ISSN: 2317-0840

Deep learning classification of apple leaf diseases: comparison of neural networks

Renísio Bráulio Baldine[1†], Karina Vieira dos Santos Fonseca[1],
Eric Batista Ferreira[2]

[1] *Master student in Applied Statistics and Biometrics, Federal University of Alfenas, Brazil.*
[2] *Professor at the Department of Statistics, Federal University of Alfenas, Brazil.*

Abstract: *The production of apples is a significant segment of the global agricultural industry, often threatened by diseases and pests. This study investigates the use of convolutional neural networks (CNNs) to classify images of apple tree leaves, distinguishing between healthy leaves and those affected by rust and scab. The objective is to develop an approach for the early de- tection of these fungal diseases. High-resolution images were collected, considering variations in lighting, angles, and backgrounds. Eighteen pre-trained CNN architectures available in Keras were tested and evaluated using metrics such as accuracy, precision, recall, and F1-score. The EfficientNetV2B2 and DenseNet201 networks showed the best results, with an accuracy of 99%. To enhance classification performance, ensemble techniques were explored, including combining all networks and selecting only the most accurate ones. Although promising, challenges such as computational complexity and the need for real-time processing in practical applications remain. The findings demonstrate the potential of CNNs and ensemble methods in supporting early detection of diseases in apple orchards, providing valuable tools for producers to manage infestations more effectivelys.*

Keywords: *Deep learning; Convolutional neural networks; Apple trees; Rust; Scab; Ensembles.*

Introduction

The global production of apples is an important segment of the agricultural industry and plays a fundamental role in food supply and the global economy. According to the Food and Agriculture Organization of the United Nations (FAO, 2021), the global apple production in 2021/2022 was 86.5 million tons, with production concentrated in key regions worldwide. According to AtlasBig (2021), China is the world's largest apple producer, with an annual production of about 45 million tons. The country concentrates its production in mountainous regions of the northwest, such as Shaanxi and Shandong, which offer an ideal climate for fruit cultivation.

The United States follows with 4.6 million tons; Poland with 3.6 million tons; Turkey with 2.9 million tons; India with 2.8 million tons; Iran with 2.8 million tons; Italy with 2.5 million tons; Russia with 1.8 million tons; and France with 1.8 million tons. Brazil is the 13th largest apple producer in the world, with an annual production of about 1 million tons. National production is concentrated in the southern part of the country, in states such as Santa Catarina, Rio Grande do Sul, and Paraná.

Apple orchards face significant challenges and suffer millions of dollars in annual losses due to various biotic and abiotic factors. The continuous management of stress and the multi-year impacts of fruit tree loss are constant issues for producers. During the growing season, orchards are constantly threatened by a variety of insects, fungi, bacteria, and viral pathogens. The incidence and severity of these infections can lead to consequences ranging from unappealing cosmetic appearances, poor marketability, and low fruit quality, to significant reductions in yield or even complete loss of fruits or trees, resulting in enormous economic losses (THAPA, 2020).

Additionally, it is important to highlight that incorrect diagnoses can lead to the improper use of chemical products, resulting in either unnecessary or insufficient application. This

[†]Autor correspondente: renisio@msn.com

Manuscrito recebido em: 27/06/2024
Manuscrito revisado em: 03/12/2024
Manuscrito aceito em: 11/12/2024

scenario can result in the selection of resistant pathogen strains, which increases production costs, intensifies environmental and health impacts, and in more severe cases, can even lead to significant outbreaks. Modern high-density apple orchards, often composed of a few highly susceptible varieties, are particularly vulnerable to the rapid spread of pathogens. In critical situations, infestations can quickly spread throughout the orchard, resulting in considerable losses and, in some cases, even complete destruction of the planting (PEIL, 2009).

The early detection of pests and diseases is a fundamental aspect of efficient orchard and crop area management. The proper and timely implementation of pest and disease management programs directly depends on this detection, as early interventions can minimize the negative impacts caused by these agents on agricultural production. According to Gupta, Slawson e Moffat (2022), early detection of pests and diseases is one of the main strategies for integrated pest and disease management in agriculture. To achieve this goal, researchers rely on risk prediction models for diseases and pests. These models are based on detailed information about the incidence, severity, and timing of infections, as well as current and forecasted meteorological data. In the context of contemporary agriculture, this approach envisions a scenario where farmers can accurately predict when and where pests and diseases will attack their crops, allowing them to act quickly and precisely, preventing losses and ensuring abundant and healthy harvests. This method protects the environment and human health by reducing the use of pesticides.

Technologies in the Field

This reality is becoming possible thanks to the integration of various technologies such as satellite imagery, unmanned aerial vehicles (UAVs), geostatistics, and artificial intelligence (AI).

1. **Satellite Imagery for a Holistic View:** Bitemporal satellite images, combined with environmental and plant growth data, offer a comprehensive view of crop health. This technology allows for distinguishing different diseases and pests, enabling more precise and effective management (MA et al., 2019).

2. **Geostatistics for Pest Mapping:** Helps to map the spatial distribution of insect populations, facilitating the identification of areas at higher risk of infestation. With this information, pest control efforts can be directed to specific areas, optimizing resource use and minimizing environmental impact (SCIARRETTA; TREMATERRA, 2014).

3. **Aerial Monitoring with UAVs:** Equipped with advanced technology, UAVs can fly over crops, collecting valuable data on agricultural operations. By analyzing this data, parameters such as flight speed and altitude can be optimized, enhancing pest and disease control (WANG et al., 2019).

4. **The Power of Artificial Intelligence:** According to Ludovico et al. (2023), various models are used for time series forecasting, among which statistical models and those based on computational intelligence stand out. For example, AI, with its deep learning algorithms, can analyze images of healthy and diseased plants, accurately identifying the crops and diseases present (MOHANTY; HUGHES; SALATHE, 2016; ZHANG et al., 2023). This technology allows for early detection of pests and diseases, enabling immediate interventions to prevent production losses and ensure high-quality harvests.

Ongoing research explores new AI applications for the rapid and accurate diagnosis of plant diseases, contributing to more efficient and sustainable agriculture (JUNIOR; SANTOS; SAFADI, 2019). Machine learning models, such as Convolutional Neural Networks (CNNs) are trained with large datasets of images to identify diseases in apples with high precision (SILVA, 2021). CNNs are designed for computer vision tasks like pattern recognition in images. They

are widely used for tasks such as image classification, object detection, semantic segmentation, and more.

Various studies have explored the potential of CNNs for diagnosing diseases in apple trees. Mohanty, Hughes e Salathe (2016) proposed a smartphone-assisted diagnostic system based on CNNs trained on a large dataset of leaf images from various crops, including apple trees. Fu et al. (2022) developed a model called AlexNet to classify leaf diseases in apple trees, demonstrating high accuracy. Aiming for robustness and efficiency, they proposed lightweight CNNs for apple disease detection. Wang et al. (2021) introduced enhanced CNNs with attention mechanisms to improve accuracy and efficiency in identifying various apple diseases. Turkoglu, Hanbay e Sengur (2019) integrated multi-model CNNs based on LSTM (Long Short-Term Memory) for the detection of diseases and pests in apples, demonstrating the versatility of neural networks in agricultural applications. These studies collectively highlight the significant advances made in using neural networks, especially deep learning models like CNNs, for real-time detection and diagnosis of diseases in apple trees.

Despite the potential of these algorithms, it is important to highlight the need for a comparative analysis to determine which models perform best in specific scenarios. The literature still lacks a specific comparison for this context. This work aims to fill this gap by conducting a comprehensive comparison of 18 deep neural networks that demonstrated the best results in the ImageNet competition (SZEGEDY et al., 2015; HE et al., 2016). The selection of these models is based on their recognized performance in one of the most renowned and challenging competitions in the field of computer vision. We hope to provide valuable insights that guide the choice and implementation of the most effective algorithms for monitoring and controlling diseases in apple orchards.

Images of Apple Tree Diseases

The collection and annotation of a large number of real, high-quality images by experts are crucial for training AI models with high precision (SILVA et al., 2020). This includes different image capture conditions, such as positions and angles of infected tissue, levels of ambient light, types of sensors, and climatic variations. Additionally, it is important to illustrate the effect of each disease on fruits and leaves at various stages. The diseases focused on in this work are Apple Scab and Apple Rust: Apple scab is caused by the fungus Venturia inaequalis. This disease causes the formation of dark, rough spots on the leaves and fruits of the apple tree. As the infection progresses, the spots may merge, covering large areas of the leaf surface or fruit. Besides the undesirable aesthetic aspect, apple scab can lead to premature leaf drop, impairing tree development and reducing harvest yield (AGROLINK, 2020; CULTIVAR, 2018; CULTIVAR, 2019).

On the other hand, rust is another common fungal disease that affects apple trees, caused by fungi of the genus Gymnosporangium. Rust is characterized by the formation of small orange spots on the underside of the leaves and on the fruits of the tree. Over time, these spots may become more visible and, in severe cases, can cause deformations in the leaves and premature leaf drop (TECHINFUS, 2021). In addition to scab and rust, other leaf diseases can affect apple trees, including mildew, bitter rot, trunk cancer, and gray mold (SILVA et al., 2010).

Pre-trained Neural Networks

Pre-trained CNNs are convolutional neural networks that have been trained on large datasets for specific tasks, such as image classification into a wide range of categories. These models are pre-trained on massive datasets, such as ImageNet, which contains millions of labeled images across various categories. Alsuwat et al. (2022) demonstrated the effectiveness of this method. These architectures consist of convolutional layers, fully connected layers, and can be fine-tuned through supervised training, as observed by Hu et al. (2015). The use of pre-trained

CNNs proves particularly valuable when labeled training data is scarce, enabling performance enhancement without incurring overfitting, as discussed by Girshick et al. (2014) and Wang et al. (2017).

These architectures are adaptable for different tasks, either by modifying their structures or by fine-tuning with new datasets, as evidenced by Tajbakhsh et al. (2016). In other words, the flexibility in modifying the architectures' structures allows for specific adjustments for different contexts and task requirements. These modifications may include adjustments to hyperparameters, adding additional layers, removing unnecessary layers, or even replacing fundamental components. Such adaptability enables the models to be optimized for a diverse range of machine learning problems, from image classification to object detection or semantic segmentation. Furthermore, their application extends to areas such as remote sensing, medical imaging, and object detection, leveraging the knowledge gained to optimize performance, as exemplified by Vishnoi, Kumar e Kumar (2021) and Cheng e Malhi (2016). The central proposition of this work is to leverage pre-trained neural networks to classify new images; in this case, the input data would be apple leaf diseases, adjusting the parameters and retraining only the final layers of the network. Thus, we can use this approach to specifically classify leaf diseases in our images, focusing on the object of interest: the identification of leaf diseases.

Ensemble Neural Network Combination

In the pursuit of excellence in machine learning, data scientists often resort to a strategy known as ensemble neural networks. This intelligent approach involves combining multiple neural networks, each contributing its own perspective and expertise, to produce more robust and accurate results.

For example, Cruz (2023) proposed an ensemble approach for sentiment analysis of tweets related to sports concussions, demonstrating the effectiveness of combining different deep neural network models. Similarly, Opitz e Maclin (1999) emphasized that ensembles consist of individually trained classifiers, such as neural networks, whose predictions are combined to classify new instances, and it was shown that this approach reduces overfitting and increases the overall model accuracy. Additionally, Lee, Hong e Kim (2009) demonstrated that the generalization ability of neural network systems can be significantly improved by combining multiple neural networks into an ensemble. Furthermore, ensemble neural networks have been applied in various fields. By leveraging the diversity and complementarity of individual models, ensembles have the potential to enhance the performance and reliability of machine learning systems, thus driving the frontier of knowledge and innovation.

Materials and Methods

Data Collection

During the 2019 growing season, real and high-quality RGB images—color images that capture red, green, and blue channels to represent realistic visual details—were captured, encompassing various symptoms of apple tree leaf diseases. As described on the competition website Kaggle (2020), these images were obtained from different commercially grown cultivars in an unsprayed apple orchard located at Cornell AgriTech in Geneva and New York, USA. The photos were taken using a Canon Rebel T5i DSLR camera, as well as smartphones, covering a wide range of lighting conditions, capture angles, scenes, and noise levels. This image collection procedure was carefully conducted to ensure the representativeness and diversity of disease symptoms observed on apple tree leaves throughout the growing period. The dataset used in this research presented complex challenges, which were carefully considered to make the analysis more comprehensive and representative. These complexities included:

1. Diverse image backgrounds: The captured images contained different backgrounds, which could hinder the accurate identification of disease symptoms. Pre-processing strategies were employed to remove background noise and highlight relevant elements.

2. Variation in capture times: The images were taken at different times of the day, which can affect lighting conditions and symptom appearance. This requires the use of normalization techniques and color correction to ensure information consistency.

3. Plant maturity stages: The images included plants at different maturity stages, which can affect symptom appearance and development. It was necessary to consider these variations to create robust classification models.

4. Presence of multiple diseases in a single image: Some photos displayed more than one disease, complicating the task of identification and classification. Multi-label detection approaches were adopted to deal with this situation.

5. Varied focus settings: The images were taken with different focus settings, which can impact the sharpness of details. Image enhancement techniques were applied to improve symptom quality and clarity.

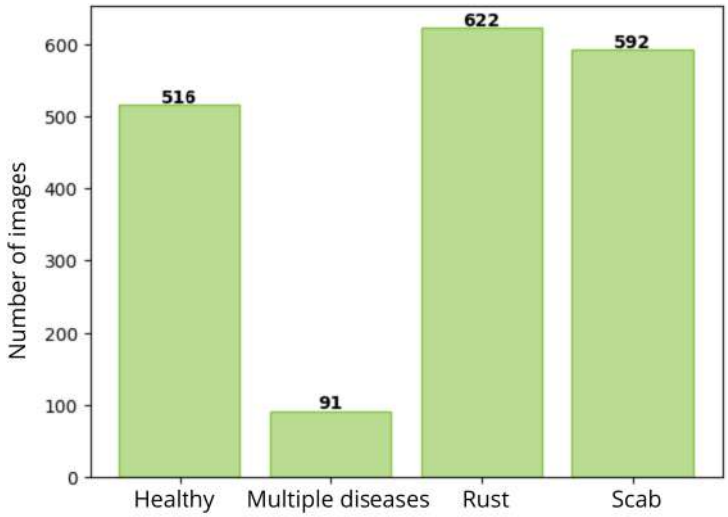
The dataset included diseases such as apple scab, cedar rust, Alternaria leaf spot, and frog eye leaf spot, along with healthy leaves, all of which were analyzed in detail. The careful handling of these complexities resulted in a reliable and representative dataset, providing a solid foundation to advance in the field of automated disease detection and diagnosis in plants.

Class Labeling and Data Splitting for Training and Testing

A Python development environment was set up, including the installation of necessary libraries for image processing and machine learning, such as TensorFlow and Keras (TENSORFLOW). The data used in this study was obtained from two CSV files available on Kaggle (2020). These files contained information about each plant image, identifying which disease the image belonged to. The four columns present in the files were: ``Healthy'', ``Multiple Diseases'', ``Rust'', and ``Scab''. Additionally, the database also contained the actual images associated with these classifications. Initially, the images were separated based on the information present in the CSV files. Each image was associated with one of the four available disease categories: ``Healthy'', ``Multiple Diseases'', ``Rust'', and ``Scab''. Thus, four distinct folders were created, each containing labeled images according to their respective disease (Figure 1).

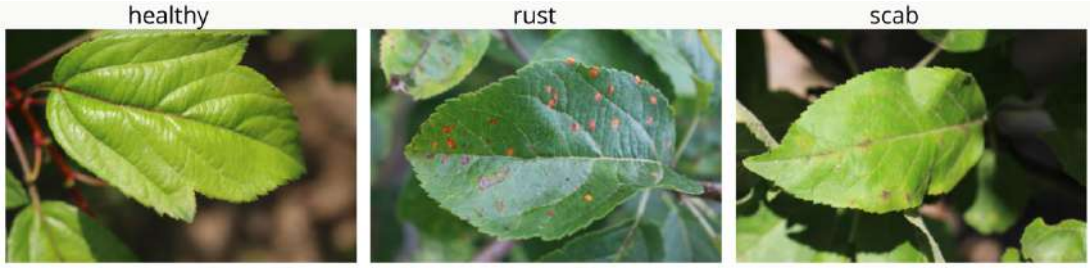
An exploratory data analysis was conducted to understand the distribution of samples in each class. For this, we used the provided quantity values, representing the number of images in each category. It was discovered that the ``Healthy'', ``Rust'', and ``Scab'' classes have a significantly larger number of images compared to the ``Multiple Diseases'' class, which has a substantially smaller representation (Figure 1). This imbalance can lead to problems during the classification model training, as the neural network may become biased towards the majority classes, impairing the correct identification of samples from the minority classes. Due to the observed data imbalance, we chose to use a training strategy focused on the classes of greater interest: ``Rust'', ``Scab'', and ``Healthy''. This is because the early detection of ``Rust'' and ``Scab'' diseases, as well as the identification of ``Healthy'' leaves, are of great relevance in agriculture. Therefore, we aim to improve the classification capability of these three specific classes (Figura 2).

Figure 1: Quantity of images per label.



Source: Adapted from Kaggle (2020).

Figure 2: Quantity of images per label.



Source: Adapted from Kaggle (2020).

To assess the model's performance, the dataset was divided into training and testing partitions using an 80-20 ratio. This standard approach in machine learning separates the data into two mutually exclusive sets: one for training the model and the other for testing it. The training set, containing 80% of the data, is used to adjust the weights of the neural network during training. Meanwhile, the test set, comprising the remaining 20%, is used to assess how well the trained model generalizes to entirely new data, of which it hasn't seen examples during weight adjustment. Evaluation on the test set provides an unbiased estimate of the model's generalization ability to make predictions on unknown data. It also helps identify whether overfitting occurred during training if the test performance is substantially worse than the training performance (Table 1).

Table 1: Training and testing data.

Variable	Number of Images	Height	Width	Number of Colors
Training Images X	1384	224	224	3
Training Images y	1384	-	-	3
Testing Images X	346	224	224	3
Testing Images y	346	-	-	3

Source: from the authors (2024).

1. Training Images variable X (1384, 224, 224, 3): This is the variable that contains the training images. The first number 1384 refers to the total number of images in the training set. The next three numbers 224, 224, 3 refer to the dimensions of each image: they are images with 224 pixels in height, 224 pixels in width, and 3 color channels (RGB).

2. Training Images variable y (1384, 3): This is the variable that contains the labels (classes) corresponding to the training images. There are 1384 labels, one for each image. Each label is represented by a one-dimensional array of length 3, where each position corresponds to one of the 3 classes.

3. Testing Images variable X (346, 224, 224, 3): This variable contains the test images, in the same format as the training images. There are 346 test images, each with dimensions of 224 x 224 pixels and 3 RGB color channels.

4. Testing Images variable y (346, 3): This variable contains the labels of the test images, in the same format as the training labels. There are a total of 346 labels, each with a length of 3 representing one of the 3 possible classes.

Neural Networks

In this study, eighteen pre-trained neural network architectures available in the Keras library for Python (TENSORFLOW) were employed to perform image classification tasks. Keras provides state-of-the-art implementations in deep learning, allowing for the construction and rapid training of deep neural models. The eighteen pre-trained architectures were selected for their ability to extract generic features from massive datasets (IMAGENET) (Table 2). This enables transfer learning and quick convergence even on smaller datasets.

Table 2: Models and their parameters.

No.	Model	Parameters	No.	Model	Parameters
1	Xception	22.9M	10	DenseNet121	8.1M
2	VGG16	138.4M	11	DenseNet169	14.3M
3	VGG19	143.7M	12	DenseNet201	20.2M
4	ResNet50	25.6M	13	EfficientNetB0	5.3M
5	ResNet50V2	25.6M	14	EfficientNetB1	7.9M
6	InceptionV3	23.9M	15	EfficientNetB5	30.6M
7	MobileNet	4.3M	16	EfficientNetB6	43.3M
8	MobileNetV2	3.5M	17	EfficientNetB7	66.7M
9	NASNetMobile	5.3M	18	EfficientNetV2B2	10.2M

Source: from the authors (2024).

Combining ensemble neural networks

This article addresses the concept of ensemble in machine learning, highlighting its utility in improving the accuracy of neural network models. Ensemble is an approach that combines the predictions of multiple individual models to produce a more robust and accurate prediction, owing to the diversity of the models and their learning approaches, which can capture different aspects of the input data (KOVALENKO, 2018). For instance, the ensemble process can be implemented through averaging predictions, majority voting, or weighting individual predictions. Initially, all predictions from these 18 neural networks were combined, regardless of the accuracy of each one. By doing this, leveraging the diversity of approaches and learning from each

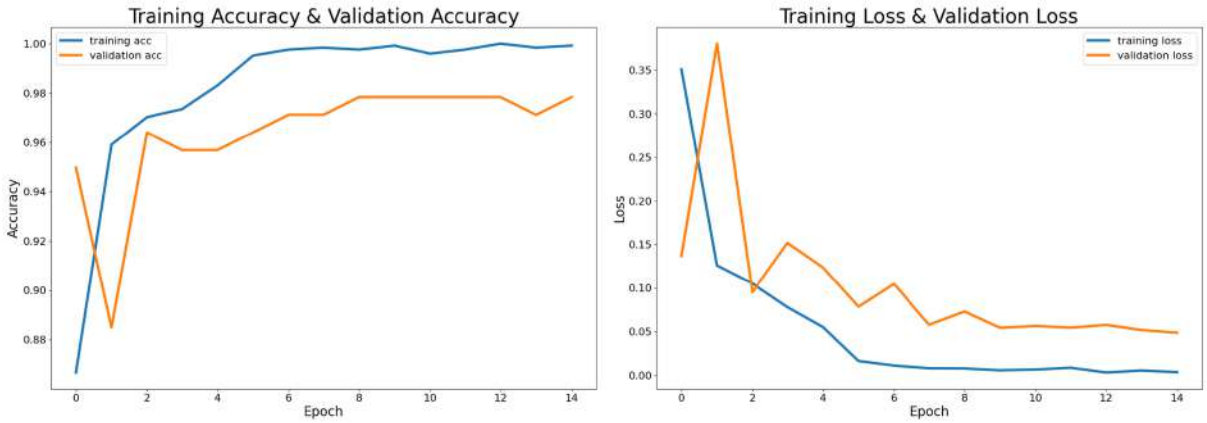
neural network to create a more robust and general prediction. In the second phase, aiming to ensure the most reliable neural networks, I will select the neural networks that achieved an accuracy of at least 95%. This means filtering out the neural networks that demonstrated a high level of accuracy in their individual predictions, combining only the predictions from these top-performing neural networks to form a new ensemble.

Results and Discussion

The training and validation graphs of a neural network were plotted, a common practice to evaluate the network's performance during training. The first graph shows the evolution of accuracy over the epochs for the training and validation sets. Accuracy is a measure of how correct the network's predictions are. It is expected that accuracy will increase during training, indicating that the network is making more accurate predictions. The graph allows for the evaluation of whether the network is suffering from overfitting (when the accuracy of the training set continues to increase, but the accuracy of the validation set starts to decrease). The second graph shows the evolution of loss over the epochs for the training and validation sets.

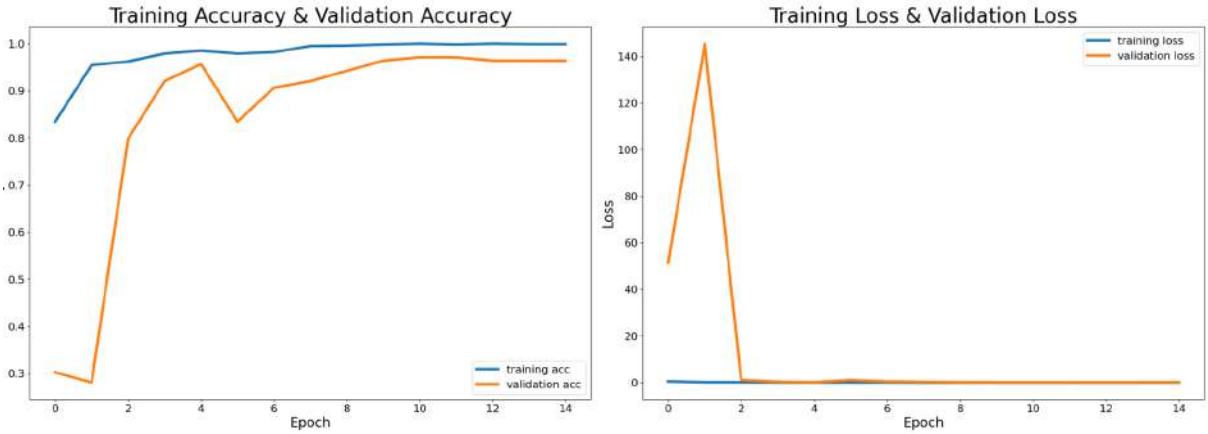
Loss is a measure of how far the network's predictions are from the correct values. It is expected that loss will decrease during training, indicating that the network is learning to make better predictions. The graph allows for the evaluation of whether the network is suffering from overfitting (when the loss of the training set continues to decrease, but the loss of the validation set starts to increase). In this section, we present the performance graphs of three powerful neural networks: EfficientNetV2B2 (Figure 3), InceptionV3 (Figure 4), and ResNet50V2 (Figure 5). For a more comprehensive analysis, including the graphs of the other 15 neural networks, see Appendix A.

Figure 3: The first graph shows the evolution of accuracy during training and validation of the neural network, and the second shows the evolution of loss during training and validation of the EfficientNetV2B2 neural network.



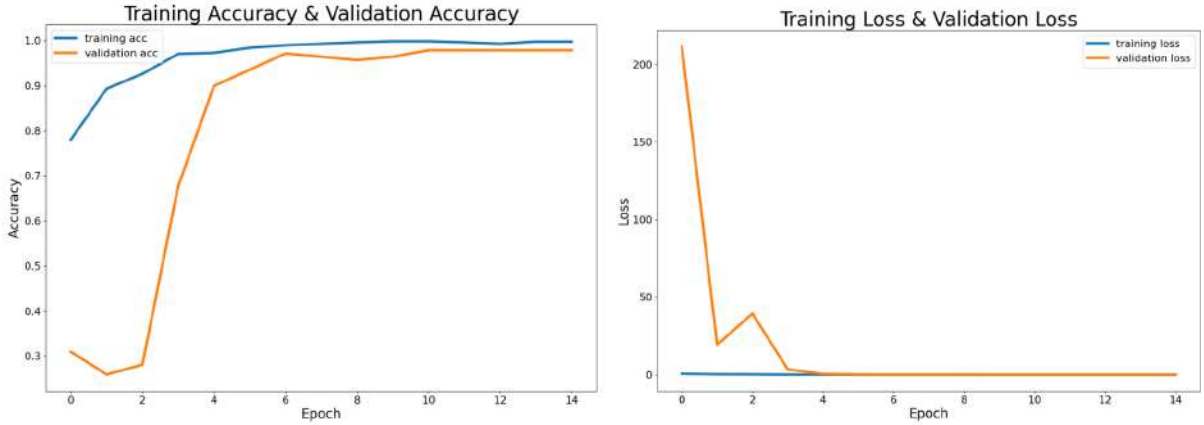
Source: from the authors (2024).

Figure 4: The first graph shows the evolution of accuracy during training and validation of the neural network, and the second shows the evolution of loss during training and validation of the InceptionV3 neural network.



Source: from the authors (2024).

Figure 5: The first graph shows the evolution of accuracy during training and validation of the neural network, and the second shows the evolution of loss during training and validation of the ResNet50V2 neural network.



Source: from the authors (2024).

The process of collecting and annotating a significant number of high-quality real images by experts is an absolutely crucial step for training AI models to achieve high accuracy (SILVA et al., 2020). The graphs indicate that most neural networks achieved consistent results, effectively avoiding overfitting. Overfitting is identified when training accuracy increases while validation accuracy declines. Additionally, loss is a key metric that indicates how closely the network's predictions match the correct values.

Confusion matrix of neural networks

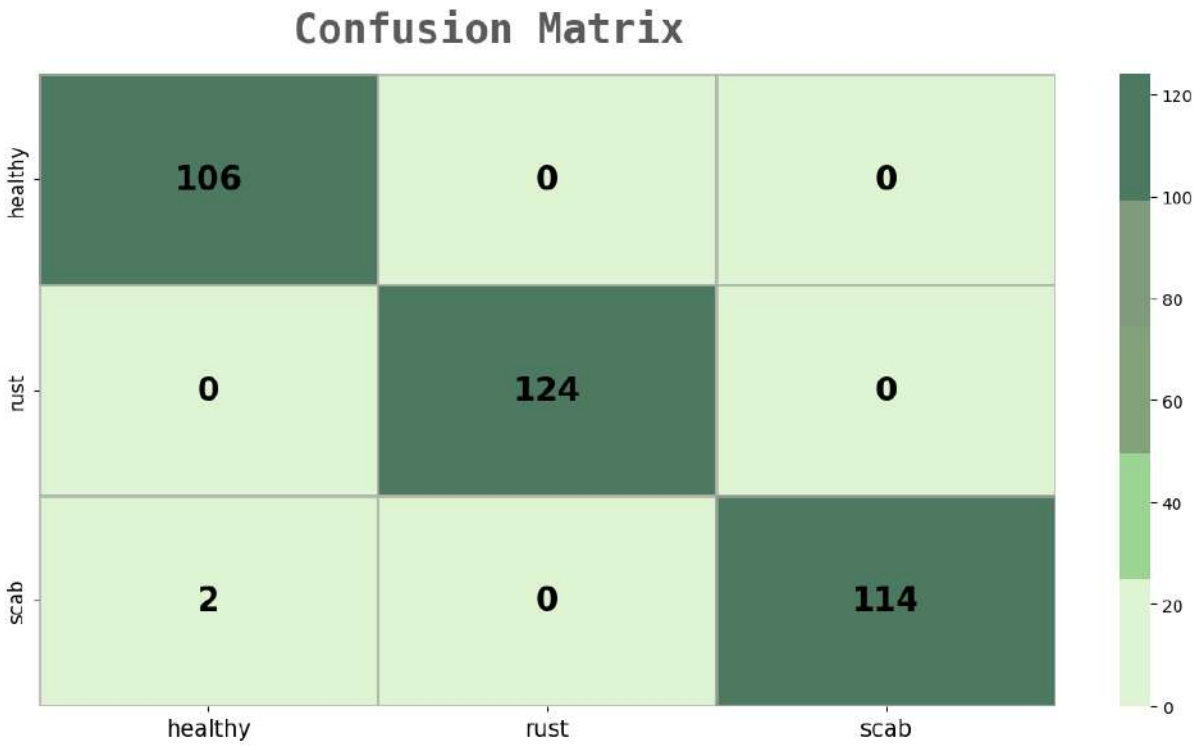
The confusion matrix plays a crucial role in evaluating an image classification model for apple tree leaves, distinguishing between healthy leaves and those affected by rust and scab diseases. This table provides a visual representation of the model's prediction accuracy for each class, allowing for a more precise analysis of overall performance and the effectiveness of the training process. Three classes were defined for the classification: ``healthy'', ``rust'', and ``scab''. In this context, the confusion matrix will have three rows and three columns corresponding to the ``healthy'', ``rust'', and ``scab'' classes. The values on the main diagonal represent the correct classifications for each class, while the values off the diagonal indicate incorrect classifications. For example, if the confusion matrix shows a high value in the cell corresponding to the ``healthy'' class and the ``healthy'' prediction, this would indicate that the model was effective in identifying healthy leaves. On the other hand, if there are significant values off the main diagonal, this could indicate confusion between the classes or areas where the model needs improvement. The performance metrics are precision, which is the proportion of samples that were correctly classified in each class, and accuracy, which is the total proportion of samples that were correctly classified. The results of the other networks can be found in Appendix B.

In the ``healthy'' class, 106 samples were correctly classified as ``healthy'', while no samples were incorrectly classified as ``rust'' or ``scab''. In the ``rust'' class, 124 samples were correctly classified as ``rust'', while no samples were incorrectly classified as ``healthy'' or ``scab''. In the ``scab'' class, 114 samples were correctly classified as ``scab'', while 2 samples were incorrectly classified as ``healthy'' (Figure 6).

In the ``healthy'' class, 102 samples were correctly classified as ``healthy,'' while 1 sample was incorrectly classified as ``rust'' and 3 samples as ``scab.'' In the ``rust'' class, 122 samples were correctly classified as ``rust,'' while 1 sample was incorrectly classified as ``healthy'' and 1 sample as ``scab.'' In the ``scab'' class, 110 samples were correctly classified as ``scab,'' while 6 samples were incorrectly classified as ``healthy.'' (Figure: 7).

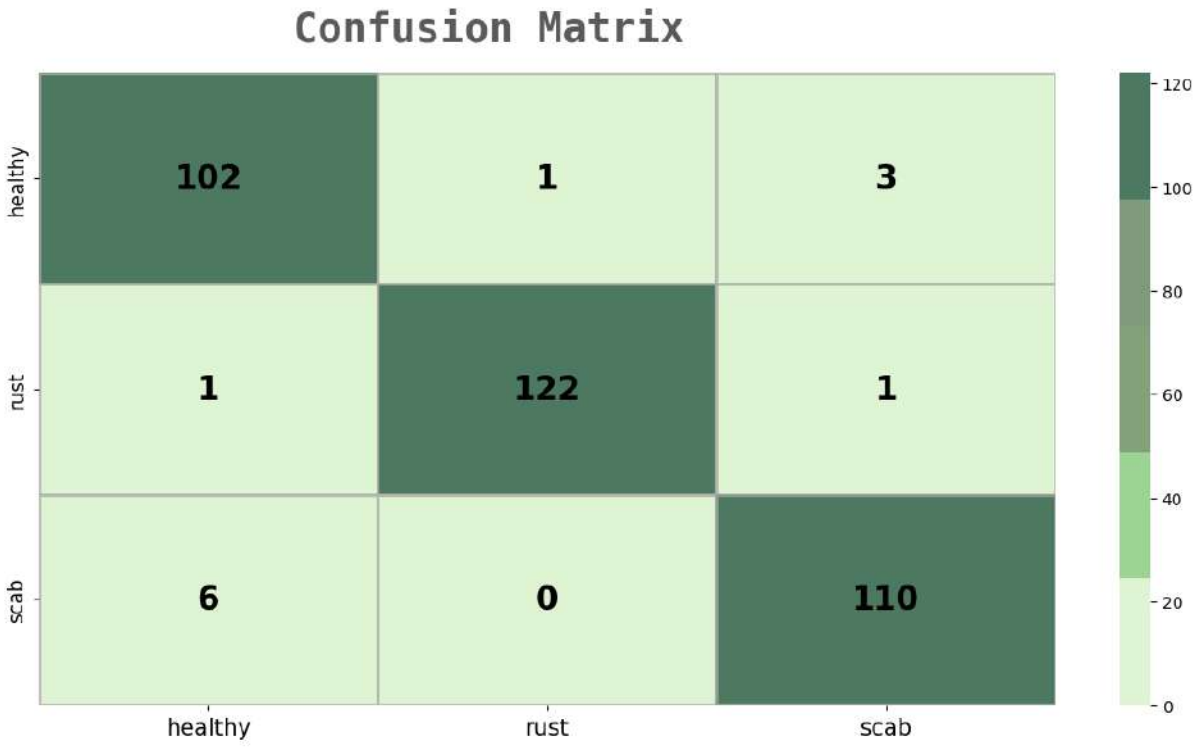
In the ``healthy'' class, 97 samples were correctly classified as ``healthy,'' while 6 samples were incorrectly classified as ``rust'' and 3 samples as ``scab.'' In the ``rust'' class, 122 samples were correctly classified as ``rust,'' while 3 samples were incorrectly classified as ``healthy'' and 1 sample as ``scab.'' In the ``scab'' class, 111 samples were correctly classified as ``scab,'' while 1 sample was incorrectly classified as ``healthy'' and 6 samples as ``rust.'' (Figure: 8).

Figure 6: Confusion Matrix for neural network EfficientNetV2B2.



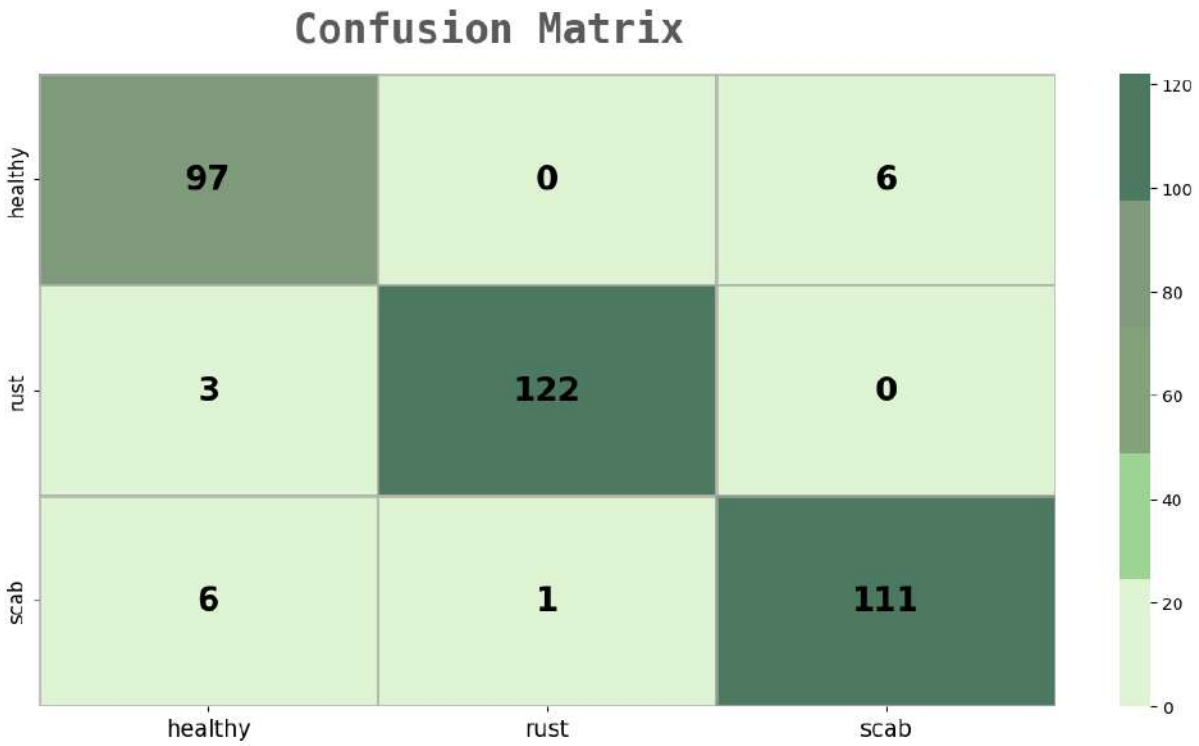
Source: from the authors (2024).

Figure 7: Confusion Matrix for neural network InceptionV3.



Source: from the authors (2024).

Figure 8: Confusion Matrix for neural network ResNet50V2.



Source: from the authors (2024).

Precision, recall, f1-score of neural networks

In Table 3, we find a detailed and comprehensive analysis of the results obtained through the application of eighteen neural networks in the task of classifying apple leaves. This analysis allows us to thoroughly evaluate the performance of the models used and identify areas for improvement. The metrics below, presented as abbreviations in the first row of each column of the table, provide valuable information about the quality and effectiveness of each model:

- M (p): Macro average (precision) represents the average precision for all classes. It assesses the model's ability to correctly predict positive instances without considering class imbalance.
- P (p): Weighted average (precision) is the average precision, taking into account the weight of each class. It provides a more complete view of performance, considering the class distribution in the dataset.
- M (r): Macro average (recall) is the average recall for all classes. It measures the model's ability to correctly identify positive instances without considering class imbalance.
- P (r): Weighted average (recall) represents the average recall, weighted by the weight of each class. This metric considers both detection capability and class distribution.
- M (f): Macro average (f1-score) is the harmonic mean of precision and recall for all classes. It offers a balanced view between the quality and quantity of predictions.
- P (f): Weighted average (f1-score) is the weighted harmonic mean of precision and recall, considering the weight of each class. It is useful in situations with class imbalance.

- Accuracy: The last column displays accuracy, which represents the proportion of instances correctly classified by the model. It provides an overall view of the model's performance across all classes (Table 4).

These evaluation metrics allow us to understand the efficiency and performance of each model comprehensively, addressing different aspects of apple leaf classification.

Table 3: Results of the neural network with precision, recall, f1-score, and accuracy metrics.

Model	M (p)	P (p)	M (r)	P (r)	M (f)	P (f)	Accuracy
EfficientNetB0	0.98	0.98	0.98	0.98	0.98	0.98	0.98
EfficientNetB1	0.98	0.98	0.98	0.98	0.98	0.98	0.98
EfficientNetB5	0.98	0.98	0.98	0.98	0.98	0.98	0.98
EfficientNetB6	0.85	0.86	0.85	0.85	0.85	0.85	0.85
EfficientNetB7	0.87	0.87	0.87	0.87	0.87	0.87	0.87
EfficientNetV2B2	0.99	0.99	0.99	0.99	0.99	0.99	0.99
VGG16	0.73	0.73	0.73	0.73	0.73	0.73	0.73
VGG19	0.76	0.76	0.76	0.76	0.76	0.76	0.76
Xception	0.93	0.93	0.92	0.92	0.92	0.92	0.92
InceptionV3	0.96	0.97	0.96	0.97	0.96	0.97	0.97
ResNet50	0.96	0.97	0.96	0.97	0.96	0.97	0.97
DenseNet169	0.98	0.98	0.98	0.98	0.98	0.98	0.98
DenseNet121	0.96	0.97	0.96	0.97	0.96	0.97	0.97
MobileNet	0.97	0.97	0.97	0.97	0.97	0.97	0.97
MobileNetV2	0.66	0.66	0.66	0.66	0.66	0.66	0.66
InceptionResNetV2	0.96	0.97	0.97	0.97	0.96	0.97	0.97
DenseNet201	0.98	0.99	0.99	0.99	0.99	0.99	0.99
ResNet50V2	0.95	0.95	0.95	0.95	0.95	0.95	0.95

Source: from the authors (2024).

The results of the experiments may indicate whether this technique can, or cannot, perform knowledge transfer, where the filters learned in a previous training are generic enough to be used in the classification of new image datasets. The adaptability of these architectures can be perceived both by the flexibility to modify their structures and by the efficient adjustment with new datasets, as highlighted by Tajbakhsh et al. (2016). The results of the experiments indicate that even when confronted with data unseen during initial training, these architectures maintain robust and reliable performance. This suggests that the models are capable of generalizing patterns learned in one context to another, demonstrating an effective knowledge transfer capability.

Table 4: Classification of neural networks by accuracy.

(Continued)

Order	Model	Accuracy
1	EfficientNetV2B2	0.99
2	DenseNet201	0.99
3	DenseNet169	0.98
4	EfficientNetB0	0.98
5	EfficientNetB1	0.98
6	EfficientNetB5	0.98
7	InceptionV3	0.97

(Continuation)

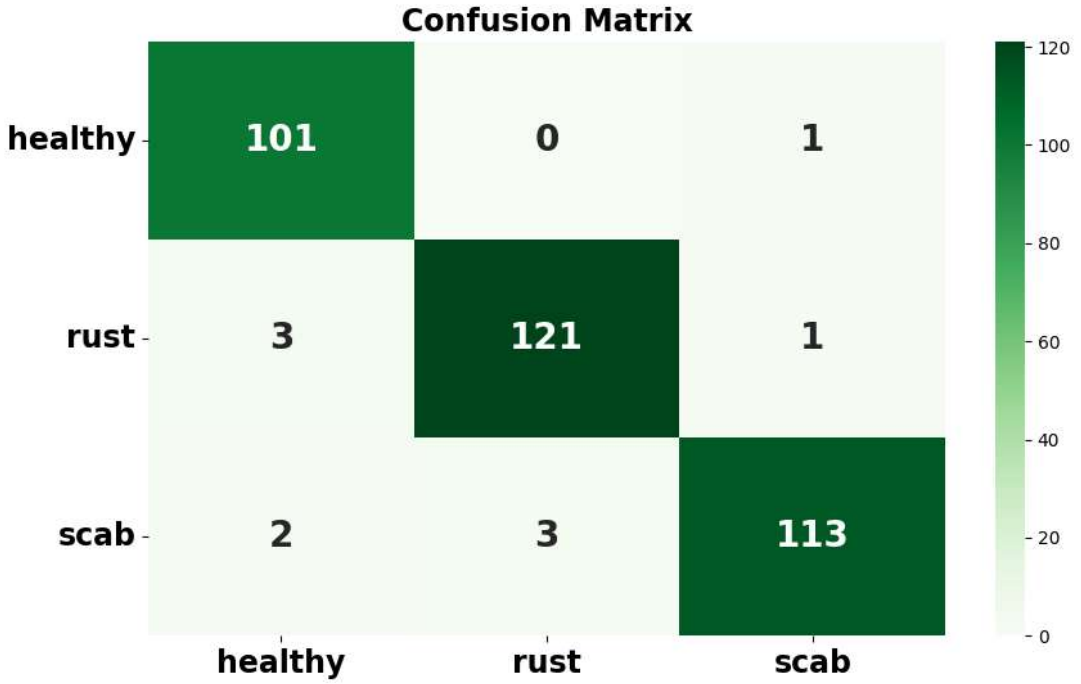
Order	Model	Accuracy
8	InceptionResNetV2	0.97
9	MobileNet	0.97
10	ResNet50	0.97
11	DenseNet121	0.97
12	ResNet50V2	0.95
13	Xception	0.92
14	EfficientNetB7	0.87
15	EfficientNetB6	0.85
16	VGG19	0.76
17	VGG16	0.73
18	MobileNetV2	0.66

Source: from the authors (2024).

Combining neural networks

Next, we will explore the concept of ensemble by voting, a strategy that combines predictions from multiple neural networks (Table 3). The goal is to evaluate the performance of this ensemble using metrics such as precision, recall, f1-score, and support, which are calculated for each class in question. These metrics provide a comprehensive analysis of the model's performance, comparing its predictions with the true labels. The results will be presented in an organized manner, in a table that simplifies the visualization and interpretation of the metrics. Initially, we proceed to create the confusion matrix by adopting an approach that involved combining all neural networks (Figure 9). This method resulted in an accuracy rate of 96.82%. Subsequently, we will present a comprehensive table detailing the precision, recall, f1-score, and support metrics for each class (Table 5). Through these values, it will be possible to perform a more in-depth analysis of the model's performance in different categories.

Figure 9: Confusion matrix of the total Ensemble-1.



Source: from the authors (2024).

In the ``healthy'' class, 101 samples were correctly classified as ``healthy'', while 1 sample was misclassified as ``rust'' and 1 sample as ``leaf spot''. In the ``rust'' class, 121 samples were correctly classified as ``rust'', while 3 samples were misclassified as ``healthy'' and 1 sample as ``leaf spot''. In the ``leaf spot'' class, 113 samples were correctly classified as ``leaf spot'', while 2 samples were misclassified as ``healthy'' and 3 samples as ``rust''.

As highlighted by Lee, Hong e Kim (2009), the generalization capability of neural network systems can be significantly enhanced by combining multiple neural networks into an ensemble. This approach not only pushes the frontier of knowledge and innovation but also demonstrates its applicability in various fields where accuracy and reliability are paramount. Therefore, the results obtained, shown in Figure 10 and Table 5, with the combination of the 18 neural networks, underscore their potential to make significant contributions to classification tasks and pattern recognition across a variety of domains.

Table 5: Metrics of ensemble-1 from all networks.

Class	Precision	Recall	F1-Score	Support
Healthy	0.95	0.98	0.97	103
Rust	0.97	0.97	0.97	125
Leaf spot	0.98	0.96	0.97	118

Source: from the authors (2024).

In the subsequent example, we will adopt the exclusive clustering approach of neural networks whose precision reached or exceeded the minimum threshold of 95%. This selection includes the following networks: EfficientNetB0, EfficientNetB1, EfficientNetB5, EfficientNetV2B2, InceptionV3, ResNet50, DenseNet169, DenseNet121, MobileNet, InceptionResNetV2, DenseNet201, and ResNet50V2, as detailed in Table 3. The underlying purpose of this strategy is to maintain performance using the smallest number of neural networks possible. Moving forward, we will present in detail the precision, recall, f1-score, and support metrics for each category (Table 6). These metrics will provide a thorough analysis of the resulting model's performance. With this specific approach, we achieved an accuracy rate of 98.41%.

Table 6: Metrics of ensemble-2 from all networks.

Class	Precision	Recall	F1-Score	Support
Healthy	0.96	0.99	0.98	103
Rust	1.00	0.99	1.00	125
Scab	0.99	0.97	0.98	118

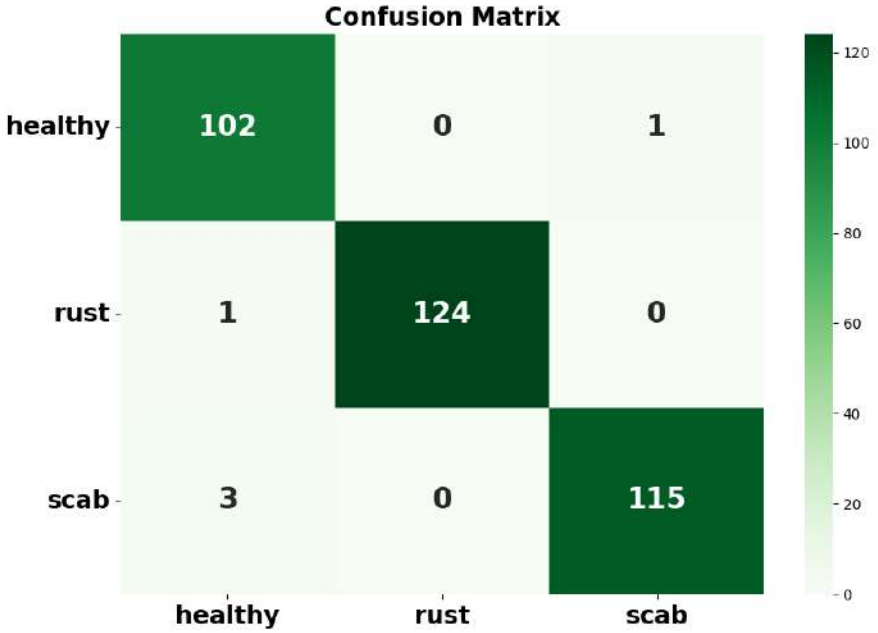
Source: from the authors (2024).

In the ``healthy'' class, 102 samples were correctly classified as ``healthy'', while 0 samples were misclassified as ``rust'' and 1 sample as ``scab''. In the ``rust'' class, 124 samples were correctly classified as ``rust'', while 1 sample was misclassified as ``healthy'' and 0 samples as ``scab''. In the ``scab'' class, 115 samples were correctly classified as ``scab'', while 3 samples were misclassified as ``healthy'' and 0 samples as ``rust''.

As observed in Figure 10 and Table 6, the results suggest that the new combination of neural networks, selected based on a minimum accuracy of 95%, resulted in even better performance in sample classification. This highlights the importance of carefully choosing individual models to compose the ensemble, aiming to improve the overall performance of the machine learning system. As mentioned by Opitz et al. (2019), ensembles consist of individually trained classifiers, such as neural networks, whose predictions are combined to classify new instances. It has been demonstrated that this approach reduces overfitting and increases the overall accu-

racy of the model. This ensemble strategy, by integrating multiple classifiers, not only enhances accuracy but also reduces the risk of overfitting, thus ensuring a more reliable generalization of the model to new data.

Figure 10: Confusion matrix of the Ensemble-2 of 12 neural networks.



Source: from the authors (2024).

Final Considerations

The analysis of neural network classification for detecting different categories of apple leaves—"healthy," "rust," and "scab"—revealed notable results. Table 4 presents the accuracies obtained by each neural network model, ranked by performance. Of the 18 neural network models evaluated, 12 achieved accuracy equal to or greater than 95%. Among these, the EfficientNetV2B2 and DenseNet201 models demonstrated the highest accuracy rates, reaching 99%. This performance underscores the effectiveness of these models in identifying the different conditions of apple leaves. Furthermore, the DenseNet169 and EfficientNetB0, B1, and B5 networks also showed high performance, with accuracies around 98%. These findings indicate that neural networks can classify apple leaves with high precision, providing valuable support for pest and disease management by enabling the early identification and treatment of affected trees.

On the other hand, models like VGG19, VGG16, and MobileNetV2 showed lower accuracies, ranging from 76% to 66%, likely due to limitations in capturing complex apple leaf features. These issues may also affect other models with lower performance. Potential improvements include fine-tuning with larger datasets, applying data augmentation, or integrating attention mechanisms to enhance feature extraction.

In this study, the initial strategy of combining neural networks yielded favorable results, though with slightly lower performance compared to the top-performing models (Table 5). This approach achieved an accuracy rate of 96.82%, providing insights into the synergy between models, identifying trends, and evaluating their impact on classifying apple leaves into categories such as ``healthy'', ``rust'', and ``scab''.

In the second stage, the methodology was refined to maximize effectiveness. An exclusive clustering process was implemented, selecting only neural networks with an accuracy of 95% or higher. This selection included models such as EfficientNetB0, EfficientNetB1, EfficientNetB5,

EfficientNetV2B2, InceptionV3, ResNet50, DenseNet169, DenseNet121, MobileNet, Inception-ResNetV2, DenseNet201, and ResNet50V2 (Table 6). This targeted selection provided a robust foundation for enhancing overall performance.

Analyzing metrics such as precision, recall, F1-score, and support for each category revealed that this refined approach significantly improved results. The final model achieved an accuracy rate of 98.41%. Although the initial combination of neural networks showed slightly lower performance, it provided valuable insights for strategy adjustments. The second stage, focused on high-performing networks, resulted in a more efficient classification model, emphasizing the importance of iterative refinement and continuous improvement.

References

AGROLINK. Sarna da maçã (venturia inaequalis). *Portal Agrolink*, 2020. Disponível em: <https://www.agrolink.com.br/problemas/sarna-da-maca_1692.html>.

ALSUWAT, M. et al. Prediction of diabetic retinopathy using convolutional neural networks. *International Journal of Advanced Computer Science and Applications*, v. 13, n. 7, 2022. Disponível em: <https://doi.org/10.14569/ijacsa.2022.0130798>.

ATLASBIG. *Principais países produtores de maçã*. 2021. Accessed on 2023-07-30. Disponível em: <https://www.atlasbig.com/pt-br/paises-por-producao-de-maca>.

CHENG, P.; MALHI, H. Transfer learning with convolutional neural networks for classification of abdominal ultrasound images. *Journal of Digital Imaging*, v. 30, n. 2, p. 234–243, 2016. Disponível em: <https://doi.org/10.1007/s10278-016-9929-2>.

CRUZ, A. D. Deep neural network approach for sentiment analysis of tweets related to sports concussion. 2023.

CULTIVAR. Sarna na maçã. *Revista Cultivar*, 2018. Disponível em: <https://revistacultivar.com.br/artigos/sarna-na-maca>.

CULTIVAR. Controle da sarna da macieira. *Revista Cultivar*, 2019. Disponível em: <https://revistacultivar.com.br/artigos/controle-da-sarna-da-macieira>.

FAO. *World apple production in 2021/2022 reaches 86.5 million tonnes*. 2021. Accessed on 2023-07-30. Disponível em: <https://www.fao.org/news/story/en/item/1425859/icode/>.

FU, L. et al. Lightweight-convolutional neural network for apple leaf disease identification. *Frontiers in Plant Science*, v. 13, 2022. Disponível em: <https://doi.org/10.3389/fpls.2022.831219>.

GIRSHICK, R. et al. Rich feature hierarchies for accurate object detection and semantic segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [s.n.], 2014. Disponível em: <https://doi.org/10.1109/cvpr.2014.81>.

GUPTA, N.; SLAWSON, D. D.; MOFFAT, A. J. Using citizen science for early detection of tree pests and diseases: perceptions of professional and public participants. *Biological Invasions*, v. 24, n. 1, p. 123–138, 2022.

HE, K. et al. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

HU, F. et al. Transferring deep convolutional neural networks for the scene classification of high-resolution remote sensing imagery. *Remote Sensing*, v. 7, n. 11, p. 14680–14707, 2015. Disponível em: <https://doi.org/10.3390/rs71114680>.

IMAGENET. *ImageNet Large Scale Visual Recognition Challenge (LSVRC)*. <https://www.image-net.org/challenges/LSVRC/>. Acesso em 1º de agosto de 2023.

JUNIOR, A. M. M.; SANTOS, P. R.; SAFADI, T. Utilização de redes neurais artificiais na classificação de danos em sementes de girassol. *Sigmae*, v. 8, n. 2, p. 564–575, 2019. Accessed: June 24, 2024. Disponível em: <https://publicacoes.unifal-mg.edu.br/revistas/index.php/sigmae/article/view/1046>.

KAGGLE. *Plant Pathology 2020 - FGVC7*. 2020. <https://www.kaggle.com/competitions/plant-pathology-2020-fgvc7/overview>. Accessed: 2023-07-31.

KOVALENKO, I. Redes neurais profundas (parte vi). ensemble de classificadores de redes neurais. *MQL5*, 2018. Disponível em: <https://www.mql5.com/pt/articles/4227>.

LEE, H.; HONG, S.; KIM, E. A new genetic feature selection with neural network ensemble. *International Journal of Computer Mathematics*, v. 86, p. 1105–1117, 2009.

LUDOVICO, S. N. et al. Agricultural commodity price prediction via machine learning algorithms. *Sigmae*, v. 11, n. 2, p. 45–69, 2023. Acesso em: 24 jun. 2024. Disponível em: <https://publicacoes.unifal-mg.edu.br/revistas/index.php/sigmae/article/view/1967>.

MA, H. et al. Integrating growth and environmental parameters to discriminate powdery mildew and aphid of winter wheat using bi-temporal landsat-8 imagery. *Remote Sensing*, v. 11, n. 7, p. 846, 2019. Disponível em: <https://doi.org/10.3390/rs11070846>.

MOHANTY, S.; HUGHES, D.; SALATHE, M. Using deep learning for image-based plant disease detection. *Frontiers in Plant Science*, v. 7, 2016. Disponível em: <https://doi.org/10.3389/fpls.2016.01419>.

OPITZ, D. W.; MACLIN, R. Popular ensemble methods: an empirical study. *Journal of Artificial Intelligence Research*, v. 11, p. 169–198, 1999.

PEIL, . W. A. The role of citizen science in the detection of invasive species. *Biological Invasions*, v. 11, n. 12, p. 2693–2707, 2009.

SCIARRETTA, A.; TREMATERRA, P. Geostatistical tools for the study of insect spatial distribution: practical implications in the integrated management of orchard and vineyard pests. *Plant Protection Science*, v. 50, n. 2, p. 97–110, 2014. Disponível em: <https://doi.org/10.17221/40/2013-pps>.

SILVA, J. et al. VisÃo computacional para detecÇÃo de doenÇas fÚngicas na agricultura. *Revista UNICA*, v. 1, n. 1, p. 1–10, 2020. Disponível em: <http://co.unicaen.com.br:89/periodicos/index.php/UNICA/article/view/67>.

SILVA, J. C. et al. Avaliaç ao de cultivares de milho para produç ao de silagem. *Revista Brasileira de Milho e Sorgo*, v. 9, n. 1, p. 6–7, 2010. Disponível em: <https://ainfo.cnptia.embrapa.br/digital/bitstream/item/200689/1/12688-2010-p.6-7.pdf>.

SILVA, J. V. d. *Estudo e an'alise de Redes Neurais Convolucionais para identificaç ao de doenças em folhas de macieira*. Dissertação (Mestrado) — Universidade de Bras'ılia, 2021. Disponível em: <https://repositorio.unb.br/handle/10482/43328>.

SZEGEDY, C. et al. Going deeper with convolutions. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

TAJBAKHSH, N. et al. Convolutional neural networks for medical image analysis: full training or fine tuning? *IEEE Transactions on Medical Imaging*, v. 35, n. 5, p. 1299–1312, 2016. Disponível em: <https://doi.org/10.1109/tmi.2016.2535302>.

TECHINFUS. Por que aparece ferrugem na macieira e o que fazer? *techinfus.com*, 2021. Disponível em: <https://ibuilder-pt.techinfus.com/yabloni/rzhavchina/>.

TENSORFLOW. *Instalação do TensorFlow com suporte a GPU (Português)*. <https://www.tensorflow.org/install/gpu?hl=pt-br>. Acesso em 1º de agosto de 2023.

THAPA, S. N. B. S. K. A. The plant pathology challenge 2020 data set to classify foliar disease of apples. *Applications in Plant Sciences*, v. 8, 2020.

TURKOGLU, M.; HANBAY, D.; SENGUR, A. Multi-model lstm-based convolutional neural networks for detection of apple diseases and pests. *Journal of Ambient Intelligence and Humanized Computing*, v. 13, n. 7, p. 3335–3345, 2019. Disponível em: <https://doi.org/10.1007/s12652-019-01591-w>.

VISHNOI, V. K.; KUMAR, K.; KUMAR, B. Plant disease detection using computational intelligence and image processing. *Journal of Plant Disease Protection*, v. 128, p. 19–53, 2021.

WANG, J. et al. Transferring pre-trained deep cnns for remote scene classification with general features learned from linear pca network. *Remote Sensing*, v. 9, n. 3, p. 225, 2017. Disponível em: <https://doi.org/10.3390/rs9030225>.

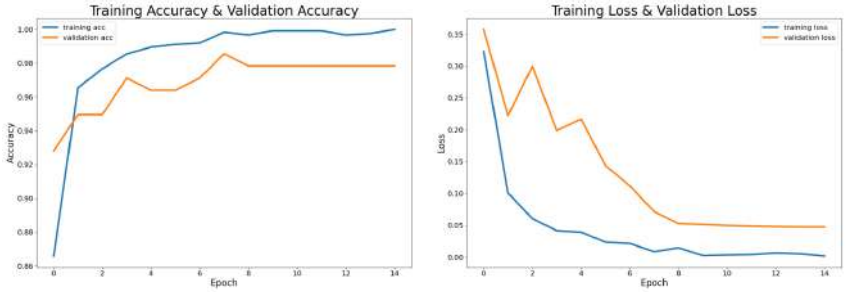
WANG, L. et al. Applications and prospects of agricultural unmanned aerial vehicle obstacle avoidance technology in china. *Sensors*, v. 19, n. 3, p. 642, 2019. Disponível em: <https://doi.org/10.3390/s19030642>.

WANG, P. et al. Identification of apple leaf diseases by improved deep convolutional neural networks with an attention mechanism. *Frontiers in Plant Science*, v. 12, 2021. Disponível em: <https://doi.org/10.3389/fpls.2021.723294>.

ZHANG, S. et al. Editorial: machine learning and artificial intelligence for smart agriculture, volume ii. *Frontiers in Plant Science*, v. 14, 2023. Disponível em: <https://doi.org/10.3389/fpls.2023.1166209>.

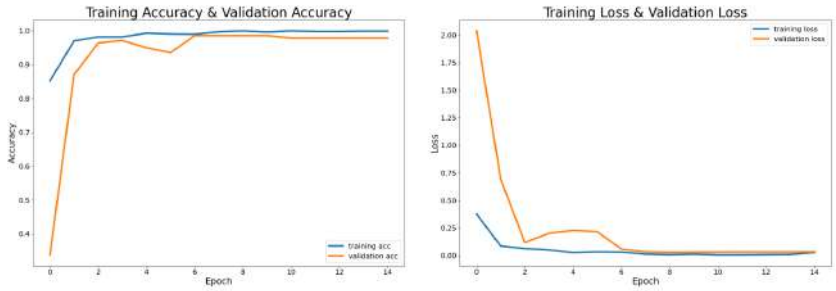
Appendix A - Training and Testing Graphs of Neural Networks

Figure 11: The first graph shows the evolution of accuracy during the training and validation of the neural network and the second shows the evolution of loss during the training and validation of the EfficientNetB0 neural network.



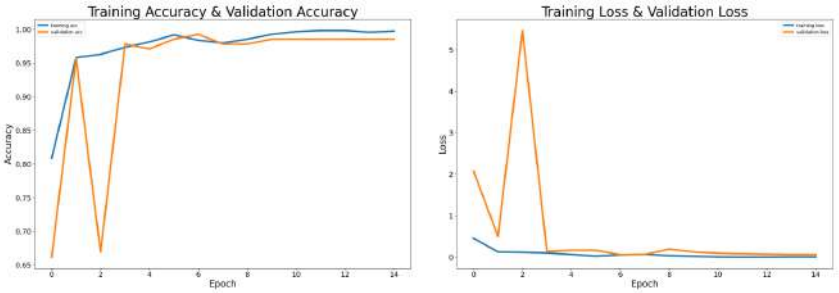
Source: Authors.

Figure 12: The first graph shows the evolution of accuracy during the training and validation of the neural network and the second shows the evolution of loss during the training and validation of the EfficientNetB1 neural network.



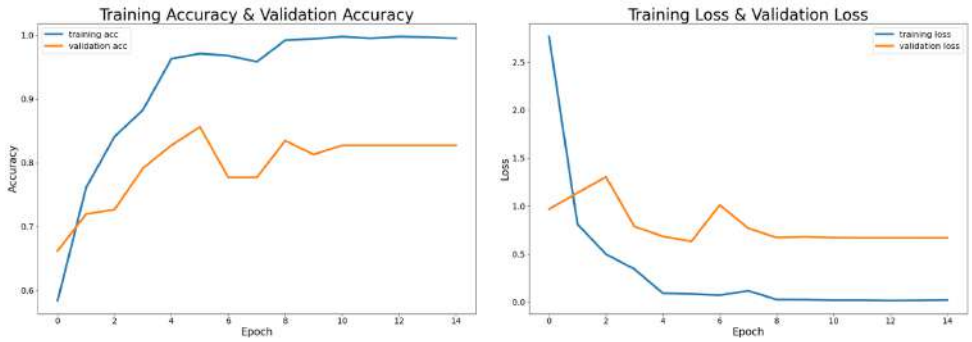
Source: Authors.

Figure 13: The first graph shows the evolution of accuracy during the training and validation of the neural network and the second shows the evolution of loss during the training and validation of the EfficientNetB5 neural network.



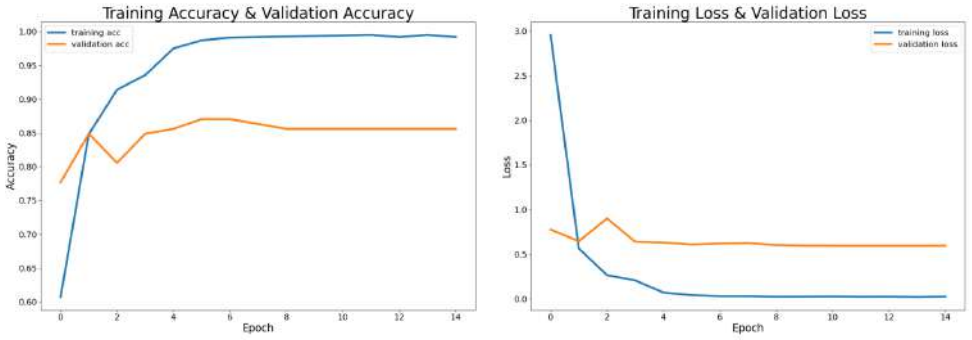
Source: Authors.

Figure 14: The first graph shows the evolution of accuracy during the training and validation of the neural network and the second shows the evolution of loss during the training and validation of the EfficientNetB6 neural network.



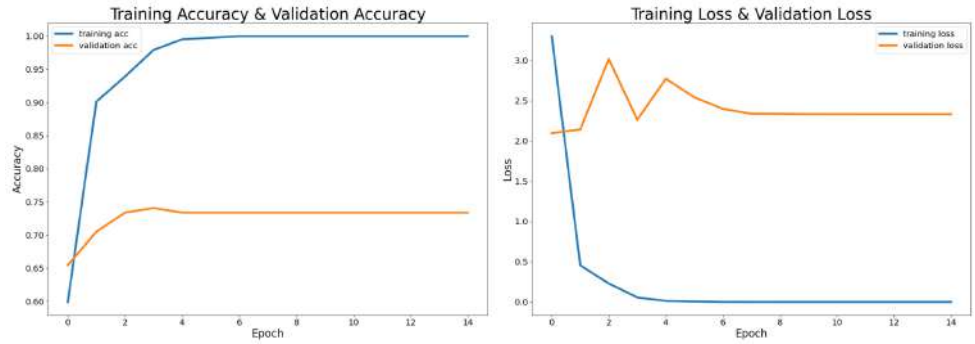
Source: Authors.

Figure 15: The first graph shows the evolution of accuracy during the training and validation of the neural network and the second shows the evolution of loss during the training and validation of the EfficientNetB7 neural network.



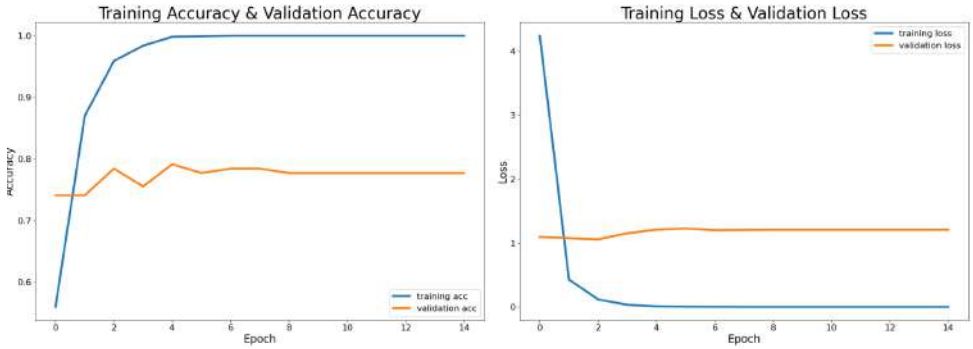
Source: Authors.

Figure 16: The first graph shows the evolution of accuracy during the training and validation of the neural network and the second shows the evolution of loss during the training and validation of the VGG16 neural network.



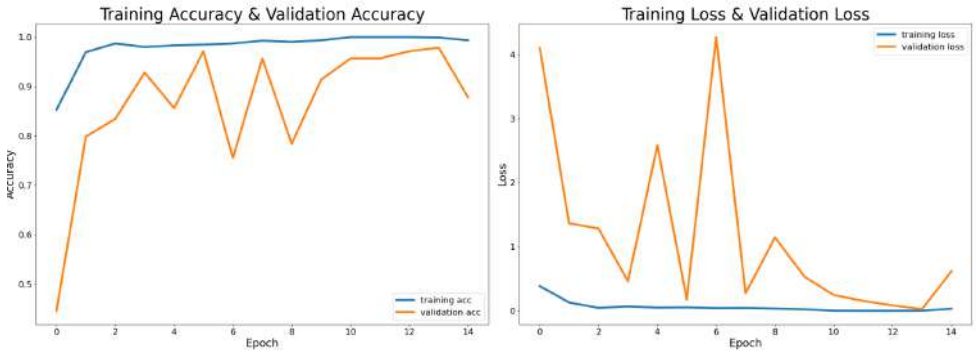
Source: Authors.

Figure 17: The first graph shows the evolution of accuracy during the training and validation of the neural network and the second shows the evolution of loss during the training and validation of the VGG19 neural network.



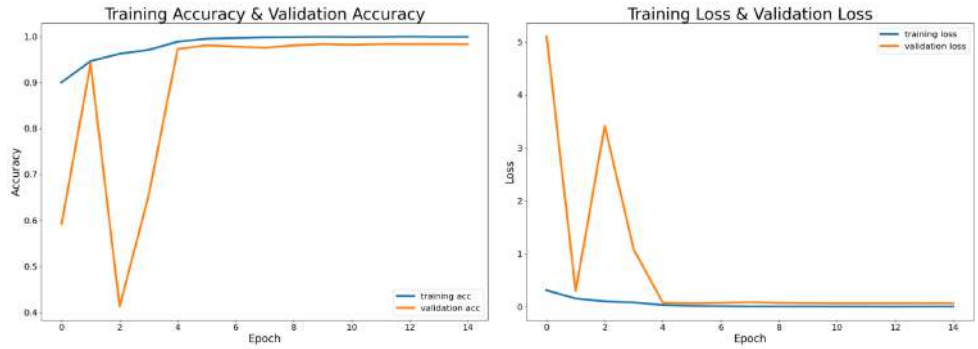
Source: Authors.

Figure 18: The first graph shows the evolution of accuracy during the training and validation of the neural network and the second shows the evolution of loss during the training and validation of the Xception neural network.



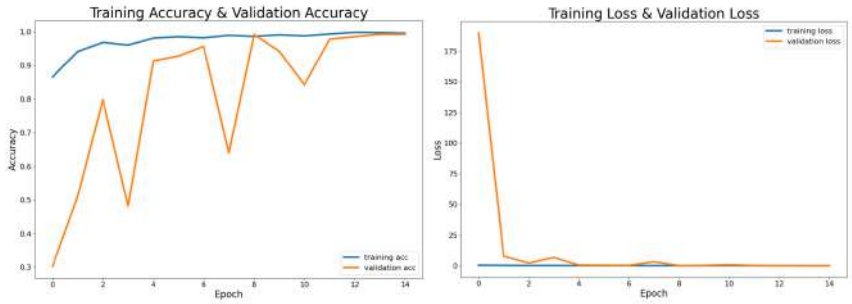
Source: Authors.

Figure 19: The first graph shows the evolution of accuracy during the training and validation of the neural network and the second shows the evolution of loss during the training and validation of the ResNet50 neural network.



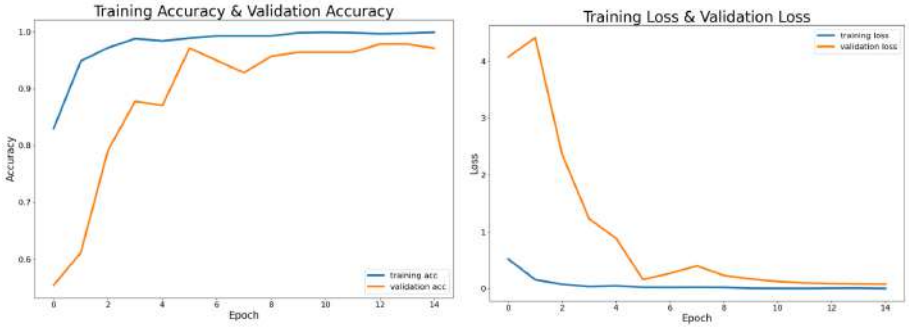
Source: Authors.

Figure 20: The first graph shows the evolution of accuracy during the training and validation of the neural network and the second shows the evolution of loss during the training and validation of the DenseNet169 neural network.



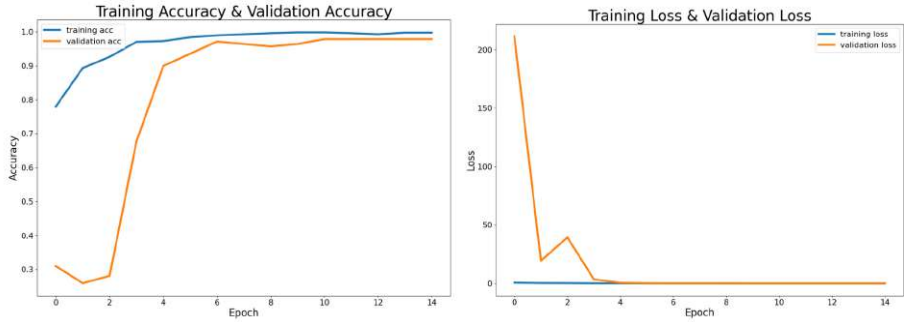
Source: Authors.

Figure 21: The first graph shows the evolution of accuracy during the training and validation of the neural network and the second shows the evolution of loss during the training and validation of the DenseNet121 neural network.



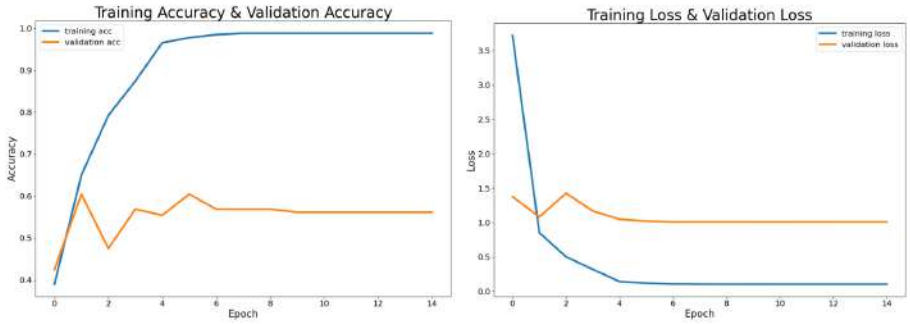
Source: Authors.

Figure 22: The first graph shows the evolution of accuracy during the training and validation of the neural network and the second shows the evolution of loss during the training and validation of the MobileNet neural network.



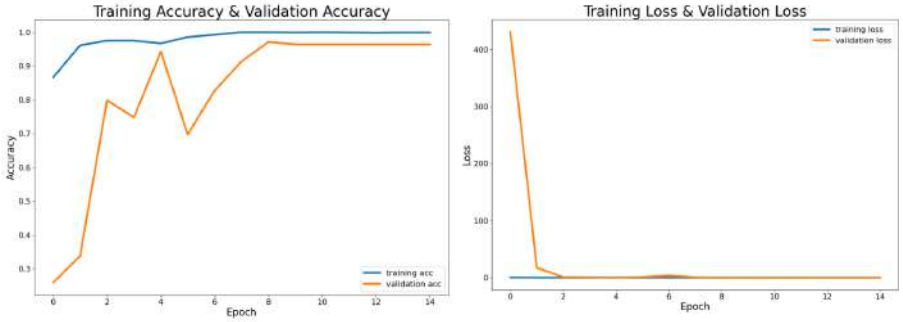
Source: Authors.

Figure 23: The first graph shows the evolution of accuracy during the training and validation of the neural network and the second shows the evolution of loss during the training and validation of the MobileNetV2 neural network.



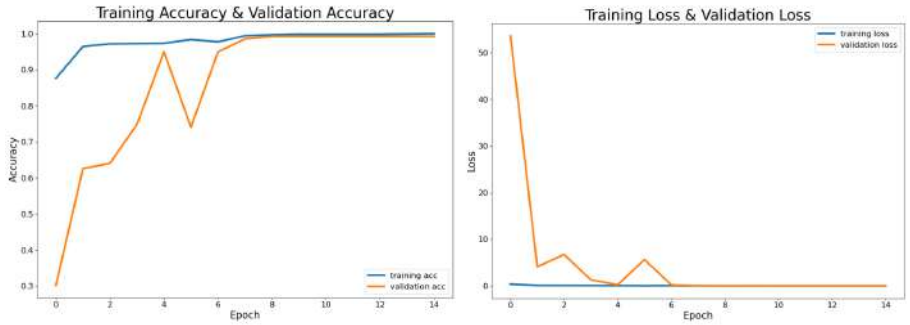
Source: Authors.

Figure 24: The first graph shows the evolution of accuracy during the training and validation of the neural network and the second shows the evolution of loss during the training and validation of the NASNetMobile neural network.



Source: Authors.

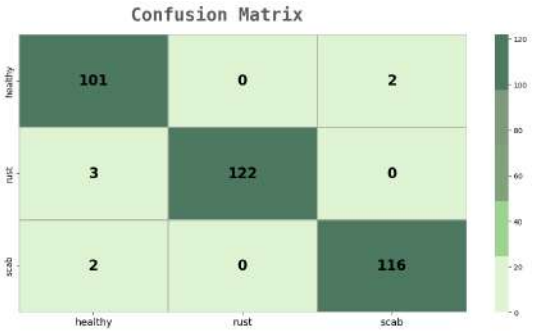
Figure 25: The first graph shows the evolution of accuracy during the training and validation of the neural network and the second shows the evolution of loss during the training and validation of the DenseNet201 neural network.



Source: Authors.

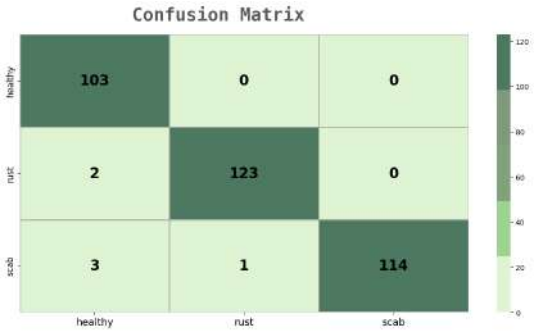
Appendix B - Confusion Matrix of Neural Networks

EfficientNetB0



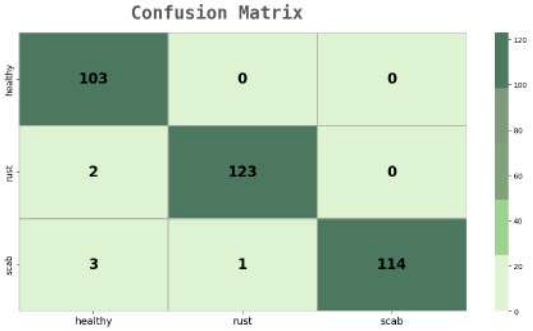
Source: Authors.

EfficientNetB5



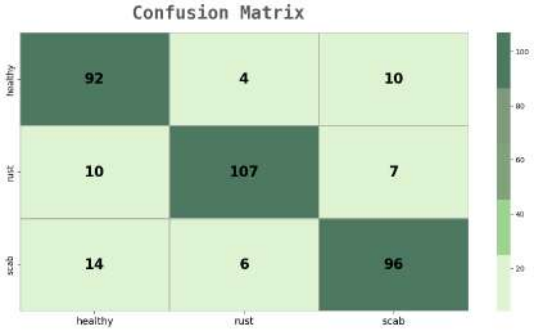
Source: Authors.

EfficientNetB1



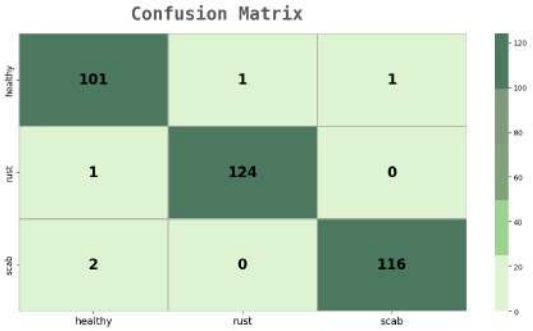
Source: Authors.

EfficientNetB6



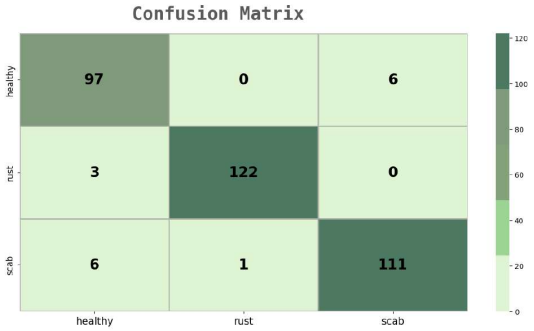
Source: Authors.

DenseNet201



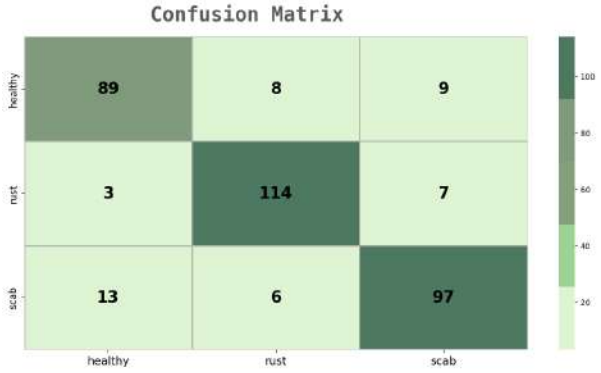
Source: Authors.

ResNet50V2



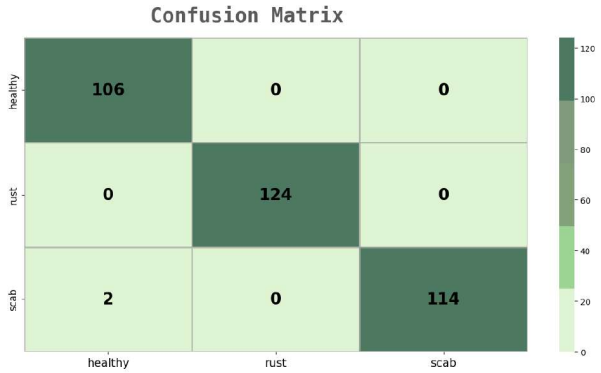
Source: Authors.

Figure 26: EfficientNetB7



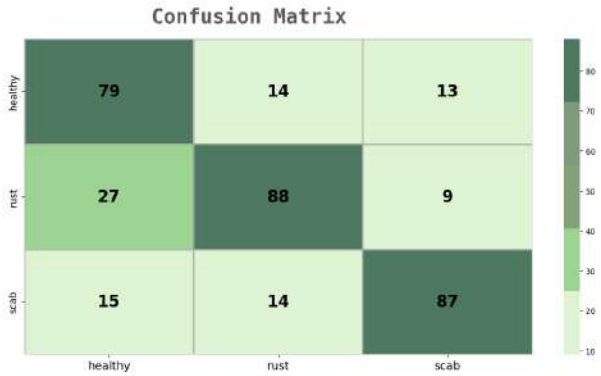
Source: Authors.

Figure 27: EfficientNetV2B2



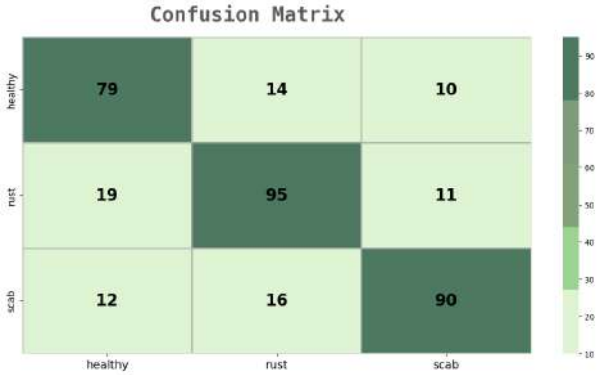
Source: Authors.

Figure 28: VGG16



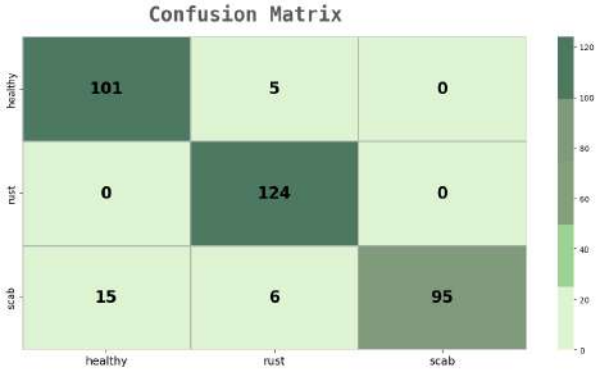
Source: Authors.

Figure 29: VGG19



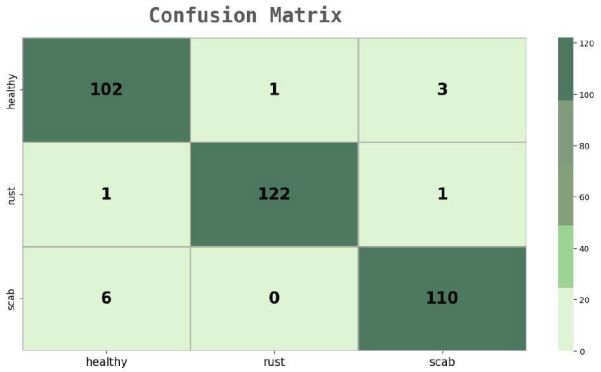
Source: Authors.

Figure 30: Xception



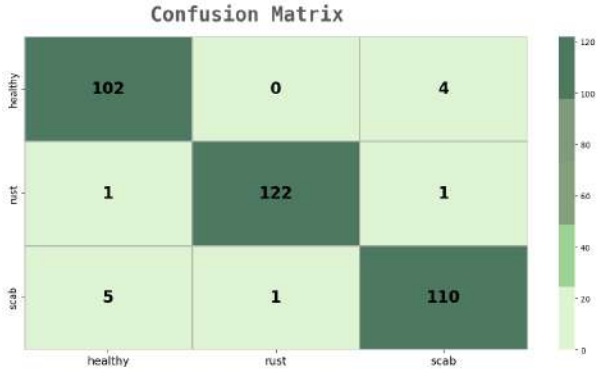
Source: Authors.

Figure 31: InceptionV3



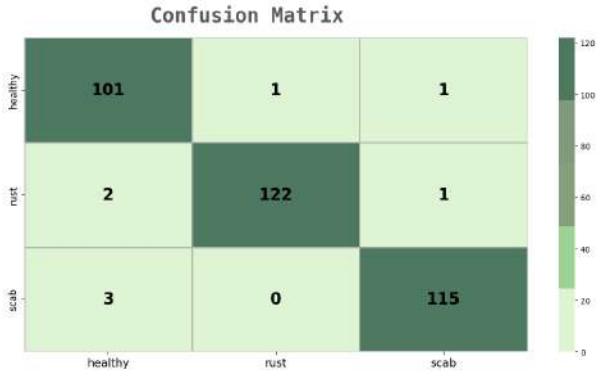
Source: Authors.

Figure 32: ResNet50



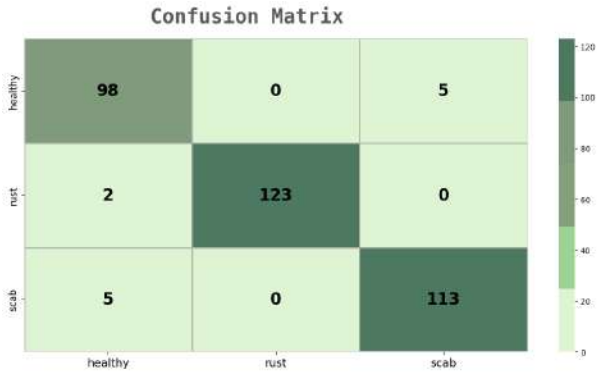
Source: Authors.

Figure 33: DenseNet169



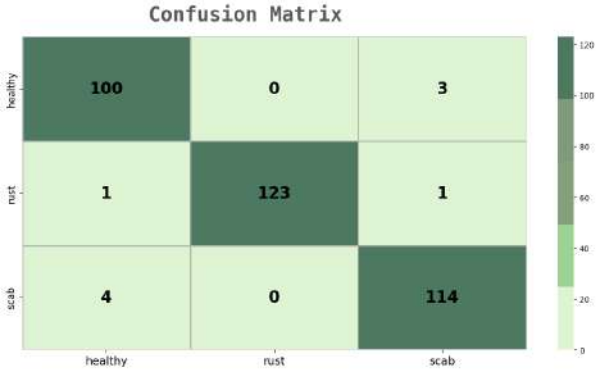
Source: Authors.

Figure 34: DenseNet121



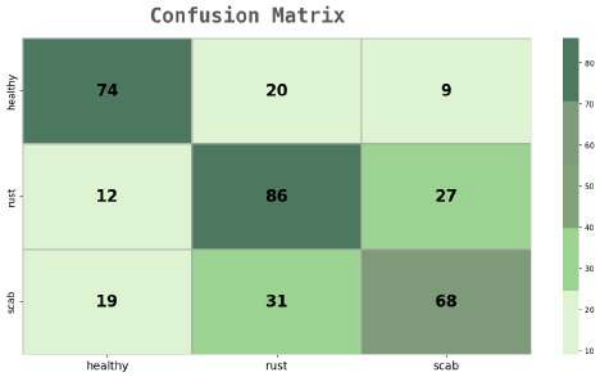
Source: Authors.

Figure 35: MobileNet



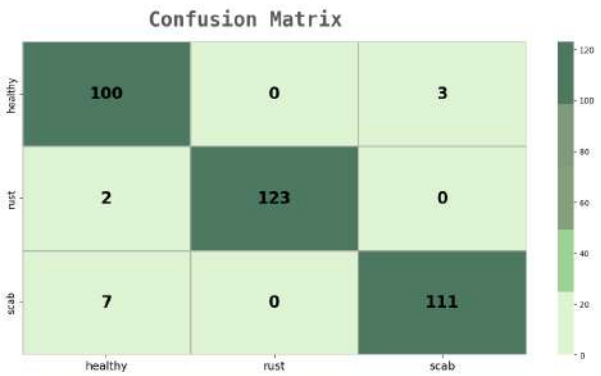
Source: Authors.

Figure 36: MobileNetV2



Source: Authors.

Figure 37: DenseNet169



Source: Authors.