International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0001
e ISSN: 2584-2854
Volume: 04 Issue: 01
January 2026
Page No: 01 - 07

An Analysis of Advanced Automated Tree Detection and Species Classification Utilizing High-Resolution Imaging and YOLO-NAS

Harjinder Singh[1], *Ranjana Sharma* [2], *Anil Kumar*[3]
[1]*Research scholar, CCSIT, TMU, Moradabad, India.*
[2]*Associate Professor, CCSIT, TMU, Moradabad, India.*
[3]*Scientist/Engineer 'SF', IIRS, Dehradun, India.*
Emails: *harjinder.mca07@gmail.com*[1], *sharmaranjana04@gmail.com*[2], *aniliirsisro@gmail.com*[3]

Abstract

Trees play a crucial role in sustaining our planet by producing oxygen, storing carbon, and offering habitats for wildlife. Artificial intelligence enhances their management by automating the detection of trees and monitoring their health, leading to more intelligent and efficient conservation efforts. Environmental preservation and the prompt and precise combat of climate change are aided by this technology. For monitoring purpose UAV's/ mobile camera/CCTV camera with AI enabled DL based models can achieve the objective discussed. In this study, we investigated DL based YOLO-NAS model's capacity to recognize whole trees in digital photos taken with high-definition or mobile cameras. According to our results, YOLO-NAS successfully detects single trees with high confidence scores and precise bounding boxes. A diverse set of photos from Google and real-time photos taken with Android phones were used to evaluate this strategy. YOLO-NAS recorded mean Average Precision (mAP) around 87.2%, Precision around 88.0%, and Recall of around 80.2% when compared to YOLOv8. However, with a mAP of 88.0%, Precision of 86.9%, and Recall of 85.1%, YOLOv8 fared better than YOLO-NAS. The two models were similarly powerful, with YOLOv8 providing superior recall and YOLO-NAS demonstrating superior precision.

Keywords: *Digital Images, YOLO, Deep learning, Object Detection, Trees Detection.*

1. Introduction

The significance of trees is vital to a healthy world. In addition to providing oxygen, which is essential for both people and animals, they also absorb carbon dioxide, which helps to slow down global warming. Additionally, trees protect soil erosion, maintain a balanced water cycle, and offer a variety of ecosystems, all of which increase biodiversity. They also remove pollutants from the air, beautify and enhance the recreational value of both urban and rural areas, and offer shade [1], [2], [3]. By introducing advanced algorithms to enable precise object classification and detection applied on video and image data, deep learning and artificial intelligence are revolutionizing object recognition. These technologies can also be used to improve tree conservation by developing models to monitor tree health, identify fire hazards, map city forests, detect pests and diseases, analyze data for conservation strategies, engage with the public, and track carbon sequestration and the climate effect [4]. Using high-resolution imagery and YOLO-NAS, this project aims to create an Advanced Automated Tree Detection and Species Classification Framework. Using high-definition images captured by UAVs and digital cameras, the framework will attempt to efficiently identify individual trees and categorize their species. A real-time object detection model named YOLO-NAS can effectively recognize and differentiate tree canopies in challenging environments due to its neural architecture search, which improves performance across various lighting conditions and backgrounds [5]. Tree conservation initiatives have been considerably aided by the integration of deep learning and AI techniques, allow

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0001
e ISSN: 2584-2854
Volume: 04 Issue: 01
January 2026
Page No: 01 - 07

to have fundamentally altered the accurate monitoring and control of trees. With the use of sophisticated models like YOLO, these high-tech devices can distinguish between different kinds of trees, identify health issues, and forecast the likelihood of pest or disease outbreaks [6]. They enable accurate mapping of urban forests, assessment of fire danger, and comprehensive analysis of the effects of climate change through sophisticated data analysis and simulations. Additionally, AI encourages public involvement and advances research by analyzing large data. [7]. The following sections have been scheduled for the remainder of the paper. The study's literature is briefly reviewed in the II part, which also serves as an introduction to the related work. The third section provides a summary of the methodology used for data collection, preprocessing, model selection, training, and evaluation. Results and discussion were presented in Section IV. The study's future scope and sources are listed in Sections V and VI

2. Related Works

A lot of work has been done towards object detection using deep learning algorithms in UAV and Satellite imagery. Here is a short literature review of the research studies based on object detection using YOLO family algorithms. Zhengyang Zhong et.al. (2024) showcased that the recent advancements in fruit detection have driven the need to balance computing efficiency and accuracy. Models like YOLOv5, YOLOv6, YOLOv7, and YOLOv8 have demonstrated improved real-time object detection capabilities. The Light-YOLO model builds on this progress by incorporating structural upgrades, including a revised Bottleneck and EMA attention mechanism, as well as enhancements to the Darknet53 backbone, such as bidirectional and skip connection modules and a decreased channel neck. With a significantly low parameter count (1.96 M) and FLOPs (3.65 G), On the ACFR Mango dataset, these enhancements effectively balance model complexity and performance, achieving mean Average Precision (mAP) as 64.0% and a mAP0.5 as 96.1%. Vasileios Moysiadis et.al. (2024) studied advancements in machine learning that improved the capacity for object identification. Models such as Detectron2 and YOLOv8 have demonstrated notable efficacy in detecting individual trees and generating precise masks. In the domain of cherry tree detection, both models have achieved noteworthy F1 scores of up to 94.85%. The refinement of masks using OTSU thresholding has substantially enhanced accuracy, resulting in an impressive Intersection over Union (IoU) of 85.30%, surpassing the scores of Detectron2 and YOLOv8. Aman et.al. (2023) introduced an object detection system for forest monitoring that is based on the state-of-the-art and highly efficient "YOLO-NAS" (You Only Look Once Neural Architecture Search) technology. The model named as YOLO-NAS offers faster and more accurate results, as well as automates model design, in contrast to existing models. Md. JanibulAlam Soeb et. al. (2023) This research endeavors to propose an artificial intelligence-driven solution for disease detection in tea leaf with the help of YOLOv7, as it is fastest single-stage object identification model, using the dataset of tea leaves which are diseased sourced from four major tea estates in Bangladesh. The dataset encompasses photos of five different types of leaf diseases, each meticulously annotated and augmented to mitigate the limitations posed by small sample sizes. Employing well-established statistical measures, the outcomes of the identification and detection process are assessed. The precision, recall mAP value & F1-score were appeared to be 96.5%, 97.3%, 96.7%, 96.4%, and 98.2% respectively, contributing the efficacy of the YOLOv7. Jakub Pawłowski et.al. (2024) Rich annotations regarding the locations and characteristics of coffee bean and white bean seed images were integrated into a database. The locations, sizes, and kinds of the seeds were ascertained using image processing techniques using You Only Look Once v8 (YOLO) models. To confirm the effectiveness and efficiency of the many approaches employed, a thorough evaluation was conducted. The findings demonstrated that the best training Convolutional neural network (CNN) model achieved an average size error of 0.58 mm for the seeds and a segmentation accuracy of 90.1% IoU. MARTINUS GRADY NAFTAL et.al. (2024) This study evaluates the performance of several YOLO models and other object detection frameworks using

e ISSN: 2584-2854
Volume: 04 Issue: 01
January 2026
Page No: 01 - 07

a newly developed dataset featuring oil palm fruit bunches. The dataset was collected from plantations in Central Kalimantan Province, Indonesia, and includes five ripeness categories: abnormal, ripe, underripe, unripe, and flower. The dataset also presents typical annotation challenges, such as partial object visibility, low-contrast imagery, occlusions, small object sizes, and image blurriness. Among the evaluated models—YOLOv6s, YOLOv6l, YOLOv7 Tiny, YOLOv7l, YOLOv8s, and YOLOv8l—the YOLOv8s Depth wise model stood out. It achieved excellent performance with a compact size of just 10.6 MB, a rapid inference time of 0.027 seconds, and strong detection metrics (mAP50: 0.75 and mAP50–95: 0.481). Additionally, its efficient training process, which converged in just 2 hours, 18 minutes, and 30 seconds, further highlights its effectiveness. With the help of Table 1, we have tried to give brief outcomes of this literature survey.

3. Methodology Used

Most researchers have adopted the following methodology depicted in Figure 1 for object detection with digital images, remote sensing, and UAV images. This flowchart outlines a comprehensive process for developing an advanced automated system for categorizing species and identifying trees using high-resolution images and "YOLO-NAS" (You Only Look Once - Neural Architecture Search). The process begins with data pre-processing, which involves enhancement, augmentation, and annotation. Next, the model selection stage is followed by model training, including the development of loss functions, feature extraction, detection head implementation, and optimization. After training process , the model is evaluated using various metrics and validation methods. Post-processing techniques such as bounding box refinement and non-maximum suppression are utilized to improve results. Finally, the model is implemented, and the outcomes are displayed, completing the process [3], [4], [7], [8], [9].

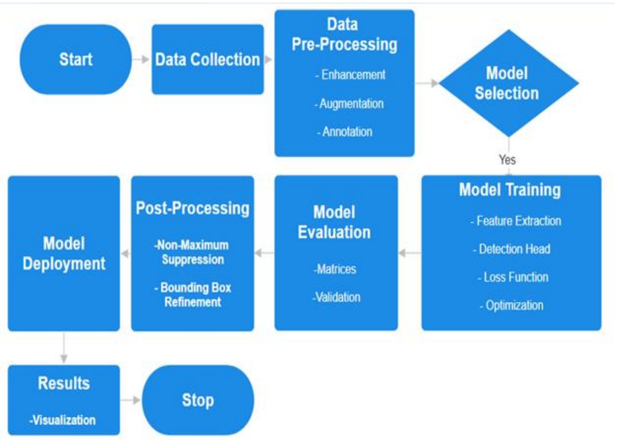

Figure 1 Methodology used (flow from Left to right) [9], [10], [11], [12], [13]

3.1 Dataset Used

We have extracted 3700 images from different free sources on the internet of 4 classes i.e. Acacia, Palm Trees, Papaya Trees, and Mango trees. The median Image size ratio used in this dataset is 480X640. Figure 2 showcased the glimpses of images of the dataset used for this study.

Papaya tree

Palm tree

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0001
e ISSN: 2584-2854
Volume: 04 Issue: 01
January 2026
Page No: 01 - 07

Mango tree

Acacia Tree

Figure 2 A few Sample images taken from the dataset used for this study

Table 1 Quick Literature Review

Author(s) and Year	Focus Area	Key Models/Technologies	Performance Metrics
Zhengyang Zhong et al.(2024)	Fruit detection	YOLOv5, YOLOv6, YOLOv7, YOLOv8, Light-YOLO, YOLO-NAS	Parameter count: 1.96 M, FLOPs: 3.65 G, mAP: 64.0%, mAP0.5: 96.1%
Vasileios Moysiadis et al. (2024)	Tree detection and mask generation	Detectron2, YOLOv8, YOLO-NAS	F1 score: up to 94.85%, IoU: 85.30%
Aman et al. (2023)	Forest monitoring	YOLO-NAS	Not specified
Md. JanibulAlamSoeb et al. (2023)	Tea leaf disease detection	YOLOv7, YOLO-NAS	Precision: 96.5%, Recall: 97.3%, mAP: 96.7%,
Jakub Pawłowski et al. (2024)	Seed analysis (coffee and white bean seeds)	YOLOv8, YOLO-NAS	IoU: 90.1%, Average size error: 0.58 mm
MARTINUS GRADY NAFTAL et al. (2024)	YOLO models and object detection technologies	YOLOv6s, YOLOv6l, YOLOv7 Tiny, YOLOv7l, YOLOv8s, YOLOv8l, YOLOv8s Depth wise, YOLO-NAS	mAP50: 0.75, mAP50-95: 0.481, Training Time: 2 hours, 18 minutes, 30 seconds

3.2 Data Preprocessing

Data Augmentation: data augmentation is a crucial method for expanding datasets by creating various copies of existing pictures. Common augmentation techniques for tree identification tasks include cropping, resizing, rotating, flipping, and adjusting photos to change their colors. The dataset can also be enhanced by adding noise, using blurring techniques, changing perspectives, combining photos, and randomly removing parts of images. we have applied rotation, flipping, and resizing techniques and extended the dataset from 3700 images to 4541 images.

Data Annotation: With the use of labelImg tool an open-source tool of Python, we have annotated around 6205 objects in the used dataset. Figure 3 represents about the class balance of the dataset.

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0001
e ISSN: 2584-2854
Volume: 04 Issue: 01
January 2026
Page No: 01 - 07

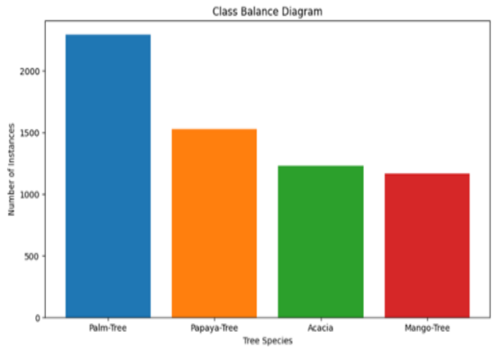

Figure 3 Class Balance Chart

3.3 Model Selection

"YOLO-NAS": YOLO-NAS (You Only Look Once - Neural Architecture Search) is an advanced version of YOLO object detection framework that sums up neural architecture search (NAS) techniques. Key features of YOLO-NAS include automated architecture optimization, hardware-aware design, dynamic depth and width scaling, compound scaling, efficient feature extraction, adaptive receptive field, anchor-free detection, multi-scale feature fusion, attention mechanisms, and optimized loss functions. These features make YOLO-NAS particularly effective for complex object detection tasks by balancing accuracy, speed, and adaptability. The YOLO-NAS architecture combines the flexibility of Neural Architecture Search with the effective single-stage detection method of YOLO. It features a customizable backbone network discovered using NAS, which is optimized for computational efficiency and feature extraction. Neck structure of network performs multi scale feature fusion, enhancing the ability of model to identify objects of various sizes. By involving feature maps, the detection head employs an anchor-free approach to predict object positions and classes. YOLO-NAS utilizes compound scaling and attention mechanisms to balance depth, breadth, and resolution, focusing on important elements for improved performance. By combining these components, we have created an exceptionally versatile and powerful object detection framework that excels in tasks such as species categorization and tree identification.

3.4 Training and Validation

The YOLO-NAS model was adapted for training on the specified dataset using Google Colab, Python 3, the Keras&Tensorflow, PyTorch library, and the analysis of the outcomes. A Google Colab Python environment outfitted with a Tesla T4 GPU (15102 MiB), two CPUs, and 12.7 GB of RAM was used to conduct the trials. The model training was performed with the following parameters: a batch size of 32, using input images sized 480x640 pixels. The learning rate was set to 0.01 with a momentum of 0.937 to stabilize and accelerate convergence. The Intersection over Union (IoU) training threshold was configured at 0.20 to determine positive object detections. Additionally, image augmentation included rotation adjustments ranging from -15 to 15 degrees, enhancing the model's robustness by introducing variability in the training data. We have gone for 200 epochs to perform training. These settings aim to optimize model performance and generalization. Table 2 represents the evaluation test that we performed on the model to evaluate it.

Table 2 **Evaluation Matrices [10], [14], [11], [12], [13]**

Performance Evaluation Tests	Formulas
Precision	TP/(TP+FP)
Recall	TP/(TP+FN)
Accuracy	(TP+TN)/(TP+TN+FP+FN)
TP= True Positive, TN= True Negative FP= False Positive, FN= False Negative	

4. Results & Discussions

YOLO -NAS model went for a training of 200 epochs and came up with 87.2% mAP, 88.0% precision and 80.2 % recall. Figure 4 shows a complete set of performance metrics for an object detection model. It is likely that the model uses the YOLO-NAS architecture for tree detection and classification. The graphs display evaluation metrics over training iterations or epochs, as well as training and validation losses. The top row shows training losses for bounding box regression (box_loss), classification (cls_loss), and a combination of detection factors (dfl_loss). These losses consistently decrease, indicating improved model performance during training. The accuracy and recall curves for class B

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2026.0001
e ISSN: 2584-2854
Volume: 04 Issue: 01
January 2026
Page No: 01 - 07

(possibly a specific tree species) suggest effective learning of class-specific traits, with a quick initial improvement followed by continuous gains.The bottom row displays validation losses, which mirror the training metrics. The mAP50 and mAP50-95 graphs show the model's overall detection performance, with values stabilizing at around 0.8 and 0.5 respectively, demonstrating robust object detection capabilities across different overlap criteria. Figure 5 depicts the successful detection and classification of Mango trees and Palm trees with convincing confidence scores i.e. 76% and 87%. The Mango tree confidence score is a bit lower because we have fewer images of mango trees in the training set of the dataset.

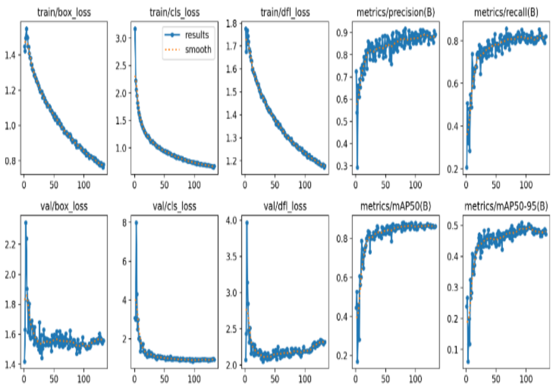

Figure 4 Training Graphs

Performance Comparison with Other Existing Models: Figures 6& 7 shows the comparison of the YOLO-NAS and YOLOv8 model's performance on this dataset. In precision, YOLO-NAS performance is higher side.In the case of Recall, YOLOv8 's performance is outstanding.

Figure 5 output sample images with Individual tree identification and classification

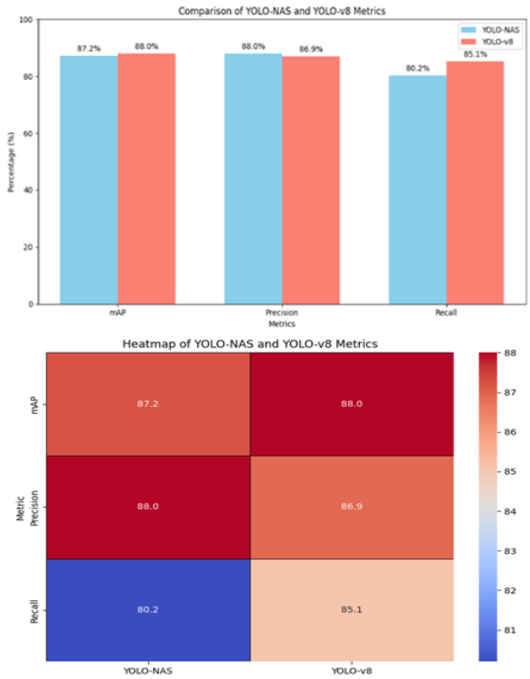

Figure 6, 7 Bar Chart & heat map of performance comparison of YOLO-NAS and YOLO v8

Conclusion &Future Work

In this paper, we present a method using existing model YOLO-NAS for detecting the whole tree as a object in a digital image taken from a mobile camera or high definition digital camera. We evaluate our method on many images randomly picked from google and also taken live photos from android phone and the model showed that it is feasible to detect them and provide individual boundary boxes with good confidence score. Further, we evaluated the performance of YOLO-NAS and YOLOv8 on the dataset used for this study. YOLO-NAS 's mAP is

87.2%, Precision is 88.0% and Recall is 80.2%. In the case of YOLOv8's mAP is 88.0%, Precision is 86.9% and Recall is 85.1%. Both models are performing well. Better precision will be offered by YOLO -NAS and recall by YOLOv8 and mAP YOLOv8 is ahead with .8% margin with YOLO-NAS. In future work, more images can be used for the training of the model so that accuracy can be improved. For all sizes of objects present in the images and overlapping of objects in the images could be improve by doing some hybridization of YOLO's latest variants with some other networks like CENTERNET and all to improve and propose a feasible model for tree detection [11], [15].

References

[1]. J. Zheng, W. Li, M. Xia, R. Dong, H. Fu, and S. Yuan, "Large-Scale Oil Palm Tree Detection from High-Resolution Remote Sensing Images Using Faster-RCNN," in IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium, Jul. 2019, pp. 1422–1425. doi: 10.1109/IGARSS.2019.8898360.

[2]. P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma, "A Review of Yolo algorithm developments," Procedia Comput. Sci., vol. 199, pp. 1066–1073, 2022.

[3]. Y. Chen, H. Xu, X. Zhang, P. Gao, Z. Xu, and X. Huang, "An object detection method for bayberry trees based on an improved YOLO algorithm," Int. J. Digit. Earth, vol. 16, no. 1, pp. 781–805, 2023.

[4]. L. Shen, B. Lang, and Z. Song, "CA-YOLO: Model optimization for remote sensing image object detection," Ieee Access, 2023.

[5]. D.-L. Pham, T.-W. Chang, and others, "A YOLO-based real-time packaging defect detection system," Procedia Comput. Sci., vol. 217, pp. 886–894, 2023.

[6]. T. Diwan, G. Anirudh, and J. V. Tembhurne, "Object detection using YOLO: Challenges, architectural successors, datasets and applications," Multimed. Tools Appl., vol. 82, no. 6, pp. 9243–9275, 2023.

[7]. U. Sirisha, S. P. Praveen, P. N. Srinivasu, P. Barsocchi, and A. K. Bhoi, "Statistical analysis of design aspects of various YOLO-based deep learning models for object detection," Int. J. Comput. Intell. Syst., vol. 16, no. 1, p. 126, 2023.

[8]. J. Terven, D.-M. Córdova-Esparza, and J.-A. Romero-González, "A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas," Mach. Learn. Knowl. Extr., vol. 5, no. 4, pp. 1680–1716, 2023.

[9]. C. Zhao, X. Shu, X. Yan, X. Zuo, and F. Zhu, "RDD-YOLO: A modified YOLO for detection of steel surface defects," Measurement, vol. 214, p. 112776, 2023.

[10]. [10] "Recognition and counting of oil palm tree with deep learning using satellite image - IOPscience." Accessed: Feb. 19, 2024. [Online]. Available: https://iopscience.iop.org/article/10.1088/1755-1315/974/1/012058/meta

[11]. H. Lai, L. Chen, W. Liu, Z. Yan, and S. Ye, "STC-YOLO: small object detection network for traffic signs in complex environments," Sensors, vol. 23, no. 11, p. 5307, 2023.

[12]. M. Sportelli et al., "Evaluation of YOLO object detectors for weed detection in different turfgrass scenarios," Appl. Sci., vol. 13, no. 14, p. 8502, 2023.

[13]. X. Zhai, Z. Huang, T. Li, H. Liu, and S. Wang, "YOLO-Drone: an optimized YOLOv8 network for tiny UAV object detection," Electronics, vol. 12, no. 17, p. 3664, 2023.

[14]. Z. Situ, S. Teng, X. Liao, G. Chen, and Q. Zhou, "Real-time sewer defect detection based on YOLO network, transfer learning, and channel pruning algorithm," J. Civ. Struct. Health Monit., vol. 14, no. 1, pp. 41–57, 2024.

[15]. S. N. Appe, G. Arulselvi, and G. Balaji, "CAM-YOLO: tomato detection and classification based on improved YOLOv5 using combining attention mechanism," PeerJ Comput. Sci., vol. 9, p. e1463, 2023.