JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Tomato Triage: Tomato Fruit Disease Detection and Pesticide Strategies using Deep Learning

[1]Vinay Kumar L, [2]K R Sumana

PG Student, The National Institute of Engineering, Mysuru, Visweswaraya Technological University, Belagaum, Karnataka, India[1]

Faculty, The National Institute of Engineering, Mysuru, Visweswaraya Technological University, Belagaum, Karnataka, India[2]

Abstract : This project leverages advanced deep learning techniques to enhance tomato farming by delivering precise disease detection and pesticide recommendations. This project addresses the crucial need for timely and accurate identification of tomato fruit diseases, which significantly affect crop yield and quality. Central to Tomato Triage is the Tomato Fruit Disease Classification system, developed using the Yolov8 algorithm. By employing image processing and machine learning, the system accurately identifies and classifies common tomato diseases, providing suitable pesticide recommendations. The dataset, sourced from the Mysuru RMC market, includes images labelled with five disease categories: Anthracnose, Bacterial Spot, Botrytis, Fruit worm Bore, and Tomato Cracking. The project utilized Google Collab for model training and Flask for deployment, with Python as the primary programming language. This initiative not only advances agricultural technology but also equips farmers with a valuable tool to improve crop management and yield.

Keywords: Deep Learning, Yolov8 Algorithm , Machine Learning, Anthracnose, Bacterial Spot, Botrytis, Fruit worm Bore, Tomato Cracking, Google Collab, Flask .

I. INTRODUCTION

Agriculture is the cornerstone of the Indian economy, providing livelihoods for a large portion of the population. Tomatoes are particularly important due to their widespread culinary use and nutritional benefits. However, tomato crops are frequently threatened by various diseases, affecting both productivity and quality. Early detection and accurate diagnosis of these diseases are essential for effective management and preventing significant crop losses. Traditional methods of disease detection rely on manual inspections by experts, which are often time-consuming, labor-intensive, and prone to errors. Recent advancements in image processing and machine learning have made automated disease detection systems increasingly viable. This project aims to develop an automated Tomato Disease Classification system using the Yolov8 algorithm, a state-of-the-art object detection model. The system classifies tomato fruit diseases into five categories: Anthracnose, Bacterial Spot, Botrytis, Fruit Worm Bore, and Tomato Cracking. The dataset, sourced from the Mysuru RMC market, provides a diverse and representative collection of images. Utilizing the Yolov8 algorithm, the model delivers accurate and efficient disease classification and pesticide recommendations, aiding farmers and agricultural professionals in timely and effective disease management.

II. Literature Survey

The literature survey highlights various advancements in tomato disease detection and classification. Liu J and Wang X et al. [1]enhance YOLO V3 for real-time detection, integrating multi-scale feature detection and object frame dimension clustering for superior accuracy and speed. Q. Wang and F. Qi et al.[2] utilize Faster R-CNN with CNNs like VGG16 and ResNet for efficient tomato disease detection, reducing workload and improving precision. H. Hong, J. Lin, and F. Huang et al.[3] review big data, AI, and IoT advancements, emphasizing their significance across sectors like healthcare and finance, and the need for interdisciplinary collaboration. J. Zhao and J. Qu et al. [4]employ YOLOv2 with K-means clustering for detecting physiological diseases in tomatoes, focusing on accurate disease detection for quality control. Behera, Santi & Mahapatra, Abhijeet & Rath, Amiya & Sethy, Prabira et al. [5]compile techniques for tomato classification and grading, highlighting the need for automated systems and comparing various classification methods such as ANN, SVM, and random forest classifiers.

III. PROPOSED SYSTEM

The proposed system uses the Yolov8 algorithm to develop an advanced Tomato Disease Classification model, known for its high accuracy and efficiency. The development process includes several key stages: data collection, preprocessing, model training, evaluation, and deployment. The dataset, sourced from the Mysuru RMC market, contains images categorized into five disease types: Anthracnose, Bacterial Spot, Botrytis, Fruit Worm Bore, and Tomato Cracking. Preprocessing steps, such as noise reduction and contrast adjustment, enhance image quality. The Yolov8 model is trained on this labeled dataset, employing techniques like data augmentation to ensure robustness. After training, the model is rigorously evaluated using metrics such as accuracy, precision, recall, and F1-score to validate its performance. Upon achieving satisfactory results, the model is deployed using Flask, providing a web-based interface for real-time disease detection. This system integrates advanced machine learning with practical agricultural applications, offering a valuable tool for improving tomato crop management through precise and timely disease identification.

IV. METHODOLOGY

The system processes uses the YOLOv8 model from Ultralytics. Images are validated and stored, then processed by the YOLO module to identify diseases with confidence scores. The detection results are annotated on the images using cvzone, displaying bounding boxes and disease information. Uploaded and processed images are saved securely, and users receive detailed feedback on disease detection, including visual annotations and treatment recommendations. The processed images and results can be viewed via a specific route in the application.

Input Module

The input module is responsible for handling and validating the images uploaded by users. The Flask application is configured to store uploaded files in the static/uploads/ directory, and the allowable file types include PNG, JPG, JPEG, GIF, and BMP. The function checks if the uploaded file has one of these extensions. This ensures that only valid image files are processed. The input module also includes routes and methods to render the upload form, handle image submissions, and provide feedback to users regarding the success or failure of their uploads.

YOLO Module

The YOLO (You Only Look Once) module integrates the YOLOv8 model from Ultralytics for real-time object detection. This model is initialized with pre-trained weights (best.pt), making it capable of identifying specific tomato diseases. In the function , the YOLO model processes the uploaded image and outputs bounding boxes around detected objects. Each detection includes a confidence score and a class label, which indicates the type of disease identified. The model is configured to run with a confidence threshold of 0.05, ensuring that even low-confidence detections are considered.

Detection Module

The detection module processes the uploaded image to identify tomato diseases. The function reads the image using OpenCV and passes it to the YOLO model for detection. The results from the model include bounding boxes, confidence scores, and class labels. A dictionary class Names maps these labels to specific disease names and their corresponding treatments. This module compiles the detection results, which are later used to annotate the image and generate a detailed report for the user.

Annotation Module

The annotation module is responsible for visually marking the detected diseases on the uploaded image. It uses the cvzone library to draw bounding boxes around detected objects and to add text labels indicating the disease name, recommended treatment, and confidence score. The functions are used for drawing rectangles and placing text on the image, respectively. This visual feedback helps users easily identify the affected areas and understand the recommended treatments for each detected disease.

Upload Module

The upload module handles the entire process of receiving, validating, and saving the uploaded images. When a POST request is made to the route, the module checks for the presence of a file and validates its extension using the function. If the file is valid, it is saved securely in the UPLOAD_FOLDER using the function from Werkzeug. The saved file path is then passed to the video_detection function for processing. If the file is missing or invalid, appropriate flash messages are displayed to the user.

Result Saving Module

The result saving module manages the storage and presentation of processed images. After the YOLO model detects diseases and the image is annotated, the resulting image is saved in the UPLOAD_FOLDER with a predefined filename (result.jpg). The OpenCV function cv2.imwrite is used to write the processed image to disk. This module also prepares the detection results, which include the disease names, recommended treatments, and confidence scores, to be passed to the template for rendering. The /display/<filename> route allows users to view the processed image by retrieving it from the upload directory. This ensures that users can see both the annotated image and the detailed detection results, providing a comprehensive output of the disease detection process.

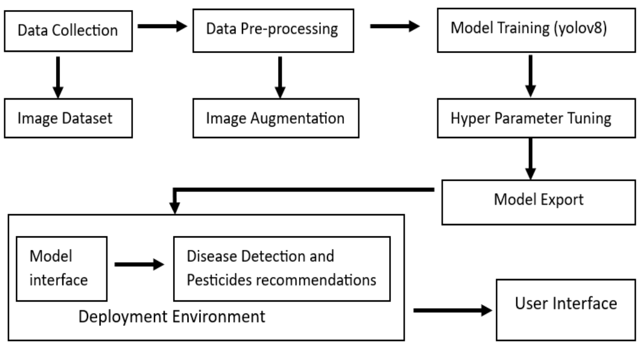

Figure 1 System Architecture

The system features a client-side interface where users can upload images of tomato plants for analysis. On the server side, the back end handles image processing and disease classification using the Yolov8 algorithm. This involves the integration of the pre-trained Yolov8 model specifically for identifying various tomato diseases. Once the image is processed and classified, the results, including disease identification and confidence scores, are displayed to the user through a dedicated result display component, providing a seamless and informative user experience

V. Result Analysis

The system's findings and strategies for addressing tomato fruit diseases, focusing on the image upload module and the disease detection and pesticide recommendation module. In the image upload module, users can upload images in allowed formats (png, jpg, jpeg, gif, bmp), which are validated and saved to the UPLOAD_FOLDER. Successful uploads are processed using the video_detection function, which detects diseases with the YOLO model. The processed image, with annotations, is saved as result.jpg, and users can view it by navigating to a specific URL, with the display_image function serving the image.

The disease detection module leverages the video_detection function, which reads the uploaded image using OpenCV and processes it with the YOLO model loaded with specified weights (best.pt). The model detects various tomato diseases, providing each detection with a disease name, recommended treatment, and confidence score. The function annotates the image with bounding boxes and labels for detected regions, saving the annotated image as result.jpg. Detection results and the annotated image are then displayed on the home.html template for user review.

Users benefit from a seamless interface that allows image uploads and detailed disease detection results. Upon uploading a valid image, it is processed to detect diseases, annotated with relevant details, and saved for user viewing. The system ensures users receive comprehensive feedback, including visual annotations and treatment recommendations, all accessible via a straightforward URL, enhancing the overall user experience and aiding in effective disease management.

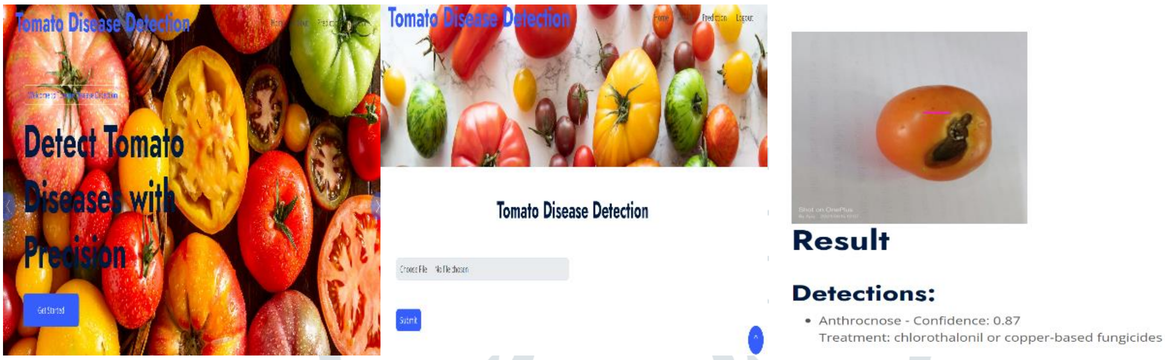

Figure 2 web interface

The confidence score that YOLOv8 outputs is a combination of these two confidences, which enables it to balance between how certain it is that a box contains an object and how certain it is about which class this object belongs to. As for your question regarding metric curves, you're correct that the confidence score is often used as the X-axis.

VI .Acknowledgment

I extend my heartfelt gratitude to The National Institute of Engineering, Mysuru, and its dedicated staff for their unwavering support. A special thanks to my guide, Smt. K R Sumana, whose guidance and encouragement were invaluable throughout this journey. I am deeply appreciative of my supportive parents, friends, and classmates, whose encouragement and support played a crucial role. Finally, I extend my sincere thanks to everyone who contributed, both directly and indirectly, to the successful completion of my project.

References

[1] Liu J and Wang X (2020) Tomato Diseases and Pests Detection Based on Improved Yolo V3 Convolutional Neural Network. Front. Plant Sci. 11:898. doi: 10.3389/fpls.2020.00898

[2] Q. Wang and F. Qi, "Tomato Diseases Recognition Based on Faster RCNN," 2019 10th International Conference on Information Technology in Medicine and Education (ITME), Qingdao, China, 2019, pp. 772-776, doi: 10.1109/ITME.2019.00176

[3] H. Hong, J. Lin and F. Huang, "Tomato Disease Detection and Classification by Deep Learning," 2020 International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), Fuzhou, China, 2020, pp. 25-29, doi: 10.1109/ICBAIE49996.2020.00012.

[4] J. Zhao and J. Qu, "A Detection Method for Tomato Fruit Common Physiological DiseasesBased on YOLOv2," 2019 10th International Conference on Information Technology in Medicine and Education (ITME), Qingdao, China, 2019, pp. 559-563, doi: 10.1109/ITME.2019.00132.

[5] Behera, Santi & Mahapatra, Abhijeet & Rath, Amiya & Sethy, Prabira. (2019). Classification & Grading of Tomatoes using Image Processing Techniques

[6] S Mohana Saranya1 , R R Rajalaxmi1 , R Prabavathi2 , T Suganya3 , S Mohanapriya1 and T Tamilselvi1" Deep Learning Techniques in Tomato Plant – A Review" S Mohana Saranya et al 2021 J. Phys.: Conf. Ser. 1767 012010 , DOI 10.1088/1742- 6596/1767/1/012010

[7] Ravi Shankar, Seema Harsha, Raj Bhandary "A Practical Guide To Identification and Control of Tomato Diseases" R& D dept,

TROPICA SEEDS PVT LTD | No 54,South End Road, 1st Floor, NamaAurore Building, Basavangudi, Bangalore 560004 INDIA

[8] Afonso M, Fonteijn H, Fiorentin FS,Lensink D, Mooij M, Faber N, Polder G and Wehrens R (2020) Tomato Fruit Detection and Counting in Greenhouses Using Deep Learning. Front.
Plant Sci. 11:571299. doi: 10.3389/fpls.2020.571299

[9] Omidi-Arjenaki, Omid & Moghaddam, Parviz & Motlagh, Asaad. (2012). "Online tomato sorting based on shape, maturity, size, and surface defects using machine vision". Turkish Journal of Agriculture and Forestry. 37. 62-68. DOI 10.3906/tar-1201-10

[10] Chen, H.-C.; Widodo, A.M.;Wisnujati, A.; Rahaman, M.; Lin,J.C.-W.; Chen, L.,Weng,C.-E. AlexNetConvolutional Neural Network forDisease Detection and Classification of Tomato Leaf. Electronics 2022, 11, 951. https://doi.org/10.3390/

[11] Hanssen, Inge & Lapidot, Moshe & Thomma, Bart. (2010). Emerging Viral Diseases of Tomato Crops. Molecular plant-microbe interactions : MPMI. 23. 539-48. DOI 10.1094/MPMI-23-5-0539.

[12] Chen, Z.; Wu, R.; Lin, Y.; Li, C.; Chen, S.; Yuan, Z.; Chen, S.; Zou, X. Plant Disease Recognition Model Based on Improved YOLOv5. Agronomy 2022, 12, 365. doi.org/10.3390/agronomy12020365

[13] Natarajan, V. & Babitha, Macha & Kumar, M.. (2020). Detection of disease in tomato plant using Deep Learning Techniques.International journal of modern agriculture,Volume 09,No 4,2020 ISSN 2305-7246

[14] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

[15] Abd Murad, Nur & Kusai, Nor & Mohd Zainudin, Nur Ain Izzati. (2016). Identification and diversity of Fusarium species isolated from tomato fruits. Journal of Plant Protection Research. 56. DOI 10.1515/jppr-2016-0032