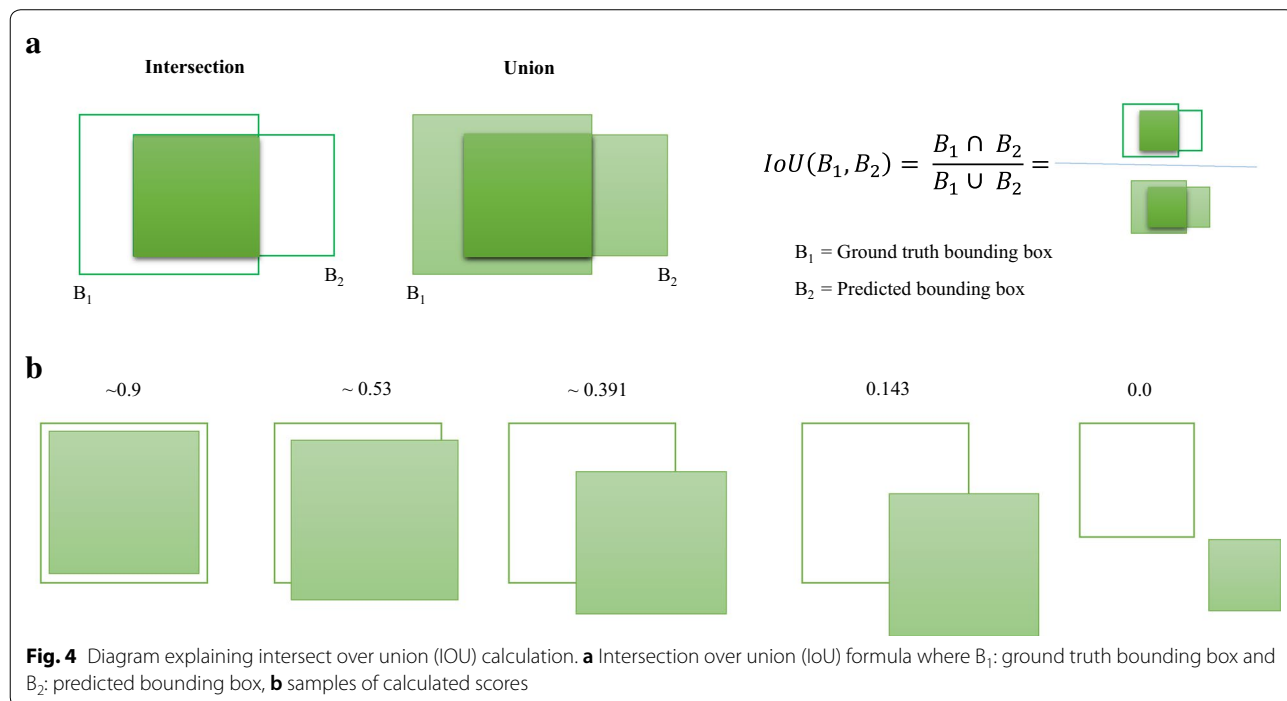


Table 2 Lists of hardware and software used in this study

Hardware and software	Specifications
Memory	128 GB
Processor	Intel Xeon E5-2667 v4 @ 3.20 GHz × 16
GPU	NVIDIA Tesla M60
Operating system	Windows Server 2016 × 64
Labeling software	Labellmg
Deep learning library	Tensor flow
Programming language	Python

Results and discussion

Banana dataset collection and annotation

Banana is liable to various types of pests and diseases for which symptoms occur in different parts of the plant (Table 1). The occurrences of these diseases depends on many factors, such as environment, temperature, humidity, rainfall, variety, season, nutrition, etc. For instance, certain diseases are localized in a particular country, region or continent, such as, *Xanthomonas* wilt of banana which is very specific to Africa. Therefore, reliable and accurate image collection at hotspots and strong labeling is very important. Since we are aiming for a global solution, we collected the image dataset of major banana diseases from different disease hotspots through our CGIAR network. Publicly available datasets poorly cover banana disease/pest symptom images, and the PlantVillage public dataset so far doesn't include banana images. We collected our own datasets of leaves infected by specific pathogens at different infection stages and other infected plant parts such as entire plants, fruit bunch, cut fruits, pseudostem and corms etc. with the help of well-trained banana experts using different cameras with various resolutions (Table 1, Fig. 2). Currently our CIAT-Bioversity, CGIAR dataset contains more than 18,000 expert pre-screened original field images, but in this study we utilized only 12,600 images to create banana image data sets. Since our ultimate aim is to develop a mobile-assisted banana disease detection tool targeting banana farmers across the globe/worldwide and the scientific community around the world, we enriched our image library with a diverse collection of images from different disease hot spots (Additional file 2: Table S1). To build a robust model, images were captured in real field scenarios on banana farms. A heterogeneous background is an essential feature of any real field images, most of the publicly available datasets are images of leaves in a controlled environment and simple background. For this reason, we tried to create many variations while collecting data from the field. The more the

variation in the dataset, the better is the generalization of the trained model. The images were captured with different camera devices (Additional file 2: Table S1) with diverse background. Furthermore, the challenging part of our image dataset is the background variations caused by the surroundings of the field, dried leaves on the floor, overlapping leaves from neighboring plants etc. This made our model more robust to adapt any changes in the real-time background.

We annotated the images to train our CNN by setting the images of different classes in distinct folders. We randomly picked 75% of images of each class and put them into a training set. Likewise, another 25% of images of each class were put into a test set. The training and the test set both contained 700 real field images per class (700×18 classes = 12,600) which has made the data set well balanced. The categories and the number of annotated samples used in our system can be seen in Table 1. We carried out a strong labeling approach whereby the banana experts confirmed the typical symptoms on each and every image of the data set, as a result we ended up with a total of 30,952 annotations (Table 1). Even though this strategy is time-consuming, we worked with three human experts to annotate the whole banana dataset which took almost 4 weeks. The tediousness of data collection and labeling had forced earlier studies [29, 30] to use small datasets to train and test classifiers. The use of small labeled datasets is also a limiting factor in machine learning, and it can lead to over or underfitting [31]. Most of the publicly available data sets are weakly labeled and resulted in poor performance.

Loss function

We summarized the total loss function (Additional file 1: Fig. S1a–f) only for the winner models (Additional file 2: Table S3). In general, we could observe that the accuracy increased while loss decreased gradually with epoch. For Corm damage images, the reported error was high until the 1500th iteration, then started to go down and after the 4000th step remained constant (Additional file 1: Fig. S1f), the same behavior was noticed in Pseudostem and Cut Fruits (Additional file 1: Fig. S1c, e), where after 2000 iterations the error remained constant until the end. For the entire plant and leaves (Additional file 1: Fig. S1a, b), although a loss was found to be below 0.3 in the last iteration, it suffered due to lot of variations, which was evident since these two models (entire plant and leaf) were found to be low accurate compared to other models studied (Table 3). The probable reason was clearly explained further by other performance metrics below.

Performance metrics and validation of developed models

In recent years, deep learning techniques, and in particular convolutional neural networks (CNNs), recurrent neural networks and long-short term memories (LSTMs), have shown great success in visual data recognition, classification, and sequence learning tasks. In the field of computer vision specifically, a set of CNN architectures have been emerging and they have proved to achieve tasks like object classification, detection and segmentation. In this paper, we retrained MobileNetV1, InceptionV2 and ResNet50 architectures using transfer learning to detect the banana pest and diseases. In order to improve the accuracy, the diseases were grouped by plant parts, and a different model was trained for each plant part (Table 1). Transfer learning is a progress that has the huge potential of being extensively used in crop phenomics and pest and disease detection. Transfer learning is particularly interesting, as its improved performance of deep neural networks by evading intricate data mining and labeling efforts [32].

There are different metrics to measure the accuracy and effectiveness in object detection models. In this study, we used mAP which is one of the widely used metrics in the literature [33, 34], especially for detection. Additionally, for each best model, a confusion matrix was generated. Earlier studies on detection revealed that the mAP score had become the accepted and standard way in competitions such as PASCAL VOC [35], ImageNet, and COCO datasets. More detail results are described below.

mAP score

The accuracy of the models based on mAP score is presented in Table 3. For the entire plant, leaves, pseudostem and fruit bunch models performed better in Faster R-CNN (faster regions with convolutional neural network) ResNet50 than others tested, which achieved an mAP score of 73%, 70%, 99%, and 97%, respectively. For cut fruits and corm, Faster R-CNN InceptionV2 worked better with the mAP accuracy of 95% and 98%, respectively. Fuentes et al. 2017 [33], used three CNN-based systems (Faster R-CNN, R-FCN and SSD) which performed object localization and disease diagnosis processes simultaneously and their system achieved more than 86.0% mean average precision on annotated tomato leaf images. In this present study, ResNet50 and InceptionV2 models have almost similar performance in all the cases compared to MobileNetV1 (Table 3). In generalized recognition, Faster R-CNN [36], models have been widely used and have achieved good results.

In this research, to achieve greater accuracy, we considered the complexity of the model as an important factor

Table 3 mAP metric score for different models developed from this study

Architecture	Model	Training time (h)	Accuracy
Faster R-CNN InceptionV2	Entire plant	30	0.728579
Faster R-CNN InceptionV2	Leaves	30	0.701833
Faster R-CNN InceptionV2	Pseudostem	30	0.999447
Faster R-CNN InceptionV2	Fruit bunch	30	0.973025
Faster R-CNN InceptionV2	Cut fruit	30	0.953296
Faster R-CNN InceptionV2	Corm	30	0.979151
Faster R-CNN ResNet50	Entire plant	20	0.734611
Faster R-CNN ResNet50	Leaves	20	0.703871
Faster R-CNN ResNet50	Pseudostem	20	0.999905
Faster R-CNN ResNet50	Fruit bunch	20	0.973634
Faster R-CNN ResNet50	Cut fruit	20	0.941152
Faster R-CNN ResNet50	Corm	20	0.976888
SSD MobileNetV1	Entire plant	50	0.446880
SSD MobileNetV1	Leaves	50	0.619923
SSD MobileNetV1	Pseudostem	50	0.98239
SSD MobileNetV1	Fruit bunch	50	0.936463
SSD MobileNetV1	Cut fruit	50	0.923548
SSD MobileNetV1	Corm	50	0.997296

to select the best architectures for the training set. This characteristic could be measured by counting the total amount of learnable parameters or the number of operations. As a result, we selected three architectures (Inception, ResNet, MobileNet). Since complexity is associated with the capacity of the model to extract more features from the images, it is expected inceptionV2 to be the most accurate among three architectures. However, it is always a trade-off between complex and simple architecture especially when you specifically think about mobile application.

We also noticed higher accuracy (more than 95%) for pseudostem, fruit bunch and corm compared to entire plant (73%) and leaf models (70%). This was expected in the entire plant model due to background noise in field environment, multiple classes (Fig. 5b) in the single image and wide angle. Wide-angle images are often more complex due to the substantial overlap of multiple leaves and symptoms are scattered in different leaves. In the case of banana it is much more complex because of the specific plant morphology and large leaf size. We also observed that during the labeling process, a single class per image was working as a ground-truth for the model (Fig. 5a, b). But in real life scenario, one image could have multiple classes as seen in Fig. 3. Developed entire plant and leaf model from this study is finding multiple classes in single image (Fig. 5b) which is more practical and useful in the real-time field application,

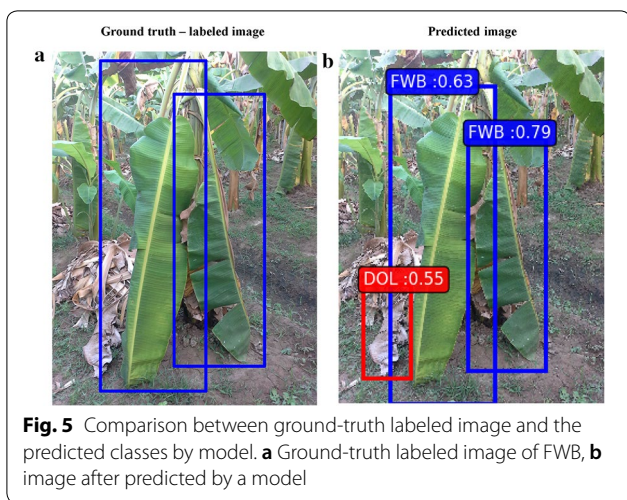


Fig. 5 Comparison between ground-truth labeled image and the predicted classes by model. **a** Ground-truth labeled image of FWB, **b** image after predicted by a model

since our ground-truth data is solely grouped on single class (Fig. 5a) which brought the mAP score lower than expected and it was the main cause. It was a little surprise for leaf model where we expect more than 90% accuracy since it is not very wide angle like the entire plant class. But these wide-angle images from field environment expected to have more background noises. To confirm these results in leaf model, we did an additional test to select images containing only one class per

image, which reflected on higher accuracy more than 70% (Additional file 2: Table S4). Moreover, this accuracy was further increased (more than 90%) when the new image dataset contain only one focused leaf per image (Additional file 2: Table S4). From these results, it is clear the complexity of the banana leaf morphology, disease symptoms, multiple classes in single image, field background noises etc. Unlike other crops such as rice, wheat, and cassava, banana leaves are very big, that makes the angle wider than other crops which increases the complexity in real-time field images (Fig. 5). In the case of pseudostem, fruit bunch, cut fruits and corm field images used in this study have more focused images towards the object with less background variation and single class per image which reduced the complexity and improved mAP score (Table 3).

Confusion matrix

In the field of deep learning, specifically the problem of statistical classification, the confusion matrix, also known as an error matrix, is a specific table layout that allows visualization of the performance of an algorithm. It considers different metrics: the true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) etc. Based on the results obtained on the test dataset, we generated a confusion matrix for each of the best architectures (Fig. 6a–f). Each confusion matrix

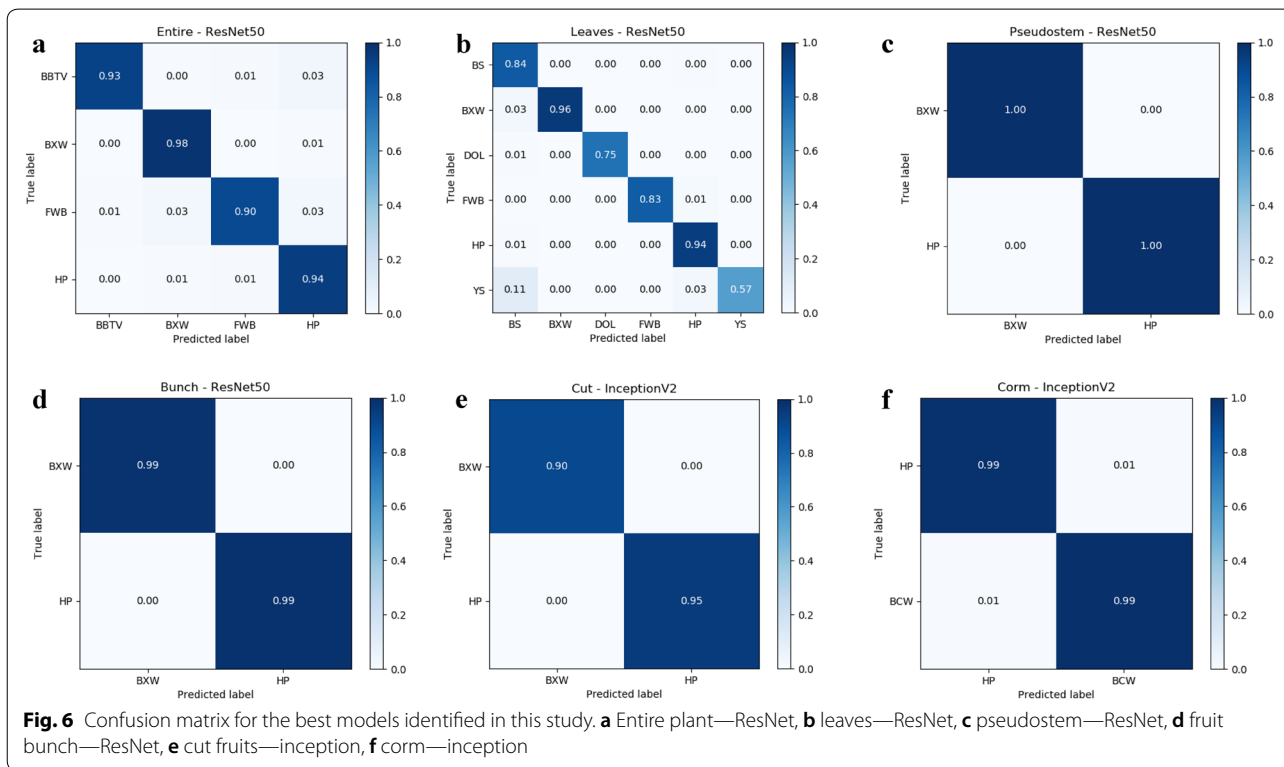


Fig. 6 Confusion matrix for the best models identified in this study. **a** Entire plant—ResNet, **b** leaves—ResNet, **c** pseudostem—ResNet, **d** fruit bunch—ResNet, **e** cut fruits—inception, **f** corm—inception

gave us a accuracy per disease (classes) and quantitative representation of the classes in which the model is misclassified or confused (Fig. 6a–f). Due to the complexity of the patterns shown in each class from different plant parts, the system tends to be confused on several classes that results in lower performance. Based on the results, we can visually evaluate the performance of the classifier and determine which classes and features are more prone to confusion. If the number of misclassifications between two particular classes becomes high, it indicates that we need to collect more data on those classes to properly train the convolutional architecture so that it can differentiate between those two classes. For this purpose, we also generated confusion matrix on our validation set for each best CNN architecture. Furthermore, it helps us to identify a future solution in order to avoid those inter-class confusions.

On comparing among the models, leaves produced a lot of confusion and low accuracy (57%) especially yellow sigatoka leaf spot classes (Fig. 6b), this was expected since YS and BS commonly produce similar symptoms in advanced stages, but early stage symptoms are unique (Additional file 1: Fig. S2). It is worth mentioning that yellow leaf spot disease appearing more frequently in Asia and Latin America and black leaf spot in Africa, and their treatment and disease controlling measures are almost similar. To handle these issues, we are currently collecting and labeling images of early stage symptoms for improving the accuracy of the model, and the ability to generalize. Because the dataset is not big enough, it was not considered in this study. We also observed medium prediction accuracy in the dried/old age leaf classes (Fig. 6b), it was obvious that, advanced stages of all leaf diseases will turn to be like dried/old age leaves and we expected this results. So early and mid-stage leaf symptoms are very important to detect the diseases with more accuracy. As we expected, the entire plant, corm, pseudostem, fruit bunch and cut fruits models, we had not found any accuracy or misclassification problems (Fig. 6a, c–f), which was ranged between 90 and 100% accuracy.

Conclusions and future directions

Many computer visioned approaches for automated crop disease detection and classification have been reported, but still, a detailed exploration of real-time pest and diseases recognition is lagging. In this paper, a novel method of using deep transfer learning method was explored in order to automatically detect banana pest and disease symptoms on different parts of the banana plants using real-time field images. This system introduces a practical and applicable solution for detecting the class and location of diseases in banana

plants, which represents a main comparable difference with other methods for plant diseases classification. The developed model was able to detect the difference between healthy and infected plant parts for different banana diseases. All images used in this study are available upon formal request through PestDisPlace (<https://pestdisplace.org/>) [37]. It consists of more than 18,000 original expertly pre-screened banana images collected on real farmer's field in Africa, Latin America and South India and was extended to more than 30,952 annotations. The experimental results achieved accuracy between 70 and 99%, of the different models tested. The robust models developed from this research will be more useful to develop the decision-support system to help early identification of pest and diseases and their management. Models developed in this study are currently utilized to develop a banana mobile app which is currently being tested by collaborative partners in Benin, DR Congo, Uganda, Colombia, and India (Additional file 1: Fig. S3). The developed model system from this study is easily transferable to other CGIAR mandatory crops.

Future work will comprise the development of a broad structure consisting of server side machinery containing a trained model and an application for smartphone devices with features such as displaying recognized diseases in other CGIAR mandatory crop such as Brachiaria, common bean, cassava, potato and sweet potato. Additionally, future work will involve disseminating the usage of the model by training it for banana disease recognition on wider applications, merging aerial images of banana growing regions captured by drones and convolution neural networks for instant segmentation of multiple diseases. By extending this research, we are hoping to achieve a valuable impact on sustainable development and strengthen banana value chains.

Additional files

Additional file 1: Figure S1. Loss function curve for the winner models. **a** Entire plant—ResNet, **b** Leaves—ResNet, **c** Pseudostem—ResNet, **d** Fruit bunch—ResNet, **e** Cut fruits—Inception, **f** Corm—Inception. **Fig. S2.** Early and late stage symptoms of banana leafspots. **a** Black sigatoka (BS) late stage, **b** Yellow sigatoka (YS) late stage, **c** Black sigatoka (BS) early stage, **d** Yellow sigatoka (YS) early stage. **Fig. S3.** Developed mobile application for Banana disease and pest detection. **a** Initial screen, **b** Image taking and Scan, **c** Diagnostic screen, **d** Recommendations and management.

Additional file 2: Table S1. Overview of banana data set collections, locations and image acquisition. **Table S2.** Description of major banana diseases and pest symptoms with their control measures. **Table S3.** Winner architecture for the models developed in this study. **Table S4.** mAP score metrics of leaf classes before and after segmentation.

Abbreviations

AI: artificial intelligence; API: Application Programming Interface; BBTV: banana bunchy top virus; BCW: banana corm weevil; BS: black sigatoka; BXW: Xanthomonas wilt of banana; CGIAR: Consultative Group on International Agricultural Research; CIAT: International Center for Tropical Agriculture; CNN: convolutional neural network; COCO: common objects in context; CM: confusion matrix; CPU: control process unit; DCNN: deep convolutional neural networks; DOL: dried/old leaves; DTL: deep transfer learning; Faster R-CNN: faster regions with convolutional neural network; FN: false negatives; FP: false positives; FWB: Fusarium wilt of banana; GPU: graphics process unit; HP: healthy plant; IoT: Internet of Things; IoU: intersection over union; LSTM: long-short term memories (LSTMs); mAP: mean average precision; SRS: single random sampling; SSD: single shot detector; TN: true negatives; TP: true positives; Tr: training; Tt: testing; Tv: validation; YS: yellow sigatoka.

Acknowledgements

The authors would like to thank the International Center for Tropical Agriculture (CIAT) IT unit for providing facilities and logistics support. Joe Tohme, Manabu Ishitani and Wilmer Cuellar from CIAT for guidance and support to do the research. The authors would also like to acknowledge Milton Valencia, Jorge Casas, Maria Montoya, Crysthian Delgado and Frank Montenegro for their help in image annotation and data collection. Thanks to Jules Ntamwira, Jean-Pierre Mafuta and Aman Omondi of Bioversity International, Africa and Deo Kantungeko of IITA, Burundifor their immense support to collect smartphone images. The farmers of Tamil Nadu Banana grower's federation, Trichy and planters of Tamil Nadu Hill Banana Growers Federation, Lower Palani Hills, Tamil Nadu India are also acknowledged for helping to collect data images. The authors also thank two anonymous reviewers for their detailed suggestions for improving the manuscript. As well as Angela Fernando, CIAT and Escalin Fernando, India for formatting and technical editing.

Authors' contributions

MGS, HR and AV designed the study, performed the experiments and are the main contributing authors of the paper. HR, AV, and MGS carried out data annotations, trained algorithms and analyzed the data. GB, SE, NS and WO collected over 17,000 images of disease and pest symptoms/damage, confirmed the symptoms and pre-screened all the images collected in Africa, Malaysia and India. MGS written the paper. All authors read and approved the final manuscript.

Funding

Funding for field smartphone image collection was provided by Bioversity International in the framework of the RTB-CC3.1 cluster and by the CIAT Agrobiodiversity Research Area to carry out the image processing work & preliminary app development (AGBIO1). This study was supported by the CGIAR Research Program on Roots, Tubers and Bananas (RTB). We thank the RTB Program Management Unit that supported this study and the CGIAR Fund Donors who support RTB (www.cgiar.org/who-we-are/cgiar-fund/fund-donor-s-2).

Availability of data and materials

The remotely sensed and field sampling data used in this study is available from the corresponding author upon reasonable request.

Ethics approval and consent to participate

Not applicable.

Consent for publication

All authors agreed to publish this manuscript.

Competing interests

The authors declare that they have no competing interests.

Author details

¹ International Center for Tropical Agriculture (CIAT), A.A. 6713, Cali, Colombia. ² Department of Soil and Crop Sciences, Texas A&M University, College Station, TX, USA. ³ Bioversity International, Bukavu, South Kivu Province, Democratic Republic of Congo. ⁴ Department of Biotechnology, Imayam Institute of Agriculture and Technology (IIAT), Affiliated to Tamil Nadu Agricultural University (TNAU), Tiruchirappalli, Tamil Nadu, India. ⁵ Bioversity International, Kampala, Uganda. ⁶ Bioversity International, c/o ILRI, Addis Ababa, Ethiopia.

Received: 3 May 2019 Accepted: 30 July 2019

Published online: 03 September 2019

References

1. FAO. Banana market review and banana statistics 2012–2013. Market and policy analyses of raw materials, horticulture and tropical (RAMHOT) Products Team. Rome; 2014.
2. Lescot T. World plantain and banana production systems. In: Proceedings XX international meeting ACORBAT: 9–13 September 2013; Fortaleza; 2013. p. 26–34.
3. Abele S, Twine E, Legg C. Food security in eastern Africa and the great lakes. Crop Crisis Control Project final report. Ibadan: Int Instit Trop Agric; 2007.
4. Nagayets O. Small farms: current status and key trends. In: The future of small farms; 2005. p. 355.
5. Blomme G, Dita M, Jacobsen KS, Perez Vicente L, Molina A, Ocimati W, Poussier S, Prior P. Bacterial diseases of bananas and enset: current state of knowledge and integrated approaches toward sustainable management. *Front Plant Sci.* 2017;8:1290.
6. Hillnhuetter C, Mahlein AK. Early detection and localisation of sugar beet diseases: new approaches. *Gesunde Pflanzen.* 2008;60(4):143–9.
7. Camargo A, Smith J. An image-processing based algorithm to automatically identify plant disease visual symptoms. *Biosyst Eng.* 2009;102(1):9–21.
8. Mohanty SP, Hughes DP, Salathe M. Using deep learning for image-based plant disease detection. *Front Plant Sci.* 2016;7:1419.
9. Intelligence G. The mobile economy Africa 2016. London: GSMA; 2016.
10. Kamilaris A, Prenafeta-Boldu FX. Deep learning in agriculture: a survey. *Comput Elect Agric.* 2018;147:70–90.
11. Ramcharan A, Baranowski K, McCloskey P, Ahmed B, Legg J, Hughes DP. Deep learning for image-based cassava disease detection. *Front Plant Sci.* 2017;8:1852.
12. Siricharoen P, Scotney B, Morrow P, Parr G. A lightweight mobile system for crop disease diagnosis. International conference on image analysis and recognition. Berlin: Springer; 2016. p. 783–91.
13. Wiesner-Hanks T, Stewart EL, Kaczmar N, DeChant C, Wu H, Nelson RJ, Lipson H, Gore MA. Image set for deep learning: field images of maize annotated with disease symptoms. *BMC Res Notes.* 2018;11(1):440.
14. Mwebaze E, Owomugisha G. Machine learning for plant disease incidence and severity measurements from leaf images. 2016 15th IEEE international conference on machine learning and applications (ICMLA). New York: IEEE; 2016. p. 158–63.
15. Hughes D, Salathe M. An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv preprint arXiv:1511.08060*; 2015.
16. Labellmg Software. <https://github.com/tzutalin/labellmg/>. Accessed 1 Feb 2019.
17. ImageNet Data Set. <http://www.image-net.org/>. Accessed 12 Mar 2019.
18. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2016. p. 770–8.
19. Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*. 2015.
20. Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H. Mobilenets: efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*. 2017.
21. Huang J, Rathod V, Sun C, Zhu M, Korattikara A, Fathi A, Fischer I, Wojna Z, Song Y, Guadarrama S. Speed/accuracy trade-offs for modern convolutional object detectors. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2017. p. 7310–1.
22. TensorFlow Python API. https://www.tensorflow.org/api_docs/python. Accessed 10 Feb 2019.
23. COCO Data Set. <http://cocodataset.org/>. Accessed 15 Feb 2019.
24. Reitermanova Z. Data splitting. In: WDS'10 proceedings of contributed papers, Part I, vol 10; 2010. p. 31–6.

25. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, Berg AC. Ssd: Single shot multibox detector. In: European conference on computer vision. Springer; 2016. p. 21–37.
26. Object Detection API Loss Functions Implementation, Tensorflow. https://github.com/tensorflow/models/blob/master/research/object_detection/core/losses.py. Accessed 5 Mar 2019.
27. Confusion Matrix for Object Detection. https://github.com/svpino/tf_object_detectioncm/blob/master/confusion_matrix.py. Accessed 10 Mar 2019.
28. Object Detection API, Tensorflow. https://github.com/tensorflow/models/tree/master/research/object_detection. Accessed 20 Feb 2019.
29. Dandawate Y, Kokare R. An automated approach for classification of plant diseases towards development of futuristic decision support system in Indian perspective. In: 2015 international conference on advances in computing, communications and informatics (ICACCI), IEEE; 2015. p. 794–9.
30. Mokhtar U, El Bendary N, Hassenian AE, Emary E, Mahmoud MA, Hefny H, Tolba MF. Svm-based detection of tomato leaves diseases. In: Intelligent Systems' 2014. Springer; 2015. p. 641–52.
31. Brahim M, Arsenovic M, Laraba S, Sladojevic S, Boukhalfa K, Moussaoui A. Deep learning for plant diseases: detection and saliency map visualisation. In: Human and machine learning. springer; 2018. p. 93–117.
32. Pan SJ, Yang Q. A survey on transfer learning. *IEEE Trans Knowl Data Eng*. 2010;22(10):1345–59.
33. Fuentes A, Yoon S, Kim S, Park D. A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors*. 2017;17(9):2022.
34. Sun J, He X, Ge X, Wu X, Shen J, Song Y. Detection of key organs in tomato based on deep migration learning in a complex background. *Agriculture*. 2018;8(12):196.
35. Everingham M, Eslami SA, Van Gool L, Williams CK, Winn J, Zisserman A. The pascal visual object classes challenge: a retrospective. *Int J Comput Vision*. 2015;111(1):98–136.
36. Zhang L, Lin L, Liang X, He K. Is faster r-cnn doing well for pedestrian detection? In: European conference on computer vision. Springer; 2016. p. 443–57.
37. Cuellar W, Mwanzia L, Lourido D, Garcia C, Martínez A, Cruz P, Pino L, Tohme J. PestDisPlace: monitoring the distribution of pests and diseases, version 2.0. International Center for Tropical Agriculture (CIAT); 2018.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Ready to submit your research? Choose BMC and benefit from:

- fast, convenient online submission
- thorough peer review by experienced researchers in your field
- rapid publication on acceptance
- support for research data, including large and complex data types
- gold Open Access which fosters wider collaboration and increased citations
- maximum visibility for your research: over 100M website views per year

At BMC, research is always in progress.

Learn more biomedcentral.com/submissions

