Research Paper
Impact Factor: 7.352
Peer Reviewed Monthly Journal
www.ijmdrr.com

IJMDRR
E- ISSN –2395-1885
ISSN -2395-1877

A REVIEW ON IDENTIFICATION OF PLANT LEAVES OF KARNATAKA WESTERN GHAT

Dr.Keshava Prasanna*   Dr. Thungamani M Anantha Krishna*****

**Department of Social & Allied Sciences, KSNUAHS, Iruvakki, Shivamogga, Karnataka.*
***Department of Social & Allied Sciences, UHS, Bagalkot, Karnataka.*
****Department of Social & Allied Sciences, KSNUAHS, Iruvakki, Shivamogga, Karnataka.*

Abstract

A Novell approach of identification of Plant Leaves in the Western Ghats of Karnataka is very important in the societal community and the need of every individual in the form of medicine, food etc. The method of identification of plant leaves is area of research which has wider impact and analysis. However, advancements in computer and information technology have introduced image recognition in agriculture. Identification of different leaves has applied machine vision algorithms to analyse features like colour, shape, and texture in images, feeding these into classifiers for accurate recognition. Recently, the YOLOv5 series object identification model has gained popularity in agriculture due to its high accuracy, speed, short training time, and low computational needs. Using YOLOv5, a model is helpful to identify plant leaves. Precision is higher for Healthy leaves, but recall for unhealthy leaves decreases as confidence increases.

Keywords: Plant Leaf Identification, Object Detection, Deep-Learning, Yolov5, Darknet.

Introduction

Identification of plant leaf accurately and promptly is essential for reducing economic consequences and maximizing crop yield. However, farmers' dependence on conventional manual techniques presents a difficulty in accurately pinpointing particular identification. This research investigates the utilization of the YOLOv5 algorithm for detecting and identifying plant leafs. This study uses the comprehensive Plant Village Dataset of Karnataka Western Ghats, which includes over one thousand photos of healthy plant leaves from different species, to develop advanced identification prediction systems in agriculture. Data augmentation techniques including histogram equalization and horizontal flip were used to improve the dataset and strengthen the model's resilience. Anassessment of the YOLOv5 algorithm to be performed which involves comparing its performance with established target identification methods including Densenet, Alexanet, and neural networks. This study's results demonstrate substantial advancements in plant leaf identification and underscore the capabilities of YOLOv5 the sophisticated tool for accurate predictions. These developments have significant significance for everyone involved in agriculture, researchers, and farmers, providing improved capacities for identification and crop protection.

The Following Are the Primary Contributions of This Research Paper

1. This study applies YOLOv5, a cutting-edge object detection framework, to plant pathology. The work improves plant leaf identification accuracy and efficiency by seamlessly incorporating YOLOv5, providing a fresh method to address agricultural concerns.
2. The study achieves high precision in recognizing and classifying various plant leaf thorough image annotation and data preprocessing. The suggested methodology's robustness provides accurate identification, assisting farmers and researchers in timely intervention and crop management.

3. The study uses the widely accepted Plant Village dataset, which helps validate and benchmark the proposed model. This dataset selection improves the generalizability of the findings, allowing for more excellent applications in plant pathology research.
4. The research provides valuable insights into the practical implementation of the YOLOv5 architecture for detecting plant leaf disease. The precise methodology can help researchers and practitioners implement and adapt this technology in real-world agricultural context.

Literature Survey

Kauret. al., 2022, use shape features on private data set using SVM with accuracy 95%, Logistic regression (LR) with 95%, KNN with 90% and RF with 93%. Yang, 2021, uses shape and texture features on Flavia dataset with an accuracy of 99.10%, Swedish with 98.40% and MEW2012 with 95.60%. Akteret. al., 2020, use high level features using Convolution Neural Network for identifying 10 Bangladeshi medicinal plants with an accuracy of 71.3%. Asghariet. al., 2019, use texture features on Image CLEF 2012, Leaf snap and Flavia with accuracies of 88.80%, 74.50% and 98.70% respectively. Turkogluet. al., 2019, use color and texture features on Flavia, Swedish, ICL and Foliage datasets with accuracies 98.94%, 99.46%, 83.71 and 92.92% respectively. Choudhuryet. al., 2018, use shape features on Leafs nap with accuracy of 84.40% and on Flavia with 98.40%. Araujoet. al., 2017, uses shape features on Image CLEF 2011 and ImageCLEF2012 datasets with accuracies of 86.20% and 64.10% respectively. Nareshet. al. 2016, use modified LBP and achieve an accuracy of above 90% on public datasets and about 78% accuracy on UOM Medicinal Plant leaf datasets. In the context of precision agriculture, there is a strong need for sophisticated techniques that improve detection efficiency because plant diseases pose significant risks to crop health and the field (Attallah, 2023; Rahman et al., 2023; Thangavel et al., 2023). In computer vision for plant identification, deep learning— specifically, the YOLOv5 architecture—has proven to be a potent tool. Deep learning holds great potential to overcome obstacles associated with various illness kinds and environmental variables (Javidan et al., 2023). Previous research has demonstrated the drawbacks of standard approaches. The release of YOLOv5 improves object identification speed and accuracy, which makes it an excellent option for applications in plant pathology (Singh et al., 2017). An in-depth analysis of the current research environment is provided by this literature review, which sets the stage for a later investigation into the use of YOLOv5 in identifying and detecting plant leaf diseases (Perveen et al., 2023).

Objectives

The methods proposed to identify the leaf using an image analysis-based approach.

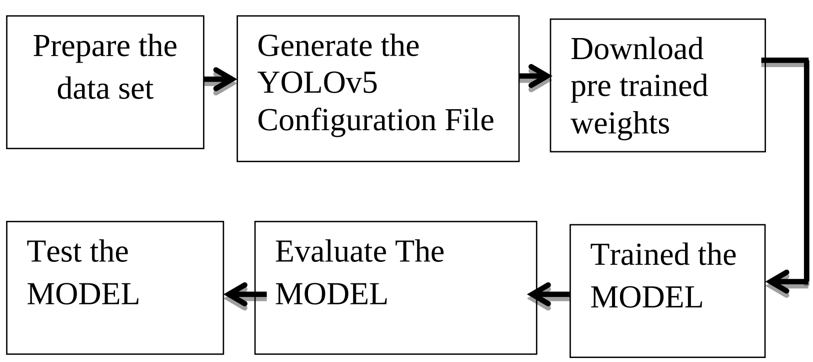
1. The Leaf structure: To identify a leaf, first we have to know the structure of the leaf. Each plant has its own leaf structure which varies from leaf structure of plants from other species.
2. Identify Plants: If we can able to identify leaf, we can easily able to identify plants. Each species has a different type of leaf which have different features like length, aspect ratio, vein density, texture,etc.
3. Plant leaves classification: If we can able to identify leaf, we can easily able to classify the plants.We make use of datasets for classifying and identifying the plants in natural condition.

Research Paper
Impact Factor: 7.352
Peer Reviewed Monthly Journal
www.ijmdrr.com

IJMDRR
E- ISSN –2395-1885
ISSN -2395-1877

Data Collection

Figure 1 Images of Different Species



Flow chart for Training a YOLO v5 Custom model

IJMDRR
E- ISSN –2395-1885
ISSN -2395-1877

Model Design

YOLOV5

Modern real-time object identification technology is used by YOLOv5 (You Only Look Once, Version 4). It was created by a research team at the University of Washington under the direction of Alexey Bochkovskiy and is an upgrade over its forerunner, YOLOv3. With single-pass image processing and real-time prediction of bounding boxes and class probabilities for several objects, YOLOv5 uses DNN architecture. To do this, the image is divided into a grid of cells, and each cell's likelihood of containing an item, its bounding box coordinates, and class probabilities are predicted. To increase the precision and speed of object recognition, YOLOv5 employs some cutting-edge approaches, including weighted residual connections, mish activation functions, and spatial pyramid pooling. On some object detection benchmarks, including COCO and KITTI, YOLOv5 has achieved cutting-edge performance. It is widely employed in many applications, such as robots, self driving cars, and surveillance. Since the YOLOv5 code is open source and accessible on Git Hub, developers can use and alter it for their projects.

YOLOv5custom training The critical steps in developing a bespoke YOLOv5 model include the preparation of the dataset, the determination of training parameters, and the implementation of model training. A concise overview of these methodologies is provided below:

Prepare the dataset creating a dataset with labeled photos is the initial stage. Bounding boxes, class labels, and images of the items that one wants to detect must all be included in the dataset. Applications such as VoTT, LabelImg, and YOLOv5 Label can be utilized to annotate the data.

Generate the YOLOv5 configuration file constructing a YOLOv5 configuration file that specifies the model architecture, hyper parameters, and training settings is the following step. The default configuration file from the YOLOv5 repository may be modified to meet specific requirements.

Download pre-trained weights one potential method for streamlining the training process is to utilize pre-trained weights designed for the YOLOv5 architecture. The Darknet framework, which YOLOv5 employs, possesses pretrained weights.

Train the model after obtaining the configuration file and the dataset, the model can start to be trained. The YOLOv5 model can be introduced on a special dataset using the Darknet framework. Throughout training, the model will improve its recognition of the objects in the dataset.

Evaluate the model after training on a validation dataset; the model's performance is evaluated. Metrics like accuracy, precision, recall, and F1-score is used to assess the model's performance.

Test the model: It can then be tested on a test dataset to evaluate its performance with brand-new, untested data. Moreover, our research utilized a data splitting scheme of 90% for training and 10% for testing. Specifically, 90% of the training set was designated for training purposes, while the remaining 10% was employed for validation. These proportions guarantee rigorous training and evaluation of the model. Moreover, due to the heavy computational tasks involved, training a GPU model takes a lot of resilience. During training, the GPU does a huge number of complicated calculations over and over again, going through huge datasets to find the best settings for the model. For each epoch, the GPU has to constantly process and update millions of factors, making changes to them to make the model more accurate. The need for endurance comes from the fact that training sessions can last for hours, days, or even weeks, based on the size and complexity of the dataset. Keeping a steady level of steadiness and computational performance over long periods of time is therefore necessary to make sure that a high-quality YOLOv5 model is trained well.

YOLOv5 for plant leaf Identification

Image Annotation: The first step is to annotate images by the disease of the particular leaf. Annotated images are as follows in Figure 1: Image labeling is a laborious and time-consuming undertaking that demands undivided attention in order to individually label each image. This procedure is accelerated through the use of a Python iteration, which facilitates the management of numerous images simultaneously. A thorough examination of the images enabled the accurate determination of bounding box ratios. Despite sincere endeavors, it is widely recognized that accurate annotations are crucial for preserving the integrity and dependability of the dataset in preparation for subsequent scientific investigations and research. Ensuring accurate annotation of images are crucial for preserving the practicality and integrity of academic projects.

Images labels: It is critical to bear in mind that labels must be created for the images. The name of the folder should specify the type of malady or food contained within.

Object data preparation: The next step is to store the number of classes, train and test text file path, and backup folder training in the text file, which will be used to generate image paths and train those images in YOLOv5 custom training.

1. Classes = 20
2. Train = Data/Train.Txt
3. Valid = Data/Test.Txt
4. •Names=Data/Obj.Names
5. Backup =/Mydrive/Yolov5/Training

Generate the YOLOv5 Configuration File: To generate the YOLOv5 custom file, obtain the original YOLO-custom configuration file from the Darknet repository. Personalize this file by altering specific attributes. Modify the number of classes in all YOLO layers to 20, and fine-tune the number of filters in the final convolution layer preceding the YOLO layer to 75. These modifications customize the arrangement to the unique demands of the desired object detection assignment. After making the necessary modifications to the class count and filter configuration, the customized YOLOv5 file is now prepared for implementation. These changes have been made to enhance the performance of the file in the intended application.

Darknet

Darknet: The Darknet neural network framework was created by Joseph Redmon. It is based on C/CUDA and is utilized in computer vision applications for object identification and image classification. Because of its speed, efficiency, and versatility, the darknet has gained popularity in both business and academic circles. Due to its open-source nature and ease of use, a wide range of users can utilize the framework. Darknet is a very useful tool with a wide range of applications since it can identify items and categorize photographs. The structure of the software is both practical and versatile, which greatly contributes to its wide usage. Darknet is a very powerful computer vision tool that performs exceptionally well on challenges using neural networks. Because of this, it is a very useful tool for practitioners and scholars in the field.

Customizable Network Architecture: Darknet allows users to define and train their neural network architectures, which can be optimized for specific computer vision tasks. This flexibility is beneficial when pre-trained models do not perform well or when a new application requires a unique architecture.

Support for Multiple Platforms Darknet can be run on various platforms, including Linux, Windows, and MacOS. Additionally, it can be compiled to run on GPUs, which significantly speeds up training and inference times.

Integration with Open CV Darknet is designed to work seamlessly with OpenCV. This popular computer vision library provides a range of image processing and analysis functions. This integration allows users to easily preprocess images and extract features for their neural network models.

Pre-trained models Darknet provides pre-trained models for various computer vision tasks, including object detection and image classification. These models can be used as a starting point for new applications or fine-tuned for specific use cases. Overall, Darknet is a robust computer vision framework that offers flexibility and speed. Its popularity in the research community and industry is a testament to its effectiveness, and it continues to be widely used and developed today.

Cloning To clone Darknet, Darknet for custom YOLO v4 training, follow these steps:

Install Git If Git is not installed on the system; install it from the official Git website.

Clone Darknet Open a terminal window and navigate to the directory where DarknetDarknet is to be cloned.

Yolo V4 Custom Weights Download Pre-Trained YOLO v4 Weights: To use YOLO v4, the pre-trained weights need to be downloaded.

Yolo V4 Training

Customize Configuration Files In the Darknet directory, navigate to the CFG folder. Here, several configuration files for different YOLO versions can be found. For YOLO v4, the “YOLOv5.cfg” file should be used as a starting point and customized to suit the needs.

Prepare Training Data To train a custom YOLO v4 model, the training data must be preparedin the YOLO annotation format. This involves creating text files for each image that contain the object annotations and their corresponding class labels.

Conclusion

The study focuses on addressing the complex issue of open-set detection of plant leaf by incorporating an image retrieval method into the YOLOv5 framework. The method uses brief annotated photos to achieve both the identification of plants at the same time, demonstrating significant progress in accuracy, flexibility, and reliability. Improving YOLOv5 to better detect small details significantly enhances the precision of diagnosing leaf and plants. The investigation’s effectiveness depends significantly on the smooth incorporation of the Plant Village dataset and the YOLOv5 architecture to accurately identify and classify different types of plant. The technique demonstrates outstanding proficiency in accurately identifying plants via picture labeling, data preparation, and intensive model training. This study significantly enhances the utilization of deep learning methods for detectingin this specific area. Future study seeks to broaden the approaches by include a wider range of plants and datasets. Enhancing the system’s performance will require investigating alternative deep-learning methods. Continual improvements and fine-tuning of procedures are being made to promote the role of technology in protecting global food production and supporting agricultural sustainability in the future.

References

1. Kaur, PriyaPinder, Sukhdev Singh, Inderjeet Singh Sandhu, and Damandeep Singh Sandhanwalia. "Leaf Recognition System Based on Supervised Machine Learning Techniques." In 2022 IEEE Delhi Section Conference (DELCON), pp. 1-6. IEEE, 2022.
2. Yang, C., "Plant leaf recognition by integrating shape and texture features", Pattern recognition, vol.112, p. 107809, 2021.
3. Liu, Xuan, Chunhua Hu, and Pingping Li. "Automatic segmentation of overlapped poplar seedling leaves combining mask R-CNN and DBSCAN." Computers and Electronics in Agriculture 178 (2020): 105753.
4. Akter, R., and Hosen, M., "CNN-based leaf image classification for Bangladeshi medicinal plant recognition", Emerging technology in computing, communication and electronics, pp. 1-6, doi: 10.1109/ETCCE51779.2020.9350900.
5. Turkoglu, M., and Hanbay, D., “Leaf-based plant species recognition based on improved local binary pattern and extreme learning machine,” Physical A: statistical mechanics and its applications, vol. 527, p. 121297, 2019.
6. Choudhury, S., Yu, J., and Samal, A., "Leaf recognition using contour unwrapping and apex alignment with tuned random subspace method", Biosystems engineering, vol. 170, pp. 72-84, 2018.
7. Araujo, V., Britto, A., Brun, A., Koerichm, A., and Palate, R., “Multiple classifier system for plant leaf recognition,” 2017 IEEE international conference on systems, man and cybernetics, pp. 1880–1885
8. Munisami, T., Ramsurn, M., Kishnah, S., and Pudaruth, S., “Plant leaf recognition using shape features and colour histogram with k-nearest neighbour classifiers,” Procedia computer science, vol. 58, pp. 740 – 747, 2015.
9. Kheirkhah F., and Asghari, H., “Plant leaf classification using GIST texture features,” IET computer vision, vol. 13, p. 369, 2018.
10. Naresh, Y. G., and Nagendraswamy, H. S., "Classification of medicinal plants: an approach using modified LBP with symbolic representation." Neurocomputing 173 (2016): 1789-1797.
11. Guillermo, G., Uzal, L., Larese, M., and Granitto, P., “Deep learning for plant identification using vein morphological patterns,” Computers and electronics in agriculture, vol. 127, pp. 418–424, 2016.
12. Kumar, P., Rao, K., Raju, A., and Kumar, D., “Leaf classification based on shape and edge feature with k-NN classifier,” 2016 2nd Int. Conf. Contemp. Comput. Informatics, pp. 548–552, 2016.
13. Salve, Pradip, MilindSardesai, Ramesh Manza, and PravinYannawar. "Identification of the plants based on leaf shape descriptors." In Proceedings of the Second International Conference on Computer and Communication Technologies, pp. 85-101. Springer, New Delhi, 2016.
14. Munisami, T., Ramsurn, M., Kishnah S., and Pudaruth, S., “Plant leaf recognition using shape features and colour histogram with k-nearest neighbour classifiers,” Procedia computer science, vol. 58, pp. 740 – 747, 2015.
15. Chaki, Jyotismita, Ranjan Parekh, and Samar Bhattacharya. "Plant leaf recognition using texture and shape features with neural classifiers."Pattern Recognition Letters 58 (2015): 61-68.