Journal of Engineering and Technology for Industrial Applications

ISSN ONLINE: 2447-0228

ITEGAM-JETIA

Manaus, v.11 n.56, p.310-321. November/December, 2025.
DOI: https://doi.org/10.5935/jetia. v11i56.2788

RESEARCH ARTICLE OPEN ACCESS

AI-POWERED EARLY WARNING FOR RED PALM WEEVIL INFESTATIONS

Fatima Bouakkaz*[1], Wided Ali[2] and Chayma Boumedjene[3]

[1,3]Echahid Cheikh Larbi Tebessi University, Tebessa , Algeria
[2]Laboratory of Vision and Artificial Intellige nce (LAVIA) ,Tebessa, Algeria.

[1]https://orcid.org/0000-0001-6133-3894, [2]https://orcid.org/0000-0002-3530-7884, [3]https://orcid.org/0009-0009-0576-2294

Email: *f_bouakkez@esi.dz, Aliwided1984@gmail.com, chaymaboumadjane@gmail.com

ARTICLE INFO

Article History
Received: October 10, 2025
Revised: November 20, 2025
Accepted: December 1, 2025
Published: December 31, 2025

Keywords:
Red palm weevil,
Deep learning,
CNN,
Audio Signals ,
Detection.

ABSTRACT

The red palm weevil (Rhynchophorus ferrugineus) is a pest known for infesting and causing harm to palm trees. Symptoms of red palm weevil infection include leaf wilting, browning of the palms, exit holes in the trunk and general weakening of the tree. These symptoms only appear after full spread within the palm and in the final stages of infection, when rescuing the palm becomes challenging and control the pest. Early detection is considered the best methods for combating the red palm weevil.

The primary goal of this study aims to design a system for early detection of this pest using deep learning techniques. Audio signals from palm trees were converted into spectrograms, and a 6-layer convolutional neural network (CNN) model was trained on a dataset of audio recordings using MATLAB. The model achieved a high accuracy of 99.78% in classifying infested and healthy palm trees. The results show that deep learning May serve as a valuable method for detecting early signs of red palm weevil infestations, contributing to the control of this pest and the preservation of palm

I. INTRODUCTION

Palm trees are iconic symbols of tropical and subtropical regions, having a significant impact on the economy, environment, and culture of numerous communities. However, the red palm weevil (Rhynchophorus ferrugineus) has emerged as a significant threat to these majestic trees, causing extensive damage and leading to the loss of countless palms worldwide. The weevil attacks the heart of the palm, often going undetected until the tree is beyond saving. Traditional methods of detection rely on visible symptoms, such as wilting leaves and exit holes, which only appear in the advanced stages of infestation. At this point, it is usually too late to rescue the tree, and the pest has likely spread to neighboring palms[1][2].

To address this challenge, there is an urgent need for early detection methods that can identify infestations before visible symptoms manifest. Recent progress in artificial intelligence (AI) and deep learning presents promising opportunities for early pest detection. This research examines the use of deep learning methods, particularly convolutional neural networks (CNNs), to detect red palm weevil infestations using audio data. By analyzing the acoustic signals emitted by palm trees, our system can distinguish between healthy and infested trees with high accuracy. This paper presents An extensive overview. of existing AI-driven strategies for red palm weevil(lRPW) detection and introduces a novel deep learning-based system that leverages spectrogram analysis for early and accurate pest identification.

Figure 1: Red palm weevil (Rhynchophorus ferrugineus).
Soucer: [2].

II. AI-BASED METHODS FOR DETECTING RPW

Several AI-Based strategies are employed for detecting red palm weevil Among these are:

- The application of machine learning, particularly neural networks, which allows for the categorization of palm trees based on data, distinguishing between healthy and infected specimens, as shown in [3].
- Deep learning provides a sophisticated method, leveraging intricate neural networks trained on multimodal data such as visual, sensory, and environmental information. This results in highly accurate detection models, crucial for timely responses to RPW outbreaks [2].
- The Internet of Things (IoT) facilitates RPW detection through a network of connected sensors placed on palm trees. These sensors monitor factors like temperature, humidity, vibrations, acoustics, and motion, with subsequent data analysis revealing potential infestations [4].
- ANOVA, a statistical analysis technique, compares data from healthy and infested palms to discern variations indicative of RPW presence [5].
- Image processing techniques analyze camera-captured images, extracting visual cues like color, texture, and shape to identify infested regions, thus streamlining the detection process [6, 7].
- Finally, signal processing analyzes acoustic emissions from both the palm trees and the weevils to detect RPW. This method uses microphones or specialized sensors to capture and analyze the specific sounds produced by these pests [8].

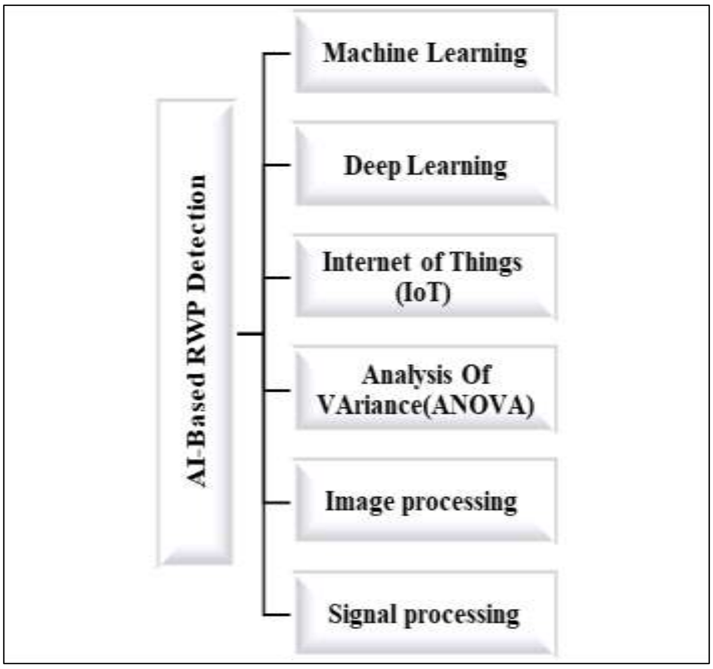

Figure 2: AI-Based Red Palm Weevil Detection (Rhynchophorus ferrugineus).
Source: Authors, (2025).

II.1 LITERATURE REVIEW

This section will focus on the use of deep learning techniques for detecting the red palm weevil (RPW). Below is a summary of related work that utilizes deep learning techniques for RPW detection An innovative method for early detection of red palm weevils (RPW) in date palm farms is presented in this paper [2]. This method combines Internet of Things (IoT) technology and deep learning. To achieve this, TreeVibes sound sensors are installed on each palm tree trunk, creating a wireless sensor network within the farm. The captured audio signals are then transmitted to a cloud server for online analysis using a finely tuned deep learning model, specifically InceptionResNet-V2. The validity of the proposed detection model was successfully confirmed using the public TreeVibes dataset.

The study [7] presents an innovative approach for detecting palm tree diseases by integrating image processing with machine learning by employing classification models like CNN and Support Vector Machine (SVM). this approach enables efficient identification of palm tree diseases by exploiting visual information extracted from images. This method offers early and accurate detection of diseases, thus contributing to improved palm tree health management. The performance of classification algorithms for predicting red palm weevil infestation in palm trees was evaluated in the study [9]. The researchers used spray logs and surveillance datasets, and tested various algorithms such as decision tree, Naive Bayes, random forest, AdaBoost, SVM, and logistic regression. The objective was to improve crop protection decisions to prevent infestations and protect palm trees. The performance of these models was evaluated in terms of precision and recall. The article also explores the use of artificial neural networks for red palm weevil recognition and examines the impact of infestation on the temperature profiles 23655of date palm trees. The system described in [10] utilizes artificial intelligence and DAS fiber optic technology to detect the sounds emitted by red palm weevil (RPW) larvae in open-air farms. Experiments were conducted in a controlled environment to train convolutional neural network (CNN) models to recognize the sound signals of infested and healthy trees.

The system incorporates a band-pass filter to eliminate environmental noise and electronic/optical interference. This paper [11] explores using transfer learning with the VGGish model, a pre-trained audio classification model, to recognize the sounds produced by Red Palm Weevils (RPW). By fine-tuning VGGish on a dataset of RPW sounds, the researchers aim to develop an effective method for detecting RPW infestations through acoustic monitoring. This approach leverages the pre-trained model's ability to extract relevant audio features, reducing the need for large training datasets specific to RPW sounds and potentially improving detection accuracy compared to training models from scratch. The focus is on adapting a general-purpose audio model to a specific bioacoustic application, offering a potentially efficient and scalable solution for early RPW detection in palm trees. In paper [12] proposed a deep learning approach for the early detection of red palm weevil (RPW) infestations by analyzing acoustic signals. Their method utilizes convolutional neural networks (CNNs) to classify audio recordings collected from palm trees as either infested or healthy. By training the CNNs on a dataset of sounds produced by RPW larvae within palm trees, the system learns to distinguish subtle acoustic differences indicative of infestation, even before external symptoms become visible. This approach Provides a non-intrusive and potentially more effective option compared to conventional inspection techniques., enabling timely intervention and mitigating the spread of RPW infestations in palm groves. The research emphasizes the capabilities of deep learning for bio-acoustic monitoring in agriculture.

In [13] explore the use of temporal Unmanned Aerial Vehicle (UAV) imagery for detecting Red Palm Weevil (RPW) infestations in date palms. Their research lever- ages multi-temporal image analysis, capturing images of palm trees at different points in time to observe changes indicative of RPW damage. By analyzing these changes in spectral and textural features over time, the authors aim to identify early signs of infestation before they become visually apparent from ground-level inspections. This approach offers a potentially scalable and efficient method for monitoring large date palm plantations, enabling early intervention and mitigating the spread of RPW. The study demonstrates the potential of combining UAV technology with temporal image analysis for precision agriculture and pest management. The paper [14] presents a sustainable approach to palm tree cultivation by utilizing the Internet of Things (IoT). It explores the use of IoT and multi-modal data for the early detection and mapping of Red Palm Weevil (RPW) infestations. This method integrates data from various sources, including acoustic sensors, thermal imaging, and environmental sensors, all collected through an IoT network deployed within palm groves. By combining these different data types, the system improves detection accuracy. Through the integration of diverse data modalities and the application of advanced data analytics and machine learning, the researchers aim to create a comprehensive system for early detection of RPW infestations and mapping their spatial distribution.

This multi-modal approach provides a more comprehensive view of palm health. compared to single- sensor methods, enabling more accurate and timely interventions, contributing to sustainable palm tree farming practices, and minimizing the economic and ecological impact of RPW. A large-scale field trial employing machine learning-assisted fiber-optic Distributed Acoustic Sensing (DAS) for the early identification of Red Palm Weevils. was presented in [15]. Their research focuses on deploying DAS technology across a substantial area of palm trees to monitor acoustic signals indicative of RPW activity. By employing machine learning algorithms, the researchers analyze the vast amounts of acoustic data collected by the DAS system to differentiate between the sounds of healthy trees and those affected by RPW. This large-scale field experiment aims to validate the effectiveness and scalability of DAS combined with machine learning for real-world RPW detection, paving the way for more efficient and comprehensive monitoring strategies in palm plantations. The following Table 1 summarize the related work of RPW with deep learning. The enumerations of citations in the body of the article must be sequenced in the order in which they appear, according to the example shown below.

Table 1: Summary of related work of RPW detection using deep learning.

Ref, Year	Objective	Techniques Used	Advantages	Disadvantages	Dataset	Accuracy	Simulatn
[9], 2021	IoT framework for RPW detection using sound data.	InceptionResNet-V2	Real-time, quick detection.	Computationally intensive; noise-sensitive.	TreeVibes sound dataset.	97.18%	Not specified
[7], 2020	Palm tree disease detection via image processing.	CNN, SVM	Early, accurate detection.	Requires advanced image processing.	Thermal images (1200 healthy, 1200 infested).	97,9 %	MATLAB
[9], 2021	Acoustic sensing for RPW detection.	ANN, CNN	Continuous, high accuracy.	Low performance in strong winds.	Acoustic data of trees.	98.8%	Not specified
[10], 2022	Real-time RPW detection via DAS.	CNN	Early detection.	Poor in high wind conditions.	Acoustic signals from trees.	99.1%	Not specified
[11], 2022	RPW detection using CNN with OF-DAS	CNN,OF-DAS	Non-invasive, real-time solution for large plantations	Data processing and system deployment challenges	Acoustic data from OF-DAS sensors	/	Not specified

[12], 2022	Early RPW detection using CNN for acoustic signal classification	CNN	Effective early detection before visible symptoms	Affected by noise.	Public TreeVibes dataset.	80%	Not specified
[13], 2023	RPW sound recognition with pre-trained models.	VGGish	High accuracy, scalable.	Sensitive to environmental noise.	TreeVibes dataset (2485 audio files).	97.1%	MATLAB
[14], 2024	IoT and multi-modal data for sustainable farming.	DL, YOLOv8	Real-time, reduced spread.	Data-heavy, computationally intensive.	UAV images (349).	99%	Not specified
[15], 2024	Field experiment for RPW detection via DAS.	CNN	Non-invasive monitoring.	Environmental factors impact results.	Data from 200 date palms (AlUla, Saudi Arabia).	/	Not specified

Source: Authors, (2025).

II.2 DISCUSSION

This table provides an overview of various studies dedicated to detecting Red Palm Weevil (RPW) using a range of techniques and datasets. The primary objective of most research is to identify RPW through non-invasive methods such as sound recognition, image processing, and acoustic sensing, with a particular emphasis on early detection, real-time monitoring, and promoting sustainable farming practices. Commonly employed techniques include Convolutional Neural Networks (CNN), Artificial Neural Networks (ANN), Support Vector Machines (SVM), and pre-trained models like InceptionResNet-V2 and VGGish. Additionally, advanced methods such as YOLOv8 and transfer learning are utilized to enhance detection capabilities. These approaches offer significant advantages, including high accuracy rates (ranging from 80% to 99%), real-time detection, scalability, and non-invasive monitoring. However, challenges remain, such as the computational intensity of these methods, their sensitivity to environmental noise, and diminished performance under adverse conditions like strong winds. The datasets used in these studies are diverse, encompassing sound recordings (e.g., TreeVibes), thermal images, acoustic data, UAV images, and field data, with many being specialized, such as thermal images of palm trees or acoustic signals from date palms. Accuracy levels vary, with VGGish achieving the highest at 99% and field experiments the lowest at 80%. While some studies utilize MATLAB for simulation, others do not specify the tools used. In conclusion, these studies underscore the effectiveness of deep learning and IoT frameworks in RPW detection, leveraging varied datasets to achieve high accuracy and real-time monitoring, despite certain environmental and computational limitations.

III. PROPOSED METHODOLOGY

In this section we will describe the implementation of our model and present its training process on the dataset. Following that, during the testing and prediction phase, we will evaluate our trained model. Finally, we will discuss the results obtained.

III.1 OVERALL ARCHITECTURE OF PROPOSED SYSTEM

The general architecture of the model is illustrated in Figure 3. The method for classifying healthy or infested palm trees involves transforming the audio signals from healthy or RPW-infested palm trees into spectrograms. Spectrograms, which are visual representations of sound features, make it easier to distinguish between healthy palm trees and those infested by the red palm weevil.

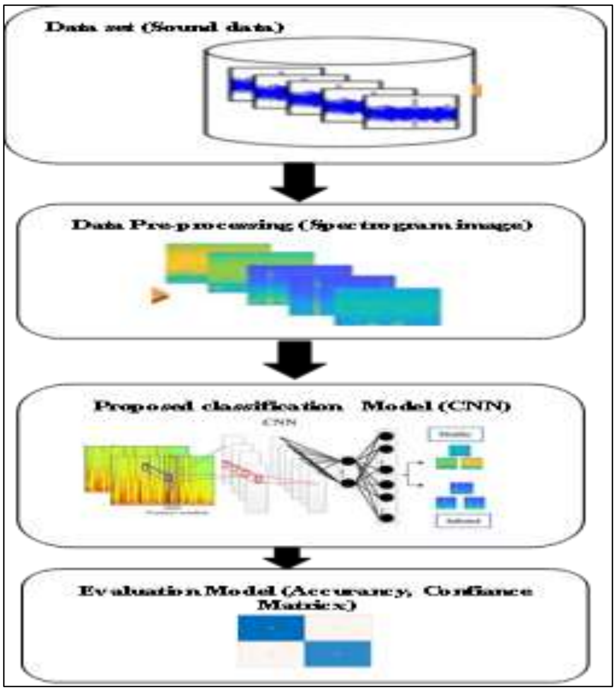

Figure 3: Overall Architecture of proposed system Red Palm Weevil Detection.
Source: Authors, (2025).

The general architecture of the model is illustrated in Figure 3. The method for classifying healthy or infested palm trees involves transforming the audio signals from healthy or RPW-infested palm trees into spectrograms. Spectrograms, which are visual representations of sound features, make it easier to distinguish between healthy palm trees and those infested by the red palm weevil.

III.1.1 Dataset Collection

The data consists of audio recordings from palm trees, some of which are infested and others healthy. The "infested" recordings come from trees that were intentionally infested with weevil larvae less than three weeks old, representing an early stage of infestation. The "healthy" recordings are collected from ten non-infested trees.Some examples of these signals are shown in the following figure 4 and figure 5:

Figure 4: Examples of Healthy Signals of data set.
Source: Authors, (2025).

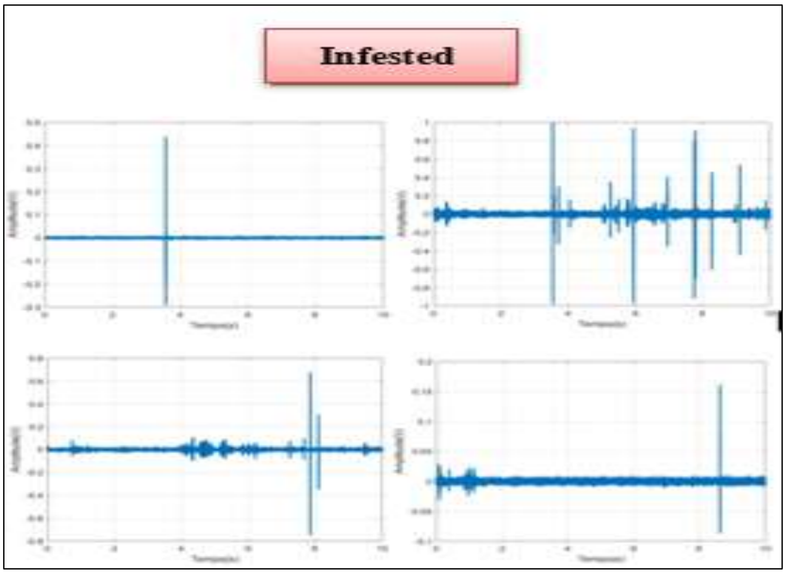

Figure 5: Examples of Infested Signals of data set.
Source: Authors, (2025).

III.1.2 Dataset Preprocessing

After selecting the dataset, a crucial step in any signal analysis is signal processing. The signals must be processed and modified to ensure full compatibility and to achieve greater efficiency for the designed model. To feed the input signals into the model, the signals are normalized for better readability. The signals were reconfigured according to the requirements of the constructed model.

III.1.3 Model creation

The problem of red palm weevil detection is defined as follows: given an audio signal, the goal is to determine whether the palm tree is infested by the weevil or not. The general architecture of the model is illustrated in Figure 6. The method for classifying healthy or infested palm trees involves transforming the audio signals from healthy or RPW-infested palm trees into spectrograms. Spectrograms, which are visual representations of sound features, make it easier to distinguish between healthy palm trees and those infested by the red palm weevil.

III.2 EXPERIMENTAL RESULTS AND ANALYSIS

- **Choice of environment :**The simulations were carried out on a PC equipped with an Intel(R) Core(TM) i3-7020U CPU @ 2.30GHz processor, 8 GB RAM, and a 64-bit Windows 10 Professional operating system. MATLAB (R2020a) was used as the simulation platform.
- **Why MATLAB?:** MATLAB, developed by MathWorks, is a software optimized for addressing scientific and technical challenges. It revolves around matrix-based operations, combining mathematics, graphical modeling, and programming. MATLAB is equipped with a vast collection of built-in and pre-written functions, utilizing simple and powerful notations to facilitate a wide range of common computational tasks. These functions, regardless of their programming language, are easily accessible and organized within MATLAB, offering a unified platform for their use.

III.2.1 Data Preparation

After downloading the dataset TreeVibes used in researches **[9,29,30]**, Figure 6 shows the two directories containing our audio datasets:

- "clean": for recordings of healthy palm trees contain 2341 wav file
- "infested": for recordings of infested palm trees contain 2294 wav file

Figure 6: Directories containing our dataset.
Source: Authors, (2025).

Using the audioDatastore function presented in Figure 7 to create an audio datastore from a data folder containing .wav files and using the names of subfolders as labels.

```
datafolder = fullfile(pwd, 'audio');%Audio datastore
ads = audioDatastore(datafolder, ...
'IncludeSubfolders',true, ...
'FileExtensions','.wav', ...
'LabelSource','foldernames');
```

Figure 7: Creation of an audio datastore.
Source: Authors, (2025).

III.2.2 Mel Spectrogram extraction for audio clips Data Preparation

Generate Mel Spectrogram: A mel spectrogram is a visual representation of the spectrum of frequencies in a sound as they vary with time, using the mel scale. The code presented in figure 8 appears to generate Mel spectrogram images for each audio file in the audio dataset and save them as JPEG files with the same name and location as the original audio files, but with the file extension changed to '.jpg'.

```
fs = 44100 ; %sampling time for melspectrogram
for i=1:length(ads.Files)
    [filepath,filename,ext] = fileparts(ads.Files{i})
    audiodata = read(ads);
    %Pre-process audio data
     path=fullfile(filepath,filename)
    %Save spectrogram as image
melSpectrogram(audiodata,fs)
   colorbar ('off');
   axis off;
f=gcf
path
saveas(f,path,'jpg');
%Crop spectrogram data only
file = [path,'jpg'];
end
move_file(ads);
```

Figure 8: Mel Spectrogram extraction from an audio datastore.
Source: Authors, (2025).

Figures 9-10 shows the spectrograms of the input audio healthy and infected

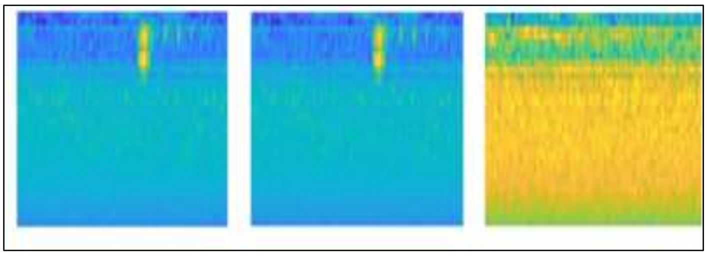

Figure 9: Examples of the spectrograms of the audio clip of healthy Palms.
Source: Authors, (2025).

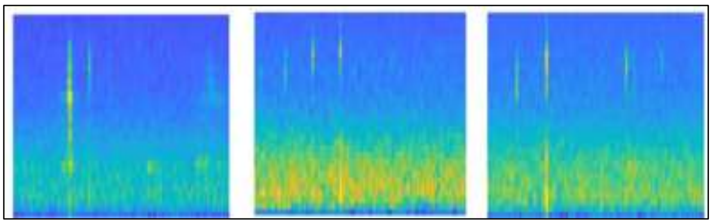

Figure 10: Examples of the spectrograms of the audio clip of infested Palms.
Source: Authors, (2025).

A function datastore Figure 11 is used to access the images stored in the "spectrograms" directory for automatic analysis, including subfolders, and assigns appropriate labels to each image based on the subfolder structure. Then, the images are resized to a consistent size of 100x100 pixels using a custom function defined by the "ReadFcn" parameter.

```
imds =
imageDatastore('Spectrograms','IncludeSubfolders',true,'LabelSource','foldernames','
ReadFcn',@(f) imresize(imread(f),[100 100]));

[trainImgs,valImgs,testImgs] = splitEachLabel(imds,0.8,0.1,0.1,'randomized');
```

Figure 11: Creation of the image datastore and resizing of the images.
Source: Authors, (2025).

III.2.3 Data set Division Preparation

Figure 12 represents the division of our dataset (80% train, 10% validation, 10% test). The division process must be random, meaning that the images will be randomly distributed among the sets using the 'randomized' function.

```
imds =
imageDatastore('Spectrograms','IncludeSubfolders',true,'LabelSource','foldernames','ReadFcn',
@(f) imresize(imread(f),[100 100]));

[trainImgs,valImgs,testImgs] = splitEachLabel(imds,0.8,0.1,0.1,'randomized');
```

Figure 12: The division of our dataset.
Source: Authors, (2025).

The variable numClasses in Figure (12) represents the number of distinct classes present in the training set of the imageDatastore. It provides the number of unique classes in the training data of the image datastore. Figure 13 shows the number of classes in the training set.

```
numClasses = numel(categories(trainImgs.Labels));
```

Figure 13: Number of classes in the training set.
Source: Authors, (2025).

III.2.4 Model CNN Creation

This code defines a CNN with multiple convolutional layers, followed by batch normalization, ReLU activations, max-pooling, and dropout regularization. The network ends with fully connected and softmax layers to classify the input images into numClasses categories. This architecture is designed to extract hierarchical features from the images and reduce overfitting during training. Figure 14 shows the MATLAB implementation of the CNN architecture of our proposed model.

```
dropoutProb = 0.2;
layers = [
    imageInputLayer([100 100 3])
    convolution2dLayer(3,16,'Padding','same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(2,'Stride',2)
    convolution2dLayer(3,32,'Padding','same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(2,'Stride',2,'Padding',[0,1])
    dropoutLayer(dropoutProb)
    convolution2dLayer(3,64,'Padding','same')
    batchNormalizationLayer
    reluLayer
    dropoutLayer(dropoutProb)
    convolution2dLayer(3,64,'Padding','same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(2,'Stride',2,'Padding',[0,1])
    dropoutLayer(dropoutProb)
    convolution2dLayer(3,64,'Padding','same')
    batchNormalizationLayer
    reluLayer
    dropoutLayer(dropoutProb)
    convolution2dLayer(3,64,'Padding','same')
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer([1 13])
    fullyConnectedLayer(numClasses)
    softmaxLayer
    classificationLayer];
```

Figure 14: Model CNN Creation.
Source: Authors, (2025).

The network is built using several types of layers that are common in CNN architectures:

a) imageInputLayer([100 100 3]) :This is the first layer of the network. It specifies the input size for the image data. The input image is 100x100 pixels with 3 color channels (RGB).

b) Convolutional Layers:These layers perform convolution operations to extract features from the input image.

- convolution2dLayer(3, 16, 'Padding', 'same'): This defines a 2D convolutional layer with 16 filters, each of size 3x3. The 'Padding', 'same' option ensures that the output size is the same as the input size (padding is added to the edges of the image).
- The following layers for convolution (and subsequent convolution layers) operate similarly, with varying numbers of filters (16, 32, 64) and padding.

c) batchNormalizationLayer :This layer normalizes the output of the previous layer to improve training stability by maintaining the mean and variance of activations. It can speed up training and prevent overfitting.

d) ReluLayer :The ReLU (Rectified Linear Unit) activation function is applied to introduce non-linearity in the network. It replaces all negative values in the output with zero.

e) maxPooling2dLayer(2, 'Stride', 2):This layer performs max-pooling, which reduces the spatial dimensions of the feature maps (height and width) by taking the maximum value from each 2x2 region. This helps reduce the number of parameters and computation, and makes the model more invariant to small translations in the image. Some of the pooling layers have adjusted padding and stride to control the output size further, like Padding=[0,1] and the custom stride.

f) DropoutLayer(dropoutProb) :The dropout layer applies dropout regularization with a probability of 0.2. During training, this randomly deactivates 20% of the neurons in the layer to prevent overfitting.

g) Fully ConnectedLayer (numClasses): This layer is a fully connected layer where the number of neurons equals numClasses, representing the number of output classes in the classification task.

h) softmaxLayer :This layer applies the softmax activation function to the output, converting the raw network outputs into probabilities for each class. It ensures that the output values sum to 1, representing a probability distribution over the classes.

i) classificationLayer :The final layer of the network. It is responsible for calculating the loss (such as cross-entropy loss) between the predicted class probabilities and the true class labels during training. This is used to update the model's **weights.**

III.2.5 Adam Optimzer

We chose the Adam optimizer for training our model. The Adam optimizer is a gradient-based optimization algorithm designed for stochastic objective functions. It is characterized by the use of adaptive estimates of lower-order moments. Adam is particularly well-suited for complex problems involving large datasets.

```
options = trainingOptions('adam', ...
    'Plots','training-progress',"MiniBatchSize",64, ...
    'ValidationData',valImgs)
```

Figure 15: Adam Optimizer.
Source: Authors, (2025).

III.2.6 Model Training

The command described in Figure 16 performs the training of the neural network using the training data, the model layers, and the specified options.

```
trainednet = trainNetwork (trainImgs,layers,options)
```

Figure 16: Model Execution.
Source: Authors, (2025).

After executing this code, the trained network object trainednet will be saved in the file "trainednet.mat".

```
save('trainednet.mat',"trainednet")
```

Figure 17: Model saves.
Source: Authors, (2025).

III.2.7 Model Testing

Figure 19 shows the function that performs classification using our trained network (trainednet) on the test set (testImgs) and assigns the predicted labels to the variable predict.

```
predict = classify(trainednet,testImgs)
```

Figure 18: Model Testing
Source: Authors, (2025).

IV. RESULTS AND DISCUSSIONS

To illustrate the results obtained by our model, we present below the accuracy and error metrics, as well as the confusion matrix, based on the number of epochs. Each epoch represents the number of times the algorithm processes the entire dataset.

IV.1 DISCUSSION ACCURACY AND ERROR:

We used our CNN model with our dataset and trained it for 8 epochs. After analysing the results, we observed the following: According to Figure 19 in the next page, the training and validation accuracy increases with the number of epochs, indicating that the model learns more information with each iteration. Our model achieved an accuracy of 99.78% for palm tree classification. Similarly, as shown in Figure 20, the training and validation error decreases as the number of epochs increases, eventually reaching a certain level of stability. This indicates that the training was successful. The following table compares the accuracy of our model with precedent studies:

Table 2: Accuracy Comparaison table.

Reference	Accuracy
[2]	97.18%
[7]	97.89
[9]	98.8%
[10]	99.1%
[11]	97.1%
[12]	80%
[14]	99%
Our Model	**99.78%**

Source: Authors, (2025).

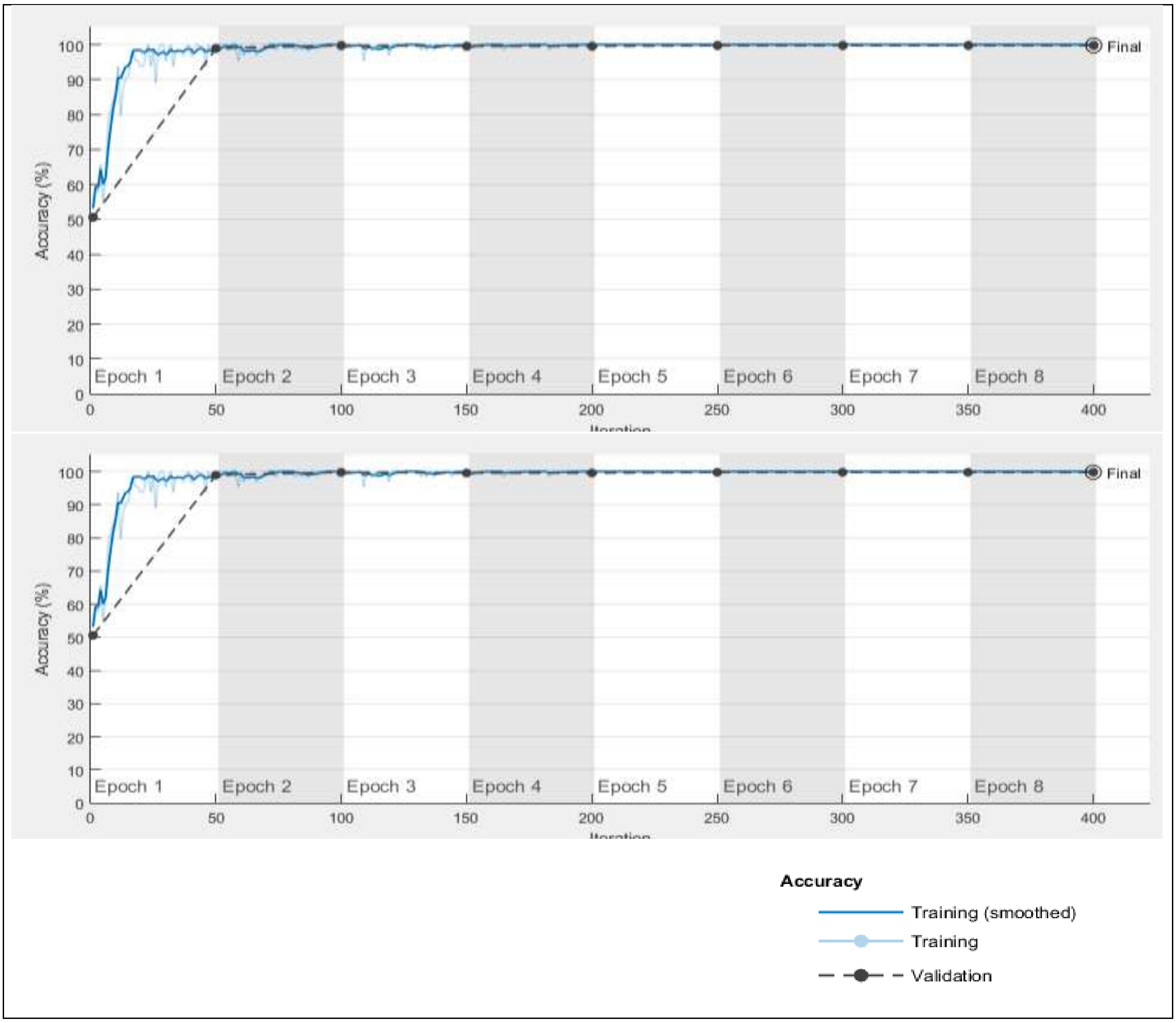

Figure 19: Model Accuracy with 8 Epochs.

Source: Authors, (2025).

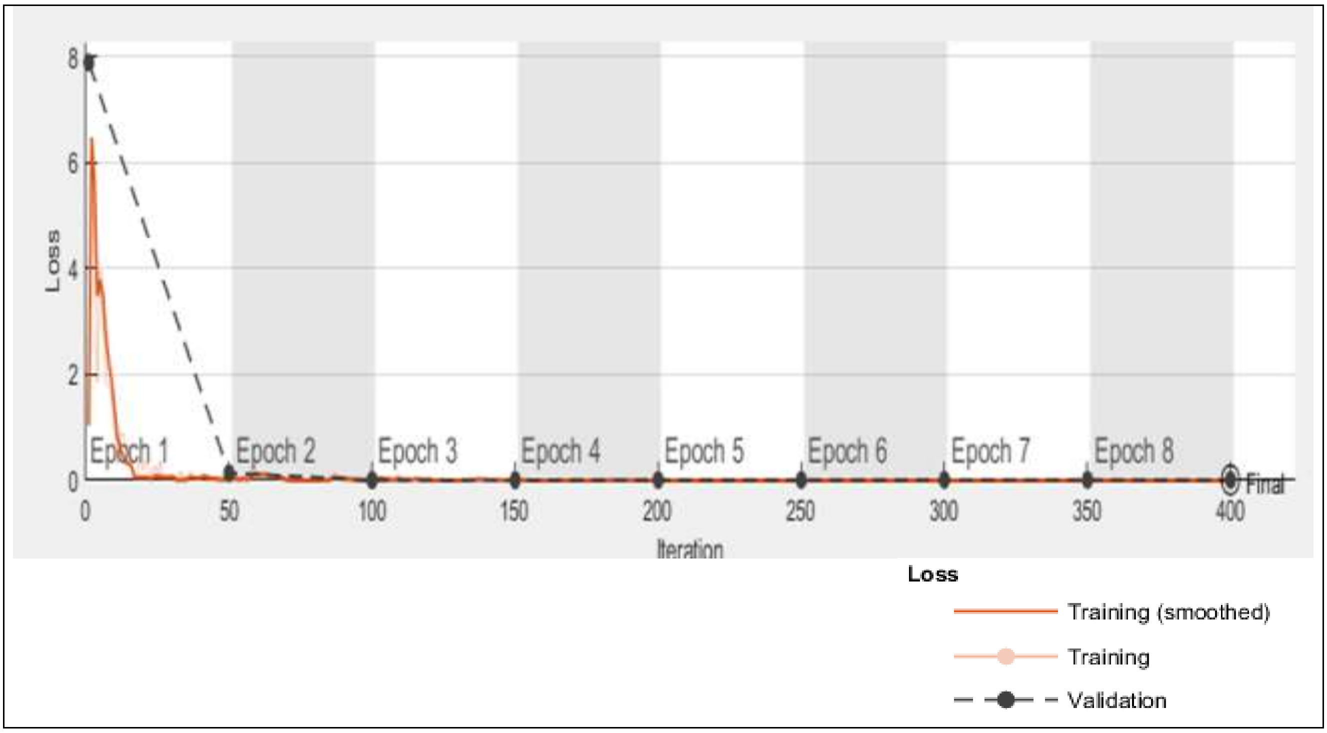

Figure 20: Model Error with 8 Epochs.
Source: Authors, (2025).

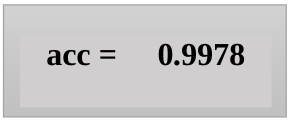

Figure 21: Accuracy of our model.
Source: Authors, (2025).

IV.1.1 Confusion Matrix:

The confusion matrix, also known as the error matrix, is a table that presents various predictions and test results by comparing them to the actual values. It allows for a detailed statistical analysis more effectively and provides clear data visualization to facilitate result interpretation. The confusion matrix of our model is shown in Figure 22

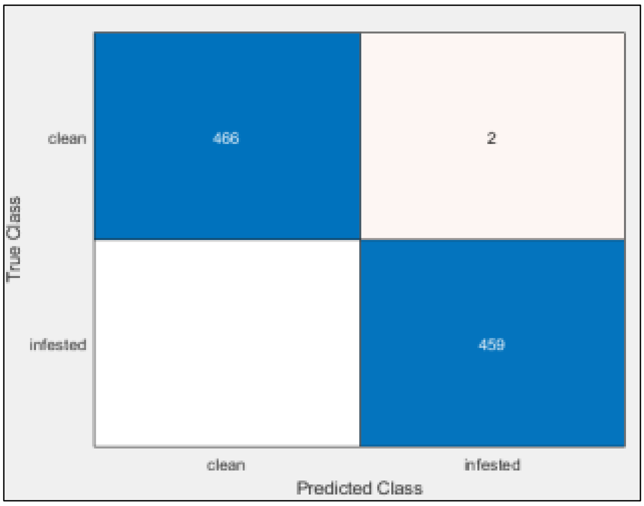

Figure 22: Confusion Matrix.
Source: Authors, (2025).

According to the confusion matrix, out of 927 spectrogram images, the model correctly identified 466 healthy palm trees and 459 infested palm trees, with only 2 errors in total. This high level of accuracy demonstrates that the model is highly promising and reliable. It can effectively serve as a tool for the early detection of red palm weevil infestations. The training results further confirm the model's excellent performance and steady progression, highlighting its potential for real-world applications in pest management and palm tree preservation.This suggests the training process was highly effective, resulting in a well-trained and stable network.

V. CONCLUSIONS

In conclusion, this study highlights the transformative potential of deep learning in the early detection of red palm weevil infestations. By leveraging a 6-layer convolutional neural network trained on audio data, we developed a highly accurate system capable of distinguishing between healthy and infested palm trees with an accuracy of 99.78%. The use of spectrograms as a visual representation of acoustic signals proved to be an effective approach for identifying subtle differences in sound patterns associated with weevil infestations. Our results demonstrate that deep learning can significantly enhance the efficiency and accuracy of pest detection, offering a non-invasive and scalable solution for palm tree protection. Future research could explore the integration of additional data modalities, such as thermal imaging and environmental sensors, to further improve detection accuracy and robustness. Additionally, deploying this system in real-world palm plantations and testing its performance under various environmental conditions will be essential for practical implementation. By continuing to refine and expand upon these AI-driven techniques, we can develop more sustainable and effective strategies for managing red palm weevil infestations, ultimately preserving the health and vitality of palm trees worldwide.

VI. FUTURE ENHANCEMENT

In this study, we addressed the challenge of early detection of the red palm weevil to prevent its spread and minimize damage to palm trees. Our primary goal was to develop an artificial intelligence-based system using deep learning to detect this pest through a convolutional neural network. Our model successfully achieved 99.78% accuracy in classifying healthy and infested palm tree cases.

VII. REFERENCES

[1] de Granville, J. J., Gayot, M., & Guitet, S. (2014). Guide des palmiers de Guyane. Office National des Forˆets (ONF).

[2] Karar, M. E., Reyad, O., Abdel-Aty, A. H., Owyed, S., & Hassan, M. F. (2021). Intelligent IoT-Aided early sound detection of red palmWeevils. Computers, Materials and Continua, 69(3), 4095-4111.

[3] Al-Saqer, S. M., & Hassan, G. M. (2011). Artificial neural networks based red palm weevil (Rynchophorus Ferrugineous, Olivier) recognition system. Am. J. Agric. Biol. Sci, 6, 356-364.

[4] Koubaa, A.; Aldawood, A.; Saeed, B.; Hadid, A.; Ahmed, M.; Saad, A.; Alkhouja, H.; Ammar, A.; Alkanhal, M. Smart Palm: An IoT framework for red palm weevil early detection. Agronomy 2020, 10, 987.

[5] El-Faki, M. S., El-Shafie, H. A. F.,& Al-Hajhoj, M. B. R. (2016). Potentials for early detection of red palm weevil (Coleoptera: Curculionidae)-infested date palm (Arecaceae) using temperature differentials. The Canadian Entomologist, 148(2), 239-245.

[6] Golomb, O., Alchanatis, V., Cohen, Y., Levin, N., Cohen, Y., & Soroker, V. (2015). Detection of red palm weevil infected trees using thermal imaging. In Precision agriculture'15 (pp. 322-37). Wageningen Academic Publishers.

[7] Alaa, H., Waleed, K., Samir, M., Mohamed, T., Sobeah, H., & Salam, M. A. (2020). An intelligent approach for detecting palm trees diseases using image processing and machine learning. International Journal of Advanced Computer Science and Applications, 11(7).

[8] Ashry, I., Mao, Y., Al-Fehaid, Y., Al-Shawaf, A., Al-Bagshi, M., Al-Brahim, S., ... & Ooi, B. S. (2020). Early detection of red palm weevil using distributed optical sensor. Scientific reports, 10(1), 3155.

[9] Hetzroni, A.; Soroker, V.; Cohen, Y. Toward practical acoustic red palm weevil detection. Comput. Electron. Agric. 2016, 124, 100–106.

[10] Wang, B., Mao, Y., Ashry, I., Al-Fehaid, Y., Al-Shawaf, A., Ng, T. K., ... & Ooi,B. S. (2021). Towards detecting red palm weevil using machine learning and fiber optic distributed acoustic sensing. Sensors, 21(5), 1592.

[11] Ashry, I., Wang, B., Mao, Y., Sait, M., Guo, Y., Al-Fehaid, Y., ... & Ooi,B. S. (2022). CNN–Aided Optical Fiber Distributed Acoustic Sensing for Early Detection of Red Palm Weevil: A Field Experiment. Sensors, 22(17), 6491.

[12] Mehanna, H. M. (2022). Develop an Acoustic Device for Red Palm Weevil Early Detection.

[13] Torky, M., Dahy, G., & Hassanien, A. E. (2023). Recognizing sounds of Red Palm Weevils (RPW) based on the VGGish model: Transfer learning methodology. Computers and Electronics in Agriculture, 212, 108079.

[14] Boulila, W., Alzahem, A., Koubaa, A., Benjdira, B., & Ammar, A. (2023). Early detection of red palm weevil infestations using deep learning classification of acoustic signals. Computers and Electronics in Agriculture, 212, 108154.

[15] Delalieux, S., Hardy, T., Ferry, M., Gomez, S., Kooistra, L., Culman, M., & Tits,L. (2023). Red palm weevil detection in date palm using temporal uav imagery. Remote Sensing, 15(5), 1380.