Early Detection of Diseases in Precision Agriculture Processes Supported by Technology

Jose A. Brenes, Markus Eger, and Gabriela Marín-Raventós

Abstract One of the biggest challenges for farmers is the prevention of disease appearance on crops. Governments around the world control border product entry to reduce the number of foreign diseases affecting local producers. Evenmore, it is also important to reduce the spread of crop diseases as quickly as possible and in early stages of propagation, to enable farmers to attack them on time, or to remove the affected plants. In this research, we propose the use of convolutional neural networks to detect diseases in horticultural crops. We compare the results of disease classification in images of plant leaves, in terms of performance, time execution, and classifier size. In the analysis, we implement two distinct classifiers, a densenet-161 pre-trained model and a custom created model. We concluded that for disease detection in tomato crops, our custom model has better execution time and size, and the classification performance is acceptable. Therefore, the custom model could be useful to use to create a solution that helps small farmers in rural areas in resource-limited mobile devices.

Keywords Diseases detection · Precision agriculture · Machine learning · Feature selection · Convolutional neural networks

1 Introduction

Currently, the world economy is facing big challenges. Today, most countries are involved in a socioeconomic slowdown, augmented by the COVID-19 pandemic,
requiring all sectors to innovate. In Costa Rica, for example, the agricultural sector had a growth equivalent to the 3.7% of GDP, in 2017. Meanwhile, in 2018, the growth of the sector was only 2.4% of GDP. That represents a decrease in the contribution to the country’s economy compared to the previous year [1]. Governments are expecting bigger decrements due to the pandemic recession. It is important to consider that with the reported decrease in production, a lot of families saw, in the last years, and will soon see their income reduced. According to the Costa Rican Performance Report of the Agricultural Sector [1], the reported GDP growth in 2018 was possible due to improvements in agricultural activities related to pineapple crops. In that area, farmers applied actions to control pests and diseases affecting pineapple crops fields.

Tomato and bell pepper crops are two of the most produced crops in the world [2]. In 2018, both crops reached high production rates in the order of 182 and 36 million tons of tomatoes and bell peppers, respectively.¹ In Costa Rica, tomatoes, and bell peppers are the two most consumed vegetables, reaching an intake of about 18.2 and 3.3 kg per capita of tomato and bell pepper, respectively [3, 4].

Nevertheless, even though horticulture is particularly important in Costa Rica, Ramírez and Nienhuis [5] emphasize on pests and diseases suffered by these types of crops. Pests and diseases force farmers to apply chemical pesticides with harmful consequences for the environment. Thus, detecting them as early as possible is especially important in order to minimize the need for these chemicals.

In addition, Abarca [6] performed an analysis of climate change and plagues in crops cultivated in the tropics. He mentioned that it is crucial to create solutions that enable early alarms against disruptive plagues. In this way, it can be possible to react fast to the plague and avoid the appearance of the disease with a high impact in crops.

Dimitrios et al. [7] state that tools implementing emergent technologies to support farmers in pest and disease management are imperative now more than ever. The high incidence of pests and plagues on horticultural crops forces us to join efforts to create tools focused on this type of crops. These solutions must enable farmers to act immediately. For practical applications, we must also consider that farms may not have ready access to powerful hardware. An important consideration is therefore also to provide solutions that can run with very little resources, such as on a standard smartphone device.

There are some IT research efforts around the world focused on contributing to the agricultural productive sector, but mainly on automatic fertigation [8], while pest and disease management have been less studied. We center our research on this area. We propose the use of machine learning techniques to detect disease appearance on crops early to enable farmers to attack them on time or to remove the affected plants avoiding its spread. To achieve it, we keep in mind all the time, that the resulting solution must have to be runnable on mobile devices with very low resources to make it accessible to small-scale farmers in rural areas.

This paper is organized as follows. In the next section, we define some concepts needed to understand our research. In the following section, we mention different studies made by other researchers that are related to our research. Afterward, we

¹https://www.fao.org/faostat/en/#data/QC.
present statistical analysis to select the features and artificial intelligence techniques to use in the tool to be provided to small farmers. Next, we present a study case in which we evaluate the feasibility of the proposal by comparing two different convolution neural networks in terms of performance, the training execution time, and the model’s resultant size. Finally, we present the conclusions of this research process.

2 Theoretical Framework

In this section, we introduce some technical terms that must be clear to achieve a comprehension of our research.

2.1 Precision Agriculture

Dwivedi et al. [9] define precision agriculture as a farming managing method based on observing, measuring, and responding to variability in crops. The goal of precision agriculture is to define a decision support system (DSS) that can be used to manage the production optimizing its processes and increasing its yield.

Bongiovanni and Lowenberg-DeBoer [10] mention that precision agriculture is based on the idea of automation of site-specific management (SSM). This is achieved by using information technology to tailor input use to reach desired outcomes or to monitor those outcomes. Precision agriculture merges technologies with agricultural processes to achieve an integrated crop management [11].

2.2 Horticultural Crops

The horticultural crops are defined as herbaceous plants cultivated with the goal of self-consumption as well as for commercialization in internal and external markets [12]. Vegetables are particularly important crops for mankind because they are an extraordinarily rich source of vitamins and nutrients. They cover crops cultivated directly in the land as lettuce, celery, and parsley; tubers including potatoes and onions; fruits such as tomatoes and bell peppers; and flowers such as broccolis and artichokes [12]. Crops that will be used in our research are described next.

Bell Pepper

Bell peppers belong to the family of Solanaceae. *Capsicum annuum* is the most cultivated species in countries located in the tropics [13]. According to [2], bell peppers are one of the most frequently produced vegetables in Costa Rica. It is cultivated in an opencast way, but it is also produced inside greenhouses. Currently, producers
can get access to a hybrid version known as “Dulcitico” developed by the researcher Carlos R. Echandi at the Fabio Baudrit Moreno Agricultural Experimental Station (EEAFBM) of the University of Costa Rica in 2013 [3]. This permits producers to get access to high quality, cheap seeds locally. “Dulcitico” is a highly productive hybrid which is also adapted climatologically to the region [14].

**Tomato**

Tomato crops also belong to the family of Solanaceae. The more cultivated species is *Solanum lycopersicum*, which has its origins in the Andes region [15]. The tomato plant is a perennial herb that can be cultivated annually [16]. Its fruits are harvested for consumption. According to [15], tomato is the most cultivated and consumed vegetable in Costa Rica.

Under adverse climatic conditions (high temperatures and humidity), tomato crops suffer many problems related to the presence of diseases which affect their production [16]. Diseases cause low-yield and quality and can even cause total crop losses. Authors point out that due to diseases’ aggressiveness, it is important to prevent their appearance.

### 2.3 Diseases in Crops

Diseases are the main limiting factor in the production of vegetables in the tropics, according to [17]. In some cases, such as in tomato crops, there are pathogens resistant to existing treatments. Since plants do not respond to agrochemicals, farmers must remove affected plants from crops to avoid a greater affectation.

The high incidence of disease appearance is due to the use of seeds created for regions distinct to the one in which the crops are cultivated [17]. For example, farmers sometimes use seeds created for temperate zones in tropic regions. This limits the plants’ resistance to diseases since crops are not adapted to the tropical climatic conditions.

The creation of hybrid plants, like “Dulcitico” in Costa Rica, enables farmers to fight some diseases. Nevertheless, even though we have seeds prepared for the prevailing climatic conditions, the best way to deal with pests and diseases is the prevention and know-how acquired over time [18].

It is important to provide farmers with tools for the early detection and control of diseases, so that they can reduce the uncontrolled propagation of diseases and can avoid loss of production. Disease detection is regularly done by examining the plants, and especially their leaves. This can be done manually but requires time and expertise. It can also be automated.
2.4 Computer Vision

Artificial intelligence (AI) techniques focused on computer vision and machine vision have been used in agriculture [19]. These techniques represent an alternative for the identification of diseases in crops achieving good results. The techniques used are based on the extraction of information (features) from images (photographs). Image information is used to create classifiers that permit categorizing new samples of images [19].

Tools that use computer vision techniques enable farmers to make a better use of pesticides [20]. When farmers detect a possible specific disease in the crop, they can apply the correct pesticide and control the quantity to apply according to the infection. Both parameters can be obtained automatically or semi-automatically through trained computer tools.

2.5 Classification Algorithms

In addition to computer vision, there are other techniques in the field of artificial intelligence that can be useful to classify images based on crop data. Many techniques are used to create classifiers which can be used to detect diseases in plant leaves images [21].

In this paper, we propose the use of techniques such as Naïve Bayes, linear discriminant analysis, and neural networks to classify images of plant leaves affected by diseases.

2.6 Image Classification

To classify images based on computer vision and machine learning techniques, it is required to extract information from the images to be analyzed. Features extracted from images represent characteristics associated with color, shape, and texture. Those features are used by algorithms in the learning process to carry out the classification.

It is possible to extract the following features from images:

- **Color:** It can be obtained by using different color models:
  - Color (RGB): The commonly used three channels representing the combination (additive) of colors Red, Green, and Blue. Each color is represented by a decimal number in the range 0–255. The color combinations can be analyzed as a cube in which the three dimensions are the three colors (RGB) [22].
  - Color (CYMK): This is a representation by using complementary colors (Cyan, Magenta, Yellow, and Black). This representation is mostly used for output devices like printers [22].
– Color (HSV): This is a representation of colors in a way that is more closely aligned with the way human vision perceives color making attributes [22]. The H represents the hue and takes values from 0 to 179; the S corresponds to the saturation which takes values from 0 to 255, and the V is the value in the range 0–255. Another representation uses lightness instead of value (in the same range). That last representation is referred to as HSL.

– Color Grayscale: This corresponds to a model that represents the amount of light in an image. The value for pixels can be in the range 0–255.

• Texture: texture is recognizable in tactile and optical ways. Texture analysis in computer vision is complex. According to [23], texture in images can be defined as a function of spatial variation of the brightness intensity of the pixels. To calculate the texture of the images, it is possible to use the Haralick method [24] with which it can be obtained 13 values representing textural features.

• Shape: for the calculation of the shape in the images, we use the Hu-moments invariants which are seven numbers, known as moment invariants, which are calculated by using weighted average of image pixel intensities. According to [25], moment invariants store all information about shapes in an image, so they can be used for shape recognition in images.

In the next section, we describe some of the existing techniques to help farmers detect diseases.

3 Related Work

The use of technological tools in agriculture is an active research field. Precision agriculture seeks to maximize agricultural production using technological solutions. This research area has been growing in the last years. Researchers around the world have been dedicated to the creation of systems and applications which enable farmers to process data collected from crops to transform it into valuable information for decision making.

Studies seeking the automation of irrigation in crops using data sensed from the cropfields in real time can be found in the literature. Meanwhile, other researchers focus on pest and disease prevention, control, and avoidance [8]. It is possible to find studies which propose automatic detection of diseases in tomato crops [26, 27], wheat [28, 29], grapes [28], among others. All these studies have similarities. They all propose solutions where real-time image processing is used to detect diseases affecting crops. Some of the mentioned studies use automatic analysis of plant leaves using machine learning and computer vision techniques, but in these cases, authors are more concerned on the precision and do not focus on the complexity of solutions and their computational cost.

The research presented in [27] shows a completely automated solution that includes from the image capturing process to the classification results. Nevertheless, even though the solution does not require farmers’ participation, the system
performance is quite similar to the one we propose in this research. So, we consider it is not necessary to automate all the process. Farmers can identify affected plants and take some pictures to feed the disease detection system. Additionally, putting static cameras in the crop field may be very expensive, and if few are used, it reduces the possibility to capture pictures of infected plants out of camera range.

Authors in [30, 31] analyze distinct techniques to recognize species through leaves color images analysis. A combination of techniques with which the researchers process color, shape, and texture features extracted from images to classify them is presented in [30]. Meanwhile, in [31], authors present a novel method to recognize plant species even with fragmented images, i.e., when the complete plant leaf is not present in the image. Both studies result relevant to this work. In both cases, the authors keep in mind a low computational cost requirement, a goal we also are seeking. Our research is somehow different. We are working with the classification of plant leaf images but we focused on disease detection present in leaves and not trying to determine the identity of the plant leaves belong to. We propose to do so analyzing only the color feature and by using convolution neural networks.

Other authors [32, 33] focus their studies on the creation of mobile applications to execute the identification of diseases. They depict that processing power in mobile devices results in a limitation. For that reason, authors recommend considering it in the creation of the solutions. A strategy to mitigate the lack of processing power is the use of cloud computing. However, we consider that solutions including data analysis in the cloud limit their use in rural areas in developing countries. In this setting, there is commonly no Internet connectivity or it is very poor.

It is also possible to find studies focusing on specific crops. First, a literature review of published proposals related to disease detection in bell pepper crops is presented in [34]. They conclude that in case of fungal diseases, automatic detection enables farmers to avoid high affectation by removing the affected plants (fungal diseases spread through spores that travel in the air).

In our research, we evaluate and propose an efficient algorithm for early detection of diseases in horticultural crops. The value of our proposal is related to our focus on two main requirements: mobile devices with very limited resources and local processing. These two requirements can enable farmers to use the solution without Internet access. Our focus also lies on the automation of the identification process, not in the control process. Our goal is to provide suggestions to farmers to help them act quickly.

4 Feature Selection for Automatic Diseases Detection: An Experiment

In 2019, a bell pepper crop (*capsicum annuum*) inside a greenhouse at EEAFBM was suffering the presence of a disease with high affectation and an extremely fast
propagation. The disease was causing plants death and a lot of problems for the researchers.

Leaves from healthy and infected plants are different. It is possible to see that the leaves of disease affected plants present degradation of colors in some rounded formations across the leaf. That affectation can also be present in other parts of the plant like the stem and the fruits. Additionally, it was possible to note that rounded formations in leaves had a different texture in comparison with the healthy sections of the leaves. In Fig. 1, we show an example of a healthy leaf (Fig. 1a) and an affected leaf (Fig. 1b), both taken from the mentioned crop. In Fig. 1b, it is possible to see the described affectation, with the rounded formations.

Different diseases exhibit different visual markers on the leaves. Thus, computer vision techniques could be used for image classification to identify the presence of a disease, as well as to distinguish between different diseases. To do that we started exploring the different features that can be extracted from images to describe them. We decided to use color (RGB), texture, and shape for the analysis, considering that the disease affects leaf color, creates circles (rounded formations), and changes the texture of the leaves. We designed an experiment to identify which feature (or combination of features) provided the best results. Different artificial intelligence techniques were used for the classification task. We also wanted to identify if there is a significant difference in which technique to use for the classification.

4.1 Objectives

The proposed three specific objectives for the experiment were:

- To determine if the features extracted from an image can affect the detection of a disease in an affected crop leaf in terms of the classification precision.
- To identify if the used classification technique can affect the precision in the detection of a disease in an image of an affected leaf.
- To determine if there exists an interaction between the features extracted from an image and the techniques used for the classification task.
The first objective allows identifying the features to use in the classification task. It is necessary to consider that we want to create a mobile solution for disease detection. In mobile solutions, it is important to take into consideration that processor and memory resources are limited. For that reason, we needed to identify the minimum number of features required in the process.

Through the second objective, we wanted to explore if there is a significant difference between the use of a classification technique instead of the use of another. That is useful to select the best technique or the most efficient technique to be run on a mobile device.

The third objective tries to determine the relation between classification techniques and features. Our goal was to identify the best combination of features and techniques to be used.

### 4.2 Methodology

We defined as an experimental unit the different arrays formed by the combinations of features extracted from images: color (RGB), texture, and shape. The precision of the classification was the response variable. The precision is obtained as the ratio between true positives and the sum of true positives and false positives, i.e., the percentage of leaves classified as infected by the algorithm that were actually infected.

As design factors we identify:

- The classification techniques, in which case we consider:
  - Neural Networks (NN): a MultiLayer Perceptron classifier implementing ReLu activation function, Adam optimizer, alpha = 0.0001, 100 neurons in the hidden layer, initial learning rate = 0.001.
  - Naïve Bayes (NB): a Gaussian Naive Bayes classifier.
  - Linear Discriminant Analysis (LDA): a classifier with a linear decision boundary implementing singular value decomposition solver and a tolerance of 0.0001.

- The features extracted from the images. We had seven combinations of features to be analyzed:
  - C: Color
  - T: Texture
  - S: Shape
  - CT: Color + Texture
  - CS: Color + Shape
  - TS: Texture + Shape
  - CTS: Color + Texture + Shape

---

2 [https://scikit-learn.org/stable/auto_examples/model_selection/plot_precision_recall.html](https://scikit-learn.org/stable/auto_examples/model_selection/plot_precision_recall.html).
We decided to fix some factors to specific values for the experiment. Those factors were:

- **Disease present in the images:** The crop was infected with a leaf spot disease known as “ojo de gallo.” All the leaves images were healthy or affected by that disease.

- **Crop and plant species:** The crop was a plantation of sweet pepper, a genetically modified hybrid known as “Dulcitoico” (*capsicum annuum*) which was created by the Fabio Baudrit Moreno Agricultural Experimental Station at the University of Costa Rica.

- **Time of day during which photographs of the leaf were taken:** Another fixed factor was the time of day during which the plant leaf images were produced. To produce good quality photographs of leaves, it is important to preserve the amount of light affecting the images. For that reason, we decided to take the photographs from 11:00 to 13:00 h Central American Time on a sunny day.

We defined as nuisance factors the presence of other diseases affecting the plants and the angle in which the images were taken. The first nuisance factor was minimized with the help of a professional in Agronomy (the person in charge of the original experiment using the crop), who checked that plants did not have another disease at the moment of image creation. The second nuisance factor was related to the amount of light present in the images since, if the angle of the camera changes, the colors in the image can vary significantly. To minimize the second nuisance factor, all the photographs were taken by a specific person, approximating the same angle and in a fixed setup with an uniform background.

We have two factors with distinct levels for each factor. For that reason, we designed a factorial experiment to analyze the previously mentioned factors and the interaction between them.

Taking into account the experimental factors mentioned above, Fig. 2 shows a general overview of the process to be followed. Each of the steps presented in Fig. 2 will be described in detail in subsequent sections.

---

**Fig. 2** General overview of the process to be followed in the experiment

We decided to fix some factors to specific values for the experiment. Those factors were:

- **Disease present in the images:** The crop was infected with a leaf spot disease known as “ojo de gallo.” All the leaves images were healthy or affected by that disease.

- **Crop and plant species:** The crop was a plantation of sweet pepper, a genetically modified hybrid known as “Dulcitoico” (*capsicum annuum*) which was created by the Fabio Baudrit Moreno Agricultural Experimental Station at the University of Costa Rica.

- **Time of day during which photographs of the leaf were taken:** Another fixed factor was the time of day during which the plant leaf images were produced. To produce good quality photographs of leaves, it is important to preserve the amount of light affecting the images. For that reason, we decided to take the photographs from 11:00 to 13:00 h Central American Time on a sunny day.

We defined as nuisance factors the presence of other diseases affecting the plants and the angle in which the images were taken. The first nuisance factor was minimized with the help of a professional in Agronomy (the person in charge of the original experiment using the crop), who checked that plants did not have another disease at the moment of image creation. The second nuisance factor was related to the amount of light present in the images since, if the angle of the camera changes, the colors in the image can vary significantly. To minimize the second nuisance factor, all the photographs were taken by a specific person, approximating the same angle and in a fixed setup with an uniform background.

We have two factors with distinct levels for each factor. For that reason, we designed a factorial experiment to analyze the previously mentioned factors and the interaction between them.

Taking into account the experimental factors mentioned above, Fig. 2 shows a general overview of the process to be followed. Each of the steps presented in Fig. 2 will be described in detail in subsequent sections.
4.3 Dataset Construction

For the execution of the experiment, we decided to create a dataset from the infected crop. We took 121 photographs of healthy and infected leaves, distributed in 83 healthy and 38 infected leaves from distinct plants in the same crop. A sample of the obtained images can be seen in Fig. 1.

To increase the size of the dataset, we made a sampling with repetition without replacement, in which we created sets of 30 images, healthy (15) and infected (15), to use in the training phase. Additionally, with the remaining images we created a second sample of 20 images of healthy (10) and infected (10) leaves. That second sample was used for the testing phase. An important aspect to consider is that both sets were disjoint; i.e., images in the training set were not present in the test set.

Additionally, we repeated the sampling process 100 times for each treatment. We eliminated repeated samples. At the end of the sampling process, we got a total amount of 2100 sets for the training phase and 2100 sets for the test phase.

4.4 Features Extraction and Images Processing

We used Python programming language to create a script with which to extract features from images. For that task, we used the packages mahotas\(^4\) and opencv\(^5\) to extract texture, color, and shape from images, respectively. We decided to store the data extracted from images in csv files. Each row in the file represents an image, and each entry starts with the label for each image indicating if the leaf in the image was healthy or infected.

We created another script for the sampling process, which loads the csv files and runs the process, creating csv files with the different datasets.

For the processing of images and the creation of classifiers, we used the package scikit-learn\(^6\) which is a library for machine learning in Python. That library provides off-the-shelf implementations of various machine learning approaches. In our case, we created several scripts to train and test classifiers using the selected algorithms.

4.5 Analysis of Results

We conducted statistical analysis to identify the effect of the classification algorithms and the features extracted from images on the classification precision. We started by exploring the data. Due to the results showed in Table 1, we discard the possibility to conduct a normality analysis.

\(^4\)https://mahotas.readthedocs.io/en/latest/.
\(^5\)https://pypi.org/project/opencv-python/.
\(^6\)https://scikit-learn.org/stable/.
Consequently, we ran the nonparametric Scheirer–Ray–Hare test in combination with a Dunn test for the post-hoc analysis [35]. The Scheirer–Ray–Hare test results are shown in Table 2. In that table, it can be observed that each of the chosen factors had a significant impact on the result, but that there is no interaction between factors.

The results of the Dunn test pointed out that treatments including Color (C) and Color + Shape (CS) are significantly different from the others. As a consequence, we prefer the use of the feature “Color” individually for a classifier or the combination of “Color” and “Shape.” The results for the algorithms show that there are no significant differences for different classification algorithms. Thus, we will prefer to select the one with the best results for images classification tasks.

Through the described experiment we could determine that the most significant image feature to use in classification is color (RGB). Also, the experiment results show that it could be useful to include the shape feature in the analysis. Both features can be used in the creation of a classifier to automate the process of disease detection. Additionally, we concluded that it is possible to use at least Naïve Bayes, linear discriminant analysis or neural networks techniques to build an image classifier.

5 Disease Detection in Tomato Plant Leaves: A Study Case

We propose to execute a study case to determine the effectiveness of a machine learning technique for the classification of tomato diseases in plant leaves images. We also want to determine if the tested models can be used to run in a mobile scenario in which the available resources are very limited.

Dey et al. [36] analyze distinct scenarios in which machine learning technique can be used to get the best results. They state that artificial neural networks (ANN) can be used when the analysis is based on pixels from images, known as hard classification. ANNs may also be useful in scenarios where the analysis is nonparametric, i.e., not
based on statistical parameters; and there is a high degree of variation in the spatial distribution of classes across the data [36].

In a literature review of distinct classification techniques, [37] argue that artificial neural networks have some advantages against other techniques. According to them, ANNs have high computational power to deal with noisy input efficiently, and they can learn from training data. That characteristic is extremely useful in contexts where the computational resources are limited, and the input data consists in photographs of plant leaves.

Albawi et al. [38] analyze the use of convolutional neural networks (CNN) for image classification. All the components of this type of ANN are described, including the most important component, the convolutional layer, an element that provides the possibility to detect and recognize features regardless of their position in the image. They report the convolutional layer gives CNNs good results in classification tasks.

We decided to use CNNs to create the classifier due to their particular suitability for the task. Before we describe the models, we used, we will first define the objectives for our case study.

5.1 Objectives

We want to identify which CNN implementation offers the best results while consuming the least computational resources. For that, we plan to compare a pre-trained CNN model and a custom CNN model regarding their relative performance in classification, the time consumed during the learning stage, and the size of the model, which is directly related with how computationally expensive it will be to use the model for classifications of new images.

5.2 Methodology

To achieve the proposed objective, we selected the densenet-161\footnote{https://pytorch.org/hub/pytorch_vision_densenet/} pre-trained model, a dense convolutional network that connects each layer to the others in a forward way. According to the documentation, this CNN model is designed especially for image classification in which data correspond to 3-channel RGB images. It achieves a score of 22.35 for top-1 error (percentage representing the times when the classifier not assign the highest probability to the correct class), and a score of 22.35 for top-5 error (percentage representing the times when the classifiers not include the correct class in the first five guesses). These scores are the lowest rates from distinct pre-trained CNN models.

To use the pre-trained model, the following assumptions must be taken into consideration:
Table 3 Defined hyper-parameters for the models

| Hyper-parameter       | Pre-trained CNN | Custom CNN |
|-----------------------|-----------------|------------|
| Input images’ size    | $224 \times 224$ | $64 \times 64$ |
| Batch size            | 48              | 48         |
| Number of epochs      | 100             | 1000       |
| Initial learning rate | 0.001           | 0.001      |
| Optimizer             | Adam            | Adam       |
| Loss function         | Negative log-likelihood | Cross-entropy |
| Activation function   | ReLu            | ReLu       |

- Data must be arranged in shape $3 \times H \times W$, where $H =$ Height and $W =$ Width must have a size of at least 224 pixels.
- The data must be loaded in a range $[0, 2]$ and then normalized using mean $= [0.485, 0.456, 0.406]$ and standard deviation $= [0.229, 0.224, 0.225]$. It is important to consider that those values are fixed and cannot be changed.

We used Python as our programming language and the pytorch\(^8\) package to load the data, create the models and run the training and test of the classifiers. For the pre-trained version of the classifier, we created a reference model, loaded the pre-trained one and overwrote the reference model with the pre-trained one. We then used this pre-trained version as a starting point and performed additional training particular to our dataset. For the custom CNN model, we created a custom network with two convolutional layers and three fully connected layers.

We used the hyper-parameters in Table 3, for the two models.

We used a scientific workstation to run the training and test phases. We executed the learning process of the models on a Nvidia Quadro P400 GPU. To calculate the analysis metrics, we used the classification report\(^9\) provided by the package scikit-learn. By using this package, we can directly obtain accuracy, precision, recall, and F1-score metrics from the test process. To calculate the time consumed in training and test stages, we used the computer clock. To estimate the size of the model, we exported the weights and the whole model to a file after the training and test stages.

Figure 3 shows the overall process to be followed. The three process steps will be described in the next sub-sections.

5.3 Dataset

The dataset we selected is provided by the project PlantVillage\(^10\) which consists of 54,303 healthy and unhealthy leaf images divided in 38 classes by species and

---

\(^8\)https://pytorch.org/.

\(^9\)https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html.

\(^10\)https://www.tensorflow.org/datasets/catalog/plant_village.
diseases. The dataset contains images for 18 crops: apple, blueberry, cherry, corn, grape, orange, peach, pepper, potato, raspberry, soybean, squash, strawberry, and tomato.

We decided to run the study case for the tomato samples only, as the dataset contained a variety of different diseases for this crop. In the future, it is possible to create a model for each crop (and for its diseases). Solutions that support all crops and diseases can be harder to execute, and farmers can usually run the classification task only for a specific known crop.

We selected the tomato samples which consist of ten classes (one healthy and nine with different diseases). A total amount of 18,160 images of tomato leaves were considered. We divided the dataset into three parts: 70% (12,712 images) for training, 15% (2724 images) for test, and 15% (2724 images) for validation.
In Fig. 4, we show several samples of photographs from this dataset. The (a) image represents a healthy leaf, while the others correspond to leaves affected by distinct diseases: bacterial spot (b), leaf mold (c) and septoria leaf spot (d).

5.4 Learning Process

The training and validation stages were run over the entire dataset. Since the learning process converges well within 100 epochs for the pre-trained model, based on empirical observation, we decided to set the number of epochs in 100 repetitions. Regarding the custom model, we decided to increase the number of epochs by 10-times based on the behavior of the training loss. In each epoch, we loaded the dataset in batches and proceeded to extract the RGB from those images. Next, we performed gradient descent for the total of iterations. At the end of the process, we calculated the performance metrics with the classification report based on the test dataset.

We used the test dataset to compare the results obtained from the training and validation stages. Test dataset provided an unbiased evaluation of the trained model, which was fitted with training and validation datasets. By loading the saved model, we tested the classification with the validation dataset and calculated the same metrics. Next, we compared the performance metrics for training and test stages to guarantee that they did not differ a lot, assuring the correctness of the classifiers.

We registered the execution time of the learning process. It is important to keep in mind that we use the same workstation to train both models. During the execution of the learning process, no other tasks were executed on the workstation.

As we stated above, we then analyzed the resulting size of each of the models and their weights. To do that, when the learning process finished, we proceeded to export the whole model (definition and weights), as well as only the weights.

5.5 Analysis of Results

After training, the pre-trained model densenet-161 was more accurate and precise than the custom model. Figure 5 shows the performance metrics for both models.

Additionally, Fig. 6 shows the results of the classification for the pre-trained densenet-161 model. In the figure, the confusion matrix shows the high classification rate obtained, in which the class “Target spot” is the one with most images classified incorrectly. However, in general, the classification produced by the pre-trained densenet-161 model results in a correct prediction.

Figure 7 shows the confusion matrix for the custom model. As seen in the matrix cells, most results lie in the diagonal, thus the corresponding classifications are good. “Early blight” and “Late blight” are the classes with more incorrect classifications, but they are not very numerous.
Regarding the training time, we registered a total of 28 h, 31 min, and 10 s for the pre-trained model, and a total of 04 h, 47 min, and 59 s for the custom model. That difference in time execution is given by the structure of the model. In the pre-trained model, the network is very dense, which means that neurons have more connections, and therefore, more weights must be calculated each time. In the custom model, even though we worked with fully connected layers, there were fewer connections
between neurons. This resulted in a training time that was much smaller than for the pre-trained network.

The size of the resulting model after training was also measured. We got a file size of 115,430 KB for the pre-trained model and 115,384 KB for only its weights. In the case of the custom model, we got a file size of 259 KB for the entire model and 244 KB for the weights. As can be seen, the size of the custom model is smaller than the pre-trained model in both cases (whole model and only-weights) by several orders of magnitude. When we try to load the model on a mobile device, it is easiest to load the smaller file due to limited resources. For this reason, the custom model is likely a better fit for our application.

According to the obtained results, while there are differences in accuracy and precision, they are negligible compared to the difference in model size. We may think that we can always prefer the model that has a greater accuracy and precision, but in the scenarios in which we work, we must also consider other parameters. If we consider the training time and the resulting file size of the two models, it is advisable to select the custom model.

We would like to reiterate that these results are only for tomatoes. In order to determine the suitability of our custom model, we intend to also run the process for other crops when we want to create a solution for multiple crops. In this case, having a low resource consumption algorithm becomes even more important, since different models for each crop must be included in the application, which further increases the file size and computational complexity of the overall solution.

![Confusion matrix for custom model](image)

**Fig. 7** Confusion matrix for custom model
6 Discussion

In the experiment presented in Sect. 4 of this paper, we aimed to select the best combination of features extracted from images and the most suitable artificial intelligence algorithm to be used in the construction of a low resource consumption classifier. The features extracted from images were color, shape, and texture, while the algorithms we evaluated for the classifier were linear discriminant analysis, Naïve Bayes, and neural networks.

In [30] and [31], researchers work with the three mentioned features (color, shape, and texture) and with two of the considered algorithms: linear discriminant analysis and neural networks, respectively. Nevertheless, in both of these cases, the objective of the solution was the identification of plant species. As it was mentioned above, our objective is focused on disease detection where plant leaves are affected by color degradation, distinct formations in the leaves, and texture changes. For that reason, we decided to run first an experiment to select the features and algorithms to be used, getting as a result that we can use the color features individually for the analysis. Related to the algorithm, the results show that we can use any of the three considered algorithms, since they field similar classification performance.

Since convolution neural networks (CNN) have become popular specialized algorithms, we decided to compare two distinct CNN models and use the color (RGB) feature. Remember that our goal is to reduce the computational complexity of the final solution. Keeping this in mind, we planned in a study case to identify the model with an acceptable classification performance but with the best execution time and size.

Regarding the study case results, our custom model can achieve acceptable classification performance keeping the learning process execution time low and the size of the model extremely tiny. The latter helps us run the model in devices with very low resources, accomplishing the final goal we were pursuing.

As stated in [39], even if the machine intelligence cannot be defined, it could be measured. It is important to measure the intelligence of solutions that use artificial intelligence to make decisions and to do tasks in which it is necessary to act like the human brain. The problem that we propose to solve by using machine learning techniques represents a part of more complex decision support systems that can be developed to help farmers in their daily activities. We consider that measuring the intelligence of our solution can be useful, focusing on the task of recognizing diseases in plant leaves by using supervised learning. In this scenario, our solution tries to help un-experienced farmers to detect diseases in crops, an activity that can be done by experienced farmers or specialists easily.
7 Conclusions

After conducting the experiment and the study case, we can conclude that for the scenario of disease detection in tomato crops, it is possible to create a solution by using the PlantVillage dataset and by implementing convolution neural networks for the classifier. We demonstrate that the densenet-161 model presents better results in terms of classification performance in comparison with our custom convolution neural network model. Nevertheless, it also requires significantly more training time and the size of the resulting model is several orders of magnitude larger, limiting its applicability in the field.

In the scenario of disease detection in crops, by using resource-limited mobile devices, it could therefore be better to use a custom model instead of a pre-trained model. Custom model performance is already close to the full densenet-161 model, but could be improved in the future; however, we consider that the benefits in terms of execution time and size are already enough to justify its usage.

Our proposed classifier is currently designed to recognize diseases in only one type of crop. However, due to the low size of the model, we can create several models for disease detection for each crop type and load them on mobile devices. In addition, the results obtained for tomato crop diseases reveal that it could be better to focus our efforts on one crop at a time, adjusting the solution for each crop. The main effort to do this is to create datasets for each type of crop that include enough healthy and sick plant leaves.

As future work, we propose to improve our custom model implemented, by changing the CNN structure and hyper-parameters. Additionally, we propose to divide the disease detection into two stages: discriminating between healthy and infected, and when detecting that there is a disease present, running the classification to determine the type of disease.

We also plan to build a mobile application to carry out the detection in real time with the support of farmers. However, we consider that all solutions created for farmers must also be analyzed from the perspective of user experience to guarantee their usability and accessibility. Intelligent systems as the one we intend to create, having the farmer and its conditions in mind, can be part of our academic contribution to agriculture sustainability.

Acknowledgements This work was partially supported by the Research Center for Communication and Information Technologies (CITIC), the School of Computer Science and Informatics (ECCI), and the Fabio Baudrit Moreno Agricultural Experimental Station (EEAFBM) at the University of Costa Rica. Research Project No. 834-B9-189. We are grateful to G.B.S., M.A.C., and R.Z.M. for their contributions in this research.
References

1. Secretaría Ejecutiva de Planificación Sectorial Agropecuaria (SEPSA), Desempeño del Sector Agropecuario. 3, 1–10 (2018) Available at: https://www.sepsa.go.cr/docs/2019-003_Desempeño_Sector_Agropecuario_2018.pdf

2. J.E. Monge Pérez, M. Lória Coto, Producción de chile dulce (Capsicum annuum) en invernadero: efecto de densidad de siembra y poda”. Posgrado Y Sociedad. Revista Electrónica Del Sistema De Estudios De Posgrado, 16(2), 19–38 (2018). https://doi.org/10.22458/tpys.v16i2.2269

3. R. Martarrita, J. Aguilar, L. Meza, Adaptabilidad de seis cultivares de chile dulce bajo invernadero en Guanacaste. Alcances Tecnológicos. 12(1), 13–23 (2018). ISSN-1659–0538

4. J. Rojas, M. Castillo, Planeamiento de la agro-cadena del tomate en la región central sur de Costa Rica (Puriscal, Costa Rica 2017). Disponible en https://www.mag.go.cr/bibliotecavirtual/E70-4158.pdf

5. C. Ramírez, J. Nienhuis, Cultivo protegido de hortalizas en Costa Rica. Tecnología En Marcha. 25(2), 10–20 (2012)

6. S. Abarca, Análisis y comentario: Cambio climático y plagas en el trópico. Alcances Tecnológicos. 12(1), 59–65 (2018). ISSN-1659–0538.

7. D.I. Tsitsigiannis, P. P. Antoniou, S. Tjamos, Major diseases of tomato, pepper and eggplant in greenhouses. in The Fruiting Species of the Solanaceae, ed by H. Passam. European J. Plant Sci. Biotechnol. 2(Special Issue 1), 106–124 (2008)

8. J.A. Brenes, A. Martínez, C.Y. Quesada-López, M. Jenkins, Sistemas de apoyo a la toma de decisiones que usan inteligencia artificial en la agricultura de precisión: un mapeo sistemático de literatura (Revista Ibérica De Sistemas y Tecnologías de Información RISTI 2020). https://doi.org/10.17013/risti.n.pi-pf

9. A. Dwivedi, R.K. Naresh, R. Kumar, R. Yadav, R. Kumar, Precision agriculture (2017)

10. R. Bongiovanni, J. Lowenberg-DeBoer, Precision agriculture and sustainability. Precis Agricul 5, 359–387 (2004). https://doi.org/10.1023/B:PRAG.0000040806.39604.aa

11. G. Davis, W. Casady, R. Massey, Precission Agriculture: An Introduction. Available at https://extension2.missouri.edu/wq450#:~:text=Precision%20agriculture%20merges%20with%20precision%20agriculture%20within%20a%20farm%20field (1998)

12. Oficina de las Naciones Unidas contra la Drog y el Delito (UNODC), Manual para el productor: El cultivo de hortalizas en Proyecto manejo integral de los recursos naturales en el trópico de Chacabamba y las Yungas de la Paz. Primera edición (La paz, Bolivia 2017)

13. Ministerio de Agricultura y Ganadería (MAG) de Costa Rica, Agrocadena Regional Cultivo CHILE DULCE. (Grecia, Costa Rica 2007). Disponible en https://www.mag.go.cr/bibliotecavirtual/E70-4281.pdf

14. J. Bolaños, L. Barrantes C.Y. Echandi, K. Bonilla, Manual técnico basado en experiencias con el híbrido ‘Dulcítico’ (Capsicum annuum). (San José, Costa Rica Instituto Nacional de Innovación y Transferencia en Tecnología Agropecuaria (INTA) 2018). 978–9968–586–34–4

15. Instituto Nacional de Innovación y Transferencia Tecnológica Agropecuaria (INTA), Manual técnico del cultivo de tomate (Solanum Lyco-persicum) (San José, Costa Rica 2017). ISBN 978–9968–586–27–6. Available at https://www.mag.go.cr/bibliotecavirtual/F01-10921.pdf

16. Organización de las Naciones Unidad para la Alimentación y la Agricultura, El cultivo de tomate con buenas prácticas agrícolas en la agricultura urbana y periurbana (2013). Disponible a https://www.fao.org/3/a-i3359s.pdf

17. H.Y. Thurston, J. Galindo, Enfermedades de cultivos en el trópico”. Centro Agronómico Tropical de Investigación y Enseñanza (CATIE). Programa de Mejoramiento de Cultivos Tropicales, con permiso de la American Phytopathological Society (Turrialba, Costa Rica 1989). ISBN 9977–57–072–8.

18. Australian Goverment, Sweet Pepper IPM Factsheet (2017). Disponible en https://ahr.com.au/wp-content/uploads/2017/11/Sweet-pepper-IPM-factsheet-20th-August-2017.pdf
19. B.S. Ghyar, G.K. Birajdar, Computer Vision-Based Approach to Detect Rice Leaf Diseases Using Texture and Color Descriptors. in 2017 International Conference on Inventive Computing and Informatics (ICICI) (Coimbatore, 2017), pp. 1074–1078. https://doi.org/10.1109/ICICI.2017.8365305

20. S. Phadikar, J. Sil, Rice Disease Identification Using Pattern Recognition Techniques. in 2008 11th International Conference on Computer and In-formation Technology (Khulna, 2018), pp. 420–423. https://doi.org/10.1109/ICCITECHN.2008.4803079

21. J.N. Reddy, K. Vinod, A.S. Remya Ajai, Analysis of Classification Algorithms for Plant Leaf Disease Detection. in 2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT) (Coimbatore, India, 2018), pp. 1–6. https://doi.org/10.1109/ICECCT.2019.8869090

22. N. Ibraheem, M. Hasan, R. Khan, P. Mishra, Understanding color models: a review. ARPN J. Sci. Technol. 2 (2012)

23. L. Armi, S. Fekri Ershad. Texture image analysis and texture classification methods—a review. ARPN J. Sci. Technol. 2 (2019)

24. R. Haralick, K. Shanmugam, I. Dinstein, Textutal features for image classification. IEEE Trans. Syst. Man and Cybern. SMC 3(6), 610–621 (1973). United States

25. M. Rhouma, S. Zafer, R. Khan, M. Hussain, Mohammad, Improving the performance of hu moments for shape recognition. Int. J. Appl. Environ. Sci (2014)

26. H. Sabrol, K. Satish, Tomato Plant Disease Classification in Digital Images Using Classification Tree, in 2016 International Conference on Communication and Signal Processing (ICCSPP) (Melmaruvathur, 2016), pp. 1242–1246. https://doi.org/10.1109/ICCSLP.2016.7754351

27. R.G. de Luna, E.P. Dadios, A.A. Bandala, Automated Image Capturing System for Deep Learning-based Tomato Plant Leaf Disease Detection and Recognition. in TENCON 2018—2018 IEEE Region 10 Conference (Jeju, Korea (South), 2018), pp. 1414–1419. https://doi.org/10.1109/TENCON.2018.8650088

28. H. Wang, G. Li, Z. Ma, X. Li, Image Recognition of Plant Diseases Based on Backpropagation Networks. in 2012 5th International Congress on Image and Signal Processing (Chongqing, 2012), pp. 894–900. https://doi.org/10.1109/CISP.2012.6469998

29. H. Wang, G. Li, Z. Ma, X. Li, Application of Neural Networks to Image Recognition of Plant Diseases. in 2012 International Conference on Systems and Informatics (ICSAI2012) (Yantai, 2012), pp. 2159–2164. https://doi.org/10.1109/ICSAI.2012.6223479

30. A. Tharwat, T. Gaber, T.M. Awad, N. Dey, A.E. Hassanien, Plants Identification Using Feature Fusion Technique And Bagging Classifier. in The 1st International Conference on Advanced Intelligent System and Informatics (AISI2015), November 28–30, 2015, Beni Suef, Egypt (Springer, Cham, 2015), pp. 461–471

31. J. Chaki, N. Dey, L. Moraru, F. Shi, Fragmented plant leaf recognition: Bag-of-features, fuzzy-color and edge-texture histogram descriptors with multi-layer perceptron. Optik 181, 639–650 (2019)

32. A.L.P. Ocampo, E.P. Dadios, Mobile Platform Implementation of Lightweight Neural Network Model for Plant Disease Detection and Recognition. in 2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM) (Baguio City, Philippines, 2018), pp. 1–4. https://doi.org/10.1109/HNICEM.2018.8666365

33. N. Petrellis, Plant Lesion Characterization For Disease Recognition A Windows Phone Application. in 2016 2nd International Conference on Frontiers of Signal Processing (ICFSP) (Warsaw, 2016), pp. 10–14. https://doi.org/10.1109/ICFSP.2016.7802948

34. J. Francis, S. Anto, D. Dhas, B.K. Anoop, Identification of Leaf Diseases in Pepper Plants Using Soft Computing Techniques. in 2016 Conference on Emerging Devices and Smart Systems (ICEDSS) (Namakkal, 2016), pp. 168–173. https://doi.org/10.1109/ICEDSS.2016.7587787

35. S. Mangiafico, Scheirer–Ray–Hare Test. Available at https://rcompanion.org/handbook/F_14.html
36. N. Dey, G. Mishra, K. Jajnyaseni, S. Chakraborty, S. Nath, *A Survey Of Image Classification Methods and Techniques*. in 2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT, 2014). https://doi.org/10.1109/ICCICCT.2014.6993023

37. N. Thakur, D. Maheshwart, A review of image classification techniques. IRJET Int. Res. J. Eng. Technol. 4(11) (2017)

38. S. Albawi, T. Abed Mohammed, S. Alzawi, *Understanding of a Convolutional Neural Network* (2017). https://doi.org/10.1109/ICEngTechnol.2017.8308186

39. L.B. Iantovics, A. Gligor, M.A. Niazi, A.I. Biro, S.M. Szilagyi, D. Tokody, Review of recent trends in measuring the computing systems intelligence. BRAIN. Broad Res. Artif. Intell. Neurosci. 9(2), 77–94 (2018)