Indian Journal of Computer Science and Technology
https://www.doi.org/10.59256/indjcst.20260503132
Volume 5, Issue3 (September-December 2026), PP: 1100-1104.
www.indjcst.com
ISSN No: 2583-5300

Coconut Tree Disease Detection Using Deep Learning

Prasanna kumar Naik R[1], Chethan Kumar G S[2], Swarnalatha[3], Preethi K P[4], Manasa V Sangapur[5]

[1,2,3,4] Assistant Professor, Department of Master of Computer Applications, University BDT College of Engineering (A Constituent College of VTU, Belagavi), Davangere, Karnataka, India.

[5]Student, Department of Master of Computer Applications, University BDT College of Engineering (A Constituent College of VTU, Belagavi), Davangere, Karnataka, India.

To Cite this Article: *Prasanna kumar Naik R[1], Chethan Kumar G S[2] ,Swarnalatha[3],Preethi K P[4], Manasa V Sangapur[5], "Coconut Tree Disease Detection Using Deep Learning", Indian Journal of Computer Science and Technology, Volume 05, Issue 03 (September-December 2026), PP: 1100-1104.*

Copyright: ©2026 This is an open access journal, and articles are distributed under the terms of the Creative Commons Attribution License; Which Permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Abstract: *Coconut (Cocos nucifera) is a high value tropical plantation crops that support agricultural livelihoods and numerous agro based industries However coconut palms are very liable to foliar, stem and crown diseases such as bud basis dropping, bud rot, gray leaf spot, leaf rot, and stem bleeding which significantly reduced nut yields and threaten palm survivals if left undiagnosed Conventional diseases identifications relies on manual visuals inspection by farmers or domain experts a process that is time-consuming, subjective and difficult to scales across large plantation particularly given the height of mature coconut palms To address these challenges this papers presents an computerized Coconut Tree Disease Finding Arrangement utilizing Deep Learning for rapid and accurate image based diagnosis the proposed pipeline processes labeled coconut leaf and trunk image datasets through systematic image preprocessing including resizing pixel normalization, and facts growth to progress model generality under fluctuating lighting and background conditions A Convolutional Neural Network (CNN) architectures powered by transfers learnings is accomplished with TensorFlow and Keras to classify coconut tree samples across five ailment categories evaluated using accurateness, meticulousness, recall, F1-score and confusion matrix analysis. Deployed through a user friendly Flask web interface the system allows farmers to upload tree or leaf images and receives instants diseases classifications result alongside preventive guidance, supporting timely intervention and sustainable plantation management.*

Key Word: *Coconut Tree Disease Detection, Convolutional Neural Networks (CNN), Deep Learning, Image Classification, Precision Agriculture, Transfer Learning.*

I. INTRODUCTION

Coconut (*Cocos nucifera*) is one of the record economically and nutritionally vitals plantations crop across tropical coastal regions sustaining millions of small holder farmer and supporting diverse food oil, coir and pharmaceutical industries however coconut cultivation faces persistent yield losses caused by severe diseases affecting leaves, crowns, roots, and stems—specifically If these impurities are not perceived and treated at an premature point, they can spread rapidly across entire plantations, severely degrading nut quality and causing irreversible palm mortality [1].

Traditionally, coconut disease identification depends on manual visual inspection conducted by farmers or agricultural extension specialists. This manual approach suffers from major practical drawbacks: it is labor-intensive, inherently subjective, prone to misdiagnosis between diseases with lapping visual indicators, and trying to scale across large orchards—especially given the considerable height of mature coconut palms. Moreover, rural farmers often lack timely access to trained plant pathologists, resulting in delayed treatment or excessive, inappropriate pesticide application [2], [4].

Recent breakthroughs in Deep Learning (DL) and computer vision have transformed automated plant pathology [1], [5]. Distinct conventional image handing out procedures that require hand-crafted color or texture descriptors, Convolutional Neural Networks (CNNs) [6] automatically learn hierarchical spatial and textural representations directly from raw leaf and trunk images [2], [3]. By leveraging transfer learning with established deep architectures [7]–[10], CNN models achieve strong diagnostic accuracy even on domain-specific agricultural datasets with modest sample sizes while keeping robustness counter to variations in natural illumination and background clutter [3], [4].

To provide farmers and plantation managers with an accurate, rapid, and accessible diagnostic tool, this paper presents an computerized **Coconut Tree Disease Detection System** built on Deep Learning. The proposed workflow processes labeled coconut palm disease image datasets using resizing, pixel normalization, and data augmentation to improve generalization. A deep CNN archetypal built using TensorFlow [11] and Keras is trained to classify coconut tree images into five dissimilar sickness classes and evaluated using exactness, care, recall, F1-score, and muddle matrix investigation. Finally, the proficient model is integrated into a trivial Flask web submission that allows users to upload an image of a coconut tree or leaf and receive immediate disease identification alongside recommended preventive measures.

The principal charities of this work are summarized as follows:

- **Automated Image Preprocessing and Augmentation:** Implementing a standardized image preparation pipeline that increases strength counter to variations in orientation, scale, and field lighting conditions.
- **Deep Learning Classification Pipeline:** Training and evaluating a CNN-based assignment erudition model personalized specifically for classifying five key coconut palm diseases: bud root dipping, shoot rot, gray sprig spot, leaf rot, and trunk bleeding.
- **Accessible Web Diagnostic Interface:** Deploying the trained model inside a user-friendly Flask web submission for instant image upload and real-time disease prediction.

II. LITERATURE SURVEY

Automated plant bug diagnosis has grew significantly over the past decade, progressing from manual scouting and hand-crafted image processing techniques to deep Convolutional Neural Networks and lightweight transfer learning models [1].

A. Manual Inspection and Classical Computer Vision Approaches

Early automated plant pathology systems combined manual leaf segmentation with hand-engineered feature extraction—such as color histograms, shape descriptors, and texture matrices—paired with outmoded machine learning classifiers like Support Vector Machines (SVM) or k-Nearest Neighbors [1]. Although these classical methods demonstrated feasibility in controlled laboratory settings, their accuracy degraded sharply on real-world field images characterized by uneven sunlight, shadows, complex foliage backgrounds, and subtle early-stage lesion variations [2], [4].

B. Convolutional Neural Networks for Plant Disease Recognition

The arrival of Deep Learning [5] and deep CNN architectures [6] eliminated the need for labor-intensive feature engineering by learning discriminative lesion patterns directly from pixel data. Sladojevic et al. [4] demonstrated that fine-tuned deep CNNs can accurately classify multiple plant leaf diseases directly from raw images, distinguishing subtle fungal and bacterial symptoms. Similarly, Mohanty et al. [2] trained deep neural architectures on large leaf image collections, confirming that CNNs substantially outperform classical shallow image classifiers across diverse crop species. Extending this evaluation to real-field conditions, Ferentinos [3] benchmarked multiple deep CNN architectures across diverse plant leaf disease categories, showing high diagnostic reliability when trained with sufficient image diversity and proper regularization.

C. Transfer Learning and Lightweight Architectures

Training deep convolutional networks entirely from scratch requires massive labeled image datasets and substantial GPU compute time. To overcome this limitation in specialized agricultural domains, scholars widely adopt allocation learning using architectures pre-trained on large-scale visual datasets—such as VGG [7], Inception/GoogLeNet [8], and Residual Networks (ResNet) [9]—implemented within scalable frameworks like TensorFlow [11]. More recently, compact architectures such as MobileNetV2 [10] have gained prominence for herbal sickness uncovering because their inverted residual and depthwise separable convolution blocks reduce parameter count and inference latency without forfeiting exactness, making them ideal for web and mobile deployment.

D. Research Gaps and Motivation

Reviewing the existing literature highlights three critical gaps:

1. Limited Focus on Coconut Palms: Most widely cited plant-disease benchmarks emphasize common annual crops (such as tomato, potato, and maize), leaving specialized plantation crops like coconut comparatively underrepresented.
2. Sensitivity to Real-World Field Conditions: Models trained solely on uniform laboratory leaf backgrounds often struggle when exposed to outdoor plantation lighting, varied camera angles, and partial background clutter.
3. Lack of Practical End-User Deployment: Many deep learning studies remain offline experimental scripts without an accessible web interface where farmers can upload images for instant diagnosis.

To address these gaps, the proposed system applies image augmentation and transfer-learning-based CNN classification [3], [10], [11] specifically to five key coconut tree diseases and deploys the trained model inside a simple web interface for practical plantation use.

III. METHODOLOGY

The proposed Coconut Tree Disease Detection System follows an end-to-end Deep Learning image classification pipeline consisting of dataset collection, image preprocessing and augmentation, dataset partitioning, Convolutional Neural Network (CNN) training via transfer learning, model evaluation, model persistence, and web-based deployment.

A. Image Dataset Collection

A labeled images datasets comprisings diseased coconuts palms sample acrosses five targeted diseases classes collected and organized into distincts categories folders collecting image captured acrosses diverse angles and plantations backgrounds enables there deeps learnings models to learns realistics visual symptoms associated with there each of the five diseases rather then memorizing uniforms backgrounds.

B. Image Preprocessing and Data Augmentation

Raw agricultural images vary considerably in there resolutions aspect ratio orientation and ambients illumination Prior to models trainings a standardized preprocessings pipelines prepares every images for neurals network ingestions using TensorFlow and Keras preprocessing utilities

- **Image Resizing:** All say pictures are resized to a uniform spatial dimensions compatibles with there CNN input layers so batches can been processed consistently.
- **Pixel Normalization:** Raw pixel intensity values were rescaled to a normalized numericals ranges to stabilizes optimizations accelerates trainings convergences.
- **Data Augmentation:** To expanded effectives training diversity and reduces overfitting real time augmentations transformations includings randoms horizontals and verticals flipping slight rotations width/height shifts and zooms variations are applied exclusively to there trainings images.

C. Dataset Splitting

The curated 5 classes diseases datasets is partitioned into 3 mutually exclusives subset a trainings sets used to learns visuals diseas feature validation set used there monitor generalization did prevented overfitting across trainings epochs and a testings sets reserved strictly for unbiased finals performances evaluations.

D. Convolutional Neural Network and Transfer Learning

A **Convolutional Neural Network (CNN)** enhanced were transfers learnings is employed as there cores classifications architecturs Rather than training deeps convolutionals filters from scratch on a limited agricultural images collection there systems utilizes pre trained convolutionals backbones (such as MobileNetV2 or ResNet) whose earl and intermediate layer already captures generals visuals feature such as edges colors gradients and surfaces texture.

On top of the convolutionals features extractors customs classifications layers comprising poolings layer dropouts regularizations to minimizes overfittings and finals five classes softmax output layer added and fine-tuned on there coconuts diseases datasets During trainings convolutionals filter extracts hierarchicals symptoms patterns such as necrotics spot stems exudates yellowing margins and fungal lesions while there denses classifications head maps there visual feature to there fives targets coconuts diseases classes.

E. Model Evaluation

Once training completes, the model is evaluated on the held-out testing subset using standard multi-class image classification metrics:

- **Accuracy:** Measures the overall proportion of correctly classified coconut disease images across all five categories.
- **Precision:** Evaluates how reliably predicted disease labels correspond to true infections without excessive false alarms.
- **Recall:** Measures the model's ability to detect all actual diseased samples within each of the five classes.
- **F1-Score:** Provides a balanced harmonic summary combining precision and recall across classes.
- **Confusion Matrix:** Visualizes class-by-class predictions against actual labels to identify any visual overlap between similar disease symptoms.

F. Model Saving and Web Deployment

After validation, the trained deeps learning models is saved to disks in Keras/HDF5 models formats soon it can be reuseded without retraining For practicals field accessibilities a lightweight webs applications developed usings Flask on the backend and HTML/CSS on the frontended Users upload an twin of a diseased coconut leaf, crown, or trunk through the webpage, after which the Flask server automatically applies the exact resizing and regularization steps via TensorFlow/Keras, passes the handled image to the loaded CNN model, and displays the predicted disease category alongside preventive management suggestions.

IV. TECHNOLOGY USED

The Coconut Tree Disease Detection System is built on an open-source Python ecosystem integrating deep learning training, image preprocessing, and web deployment.

A. Programming Language and Deep Learning Framework

- **Python:** Serves as the key user interface design dialectal unifying dataset handling, deep neural network training, model evaluation, and backend web serving.
- **TensorFlow and Keras:** Provide the complete deep learning framework for loading and resizing image datasets, applying pixel normalization and real-time data augmentation, loading pre-trained transfer learning architectures, fine-tuning convolutional and dense classification layers, evaluating model performance, and saving the trained model in Keras/HDF5 format.

B. Web Application Framework and Frontend

- **Flask:** A lightweight Python web micro-framework used to host the trained deep learning model, manage image upload requests, run preprocessing and prediction pipelines, and return diagnostic outputs dynamically.
- **HTML and CSS:** Used to build and style the interactive web interface, enabling users to upload coconut tree images, preview selected files, and view predicted disease results and preventive guidelines clearly.

Technology/ Library	Category	Role in the System
Python	Programming Language	Core language for deep learning pipeline and backend logic
TensorFlow & Keras	Deep Learning Framework	Handles image preprocessing, augmentation, transfer learning, CNN training, evaluation, and model saving
Flask	Web Backend Framework	Serves the saved CNN model and handles image upload inference
HTML & CSS	Frontend Technologies	Builds and styles the image upload and diagnosis webpage

Table 1: Summary of Tools and Technologies Used

V. RESULTS AND DISCUSSION

A. Training Convergence and Classification Evaluation

The Coconut Tree Disease Detection System was trained and validated using Python, TensorFlow, and Keras across the five target coconut honor ailment classes Applying standardized resizings pixels normalizations and datas augmentations (randoms rotations, flips, shifts, and zooms) enableded the convolutionals models learns invariant visuals representations of diseases lesions rather than memorizings backgrounds noises.

During trainings both training and validations accuracies loss trends where monitored acrosses epochs to verify smooth convergences and confirms that dropouts regularizations and augmentations effectively prevented overfitting On the unseen testing subset there traineds transfer learnings CNN models was evaluated usings Accuracy Precision, Recall, F1-Score, and Confusion Matrix analysis Assessings per classes precisions and recalls ensure reliables discriminations among visually similars diseases symptoms acrosses the fives coconut diseases classes such as distinguishing gray leafs spots from leaf rot, or bud rot from bud root dropping.

B. Web Application Diagnostic Output

After validatios the traineded modeled was saved and integrated into there Flasks backends paired with an HTML and CSS frontends Through the webs interfaces a farmers or users select and upload a photographs of an affecteds coconuts leafs crowns or stems and submit it for analysiss The Flasks servers automatically resizes and normalizes the uploaded images to matchs there trainings configurations feeds it throughs the pre-loaded CNN models and immediately render there predicted diseases amongs the fives supported classes alongside practicals preventives cares recommendations on there webpage.

C. Discussion and Limitations

The implemented system highlights key advantages of Deep Learning over manual plantation scouting:

- **Rapid and Objective Screening:** Replaces slow, subjective visual inspection with consistent automated classification across the five major coconut diseases within seconds.
- **Strong Generalization with Transfer Learning:** Leveraging pre-trained convolutional feature extractors yields dependable diagnostic performance without requiring massive compute clusters or training deep networks from scratch.
- **Practical Farmer Accessibility:** Deploying the model behind a simple web upload interface eliminates technical barriers for end users.

At the same times certains practicals constraints remains Because there classifiers is trained exclusively on five disease categories input image outsides these conditions are forced into there closests matchings diseases classes Additionally diagnostic accuracies depends on uploaded images beings reasonably focused and well lit extremes motions blurs severes occlusion or distant shots taken from grounded levels toward talls palm canopies can reduce prediction confidence.

VI. CONCLUSION

This papers presented there designs and developments of an automated Systems using Deep Learning to assist farmers in identifying fives critical coconuts palms diseases rapidly and accurately By combining standardized images preprocessing and data augmentations with transfer learning Convolutionals Neurals Networks implemented in Tensor Flow and Keras there systems effectively learns discriminatives visuals features across the 5target diseases categories Evaluated through Confusions Matrix analysis and deployed within an intuitives Flasks using HTML and CSS there platforms enables farmers to upload palms images and receives immediate diseases diagnoses and preventives guidances.

In future work, the system can be expanded in several impactful directions:

- **Drone and UAV Canopy Imaging:** Integrating (UAV) cameras to capture high resolutions crowns and upper frond images of talls matures coconuts palms automatically.
- **Mobile Application with Offline Edge Inference:** Deploying lightweight quantized models (such as TensorFlow Lite) insides an Android/iOS mobiles apps so farmers in there remotes plantations can diagnoses diseases without continuous internets connectivity.
- **Multilingual Advisory and Severity Estimation:** Adding regionals language supports and automated infections severity grading with tailored treatments dosages recommendations.

REFERENCES

1. K. G. Liakos, P. Busato, D. Moshou, S. Pearson, and D. Bochtis, "Machine Learning in Agriculture: A Review," *Sensors*, vol. 18, no. 8, p. 2674, Aug. 2018.
2. S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using Deep Learning for Image-Based Plant Disease Detection," *Frontiers in Plant Science*, vol. 7, p. 1419, Sep. 2016.
3. K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, Feb. 2018.
4. S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification," *Computational Intelligence and Neuroscience*, vol. 2016, Art. no. 3289801, Jun. 2016.
5. Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.
6. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 25, 2012, pp. 1097–1105.
7. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, May 2015.
8. C. Szegedy *et al*., "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1–9.
9. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.
10. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Salt Lake City, UT, USA, Jun. 2018, pp. 4510–4520.
11. M. Abadi *et al.*, "TensorFlow: A system for large-scale machine learning," in *Proc. 12th USENIX Symp. Operating Systems Design and Implementation (OSDI)*, Savannah, GA, USA, Nov. 2016, pp. 265–283.