MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 35 Issue 2, 2025, Pages: 557-573

HYBRID LION OPTIMIZATION WITH FASTER MRCNN TO CLASSIFY PLANT LEAF DISEASE

Nandha Kumar G, Dr. V. Vijayakumar
Research Scholar, Department of Computer Science, Sri Ramakrishna College of Arts and Science, Coimbatore, India
Professor, Computer Science and Controller of Examinations, Sri Ramakrishna College of Arts and Science, Coimbatore, India

ABSTRACT

Plant leaf disease creates disaster on agricultural production and crop output. Reliable and prompt disease categorization is critical for efficient disease control and protection of crops. In this research, proposed a hybrid approach that combines Hybrid Lion Optimization (LO) with Faster Mask R-CNN to classify plant leaf diseases. Lion Optimization leverages the hunting and social behaviours of lions in nature, combined with genetic algorithms, to enhance the limitations of the Faster Mask R-CNN model. By applying LO, attempt to discover the best collection of hyper parameters or architecture configurations that make the most of the performance of the leaf disease classification system. The Faster Mask R-CNN, an advanced object detection and instance segmentation algorithm, serves as the core framework for disease classification. It extracts data using a deep CNN landscapes from plant leaf images and performs region-based detection to identify and classify diseases within those regions. The consequences of our experiments determine that the hybrid approach improves the accuracy and efficiency of classification. The hybrid optimization algorithm effectively classify plant leaf disease with accuracy 97.4%. This hybrid approach has significant implications for the field of agriculture, offering a reliable and automated solution to aid in the diagnosis and categorization of plant leaf diseases.

Keywords: Plant Leaf Disease, Classification, Lion Optimization, Faster mask RCNN, Soy Leaf, Deep Learning.

1 INTRODUCTION

Plant diseases have a detrimental impact on agricultural productivity, leading to substantial crop losses and threatening food security worldwide. The promptly and correct detection of plant leaf diseases is critical for disease control management, enabling targeted interventions to prevent the spread and minimize the economic impact of these diseases.

Traditional methods of disease identification involve physical examination by skilled professionals, which may take a long time, labor-intensive, and susceptible to human mistake. With advancements computerised strategies for plant disease identification in artificial intelligence and machine learning have emerged as a promising solution.

Plant leaf diseases are classified using the analysis of visual symptoms exhibited by the leaves. These symptoms can include discoloration, spots, lesions, deformities, and patterns that are indicative of specific diseases. By analyzing and classifying these symptoms, it becomes possible to identify the underlying diseases affecting the plants.

Computer vision-based approaches leverage image processing techniques and machine learning algorithms to extract significant elements from plant leaf images and categorise them into several disease groups automatically. These techniques enable the development of accurate and efficient systems for disease identification, assisting farmers and agricultural specialists may make educated judgements about disease control techniques with the assistance of disease management professionals.

Deep learning algorithms, notably CNNs, have demonstrated outstanding effects in a variability of computer vision applications, such as plant disease grouping, over the past few years. CNNs can learn hierarchical representations from raw image data, capturing intricate patterns and discriminative features that are essential for accurate disease classification.

Object detection and segmenting instances are two Deep Learning model applications are fundamental tasks that involve identifying objects within images and accurately delineating their boundaries. Mask R-CNN is an innovative model of a deep learning model that excels at solving these tasks with remarkable precision and efficiency.

Traditional object detection methods often rely on two-stage approaches, where regions of interest (RoIs) are first identified and then classified. Mask R-CNN builds upon this foundation by extending it to include pixel-level segmentation, enabling not only the detection and classification of objects but also the precise delineation of their boundaries.

The success of Mask R-CNN lies in its ability to simultaneously address object detection and instance segmentation, surpassing the limitations of earlier models. By integrating these tasks into a single architecture, Mask R-CNN not only localizes and classifies objects but also provides precise object masks, enabling a more comprehensive understanding of the visual scene.

The applications of Mask R-CNN are diverse and far-reaching. It has been shown to be quite successful in various domains, including autonomous driving, robotics, medical imaging, and natural scene understanding. Its ability to accurately segment objects and extract detailed information from images makes it a valuable tool for a variety of computer vision-related activities.

Furthermore, the flexibility and adaptability of Mask R-CNN have led to advancements and extensions of the original model. Researchers have explored variations of Mask R-CNN, such as feature pyramid networks (FPN), to improve performance and handle objects at different scales effectively. These developments have further enhanced the model's capabilities and expanded its potential applications.

Lion optimization (LO) algorithm is inspired by the social behavior of lions, and it uses a population-based approach to search for optimal solutions to optimization problems. The basic idea of LO algorithm is to simulate the behavior of a pride of lions as they search for prey. The lions in the pride are divided into two groups: territorial lions and nomadic lions. The territorial lions defend their territory, which is a region of the solution space that they believe contains a good solution to the optimization problem. The nomadic lions roam the solution space, looking for new and better solutions.

The LO algorithm works by iteratively updating the positions of the lions in the pride. The territorial lions update their positions by moving towards the best solution that they have found so far. The nomadic lions update their positions by randomly moving around the solution space.

2 RELATED WORKS

Crop and diseases of plants are gaining more emphasis than they previously ever did as a direct effect of changes in the climate and warming temperatures. This is due to the fact they are moving further and more swiftly than they ever have before. The tomato leaf miner is responsible for the destruction of the development framework of the tomato, which results in a loss of anywhere from 85-100% of the crop. Even though there have been significant attempts to stop its migration, the tomato leaf miner has made its way to most continents.

In order to tackle the challenging issue of complicated abstraction for tree image in the already complex backdrop, we chose tree species as the study topic and offered a rapid system solution for tree image based on the Caffe platform. This was done with the goal of solving the tough challenge.

In this research, we construct a novel model for the categorization of plant leaves based on improved segmentation and careful consideration of available features. The novel hybrid algorithm, which is a hybridization of Electric Fish Optimisation and Crowd Search Algorithm, is used to achieve the best feature selection in this scenario.

This innovative method guarantees the discovery of the Anredera cordifolia since the distinguishing characteristics have been utilised as class objects in the Mask-RCNN algorithm. The findings of the studies demonstrated that the Mask-RCNN was capable of superimposing masks and bounding boxes on top of the Anredera cordifolia characteristics that were identified.

The problem of how to boost crop yields while simultaneously maintaining the sustainable growth of ecologically friendly agriculture is a problem that affects countries all over the globe. The use of autonomous systems, technological advancements in sensing, and artificial intelligence all provide promising avenues for addressing this concern.

The purpose of this study is to build a fine recognition system for plant factory seedling phenotypes that is based on an ANN and an ant colony algorithm. In order to solve the challenges caused by the plant identification system's high latency and complicated nature.

For the sake of using machine learning strategies in IoT-based smart agricultural systems, there are still some outstanding difficulties that need to be solved. This study gives a complete assessment of vision-based machine learning approaches for the identification of plant diseases, beginning with data capture methods and moving on to the accessibility of public datasets.

This information, which may include colour, appearance, shape, and others, may be extracted with the use of the leaf image variable. Colour and surface roughness will play a role in the formation of features in the proposed system. In order to extract information about the colours, the colour pattern for the texture makes use of the GLCM and Shape extraction forms.

It is possible to lessen the severity of the effects of this illness by breeding wheat cultivars that include genes for resistance. In order to do this, well-trained specialists are required to evaluate the disease resistance of hundreds of wheat lines in the field. Manual assessment procedures need a significant amount of time and effort to complete. The outcomes of the examination are significantly impacted by the presence of human variables.

The deep learning models achieve a significant deal of success; yet, because to their inability to be interpreted, their potential applications are restricted. They were able to circumvent these restrictions by extracting quantifiable, interpretable, and computer-aided information from the images of plant leaves.

In this work, a new segmentation and classification technique based on leaf segmentation fuzzy CNN is suggested for identifying tomato leaf illnesses with complicated background interference. The system was developed in order to detect tomato leaf diseases.

Using an unmanned aerial vehicle within greenhouses, this research proposes a unique technique to real-time plant illness diagnosis and autonomous pesticide spraying. Both of these processes take place simultaneously.

The earlier strategies that were used in this setting were only predicated on classification and detection. These methods have certain drawbacks, and they did not determine the precise quantity of the infection that was present. A segmentation strategy may be used to get around the constraints imposed by earlier approaches by properly segmenting all infected areas of the disease on the leaves in the exact form of those spots. This would allow the illness to be treated more effectively.

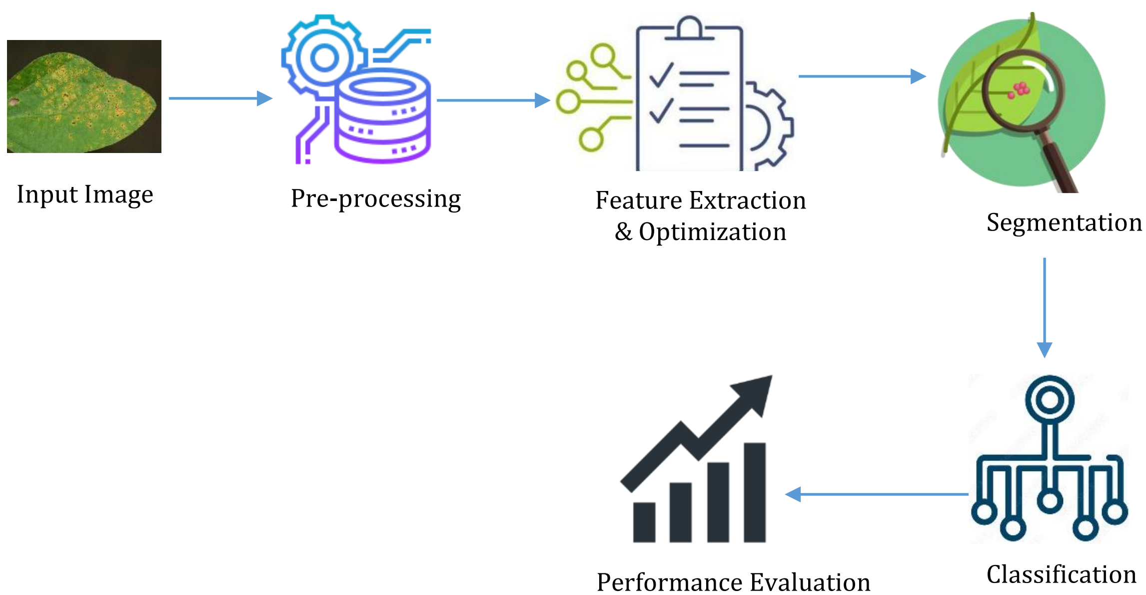
MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 35 Issue 2, 2025, Pages: 557-573

Several different methods of prognostication pertaining to liver disorders have been established. However, they are more difficult and costly to produce. As a result, the purpose of this effort is to develop an efficient approach for diagnosing liver illnesses in their early stages.

As a result of the ongoing effect of global food and environmental problems, there is a growing need for agriculture that makes use of advanced techniques. With the fast development of technology for object detection, it is now feasible to achieve excellent effectiveness and high precision in fruit detection systems. This will be discussed in the context of fruit detection.

3 PROPOSED MODEL

Leaf disease classification is an important task in agriculture for detecting and managing plant health issues. In this paper, Soybean leaf dataset has taken for experimental analysis. To enhance the Soybean leaf disease classification process, proposed a combination of Lion Optimization and Faster Mask R-CNN.



Soybean plant leaf disease classification is a common task in agriculture to identify and manage diseases that affect soybean crops. To explore the use of lightweight backbone networks, such as EfficientNet to replace heavier networks like ResNet in Mask R-CNN. These efficient backbones maintain good performance while reducing the computational complexity and memory requirements of the algorithm.

3.1 Lion Optimization Algorithm

The Lion Optimization (LO) Algorithm is a kind of method for population-based optimisation that simulates the hunting behavior of lion prides. The LO algorithm works by first creating an initial population of solutions, called lions. These lions are then randomly assigned to prides, which are groups of lions that work together to find optimal solutions. Each pride has a territorial lion, which is the strongest lion in the pride and is responsible for leading the pride to the optimal solution. The other lions in the pride are called lioness, cubs, and nomadic lions.

The LOA algorithm then proceeds through a series of steps, where the lions in each pride are updated according to the following rules:

- Territorial lions are updated by moving towards the best solution in the pride.

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 35 Issue 2, 2025, Pages: 557-573

- Lionesses are updated by moving towards the territorial lion.
- Cubs are updated by moving towards a random solution in the pride.
- Nomadic lions are updated by moving randomly in the search space.

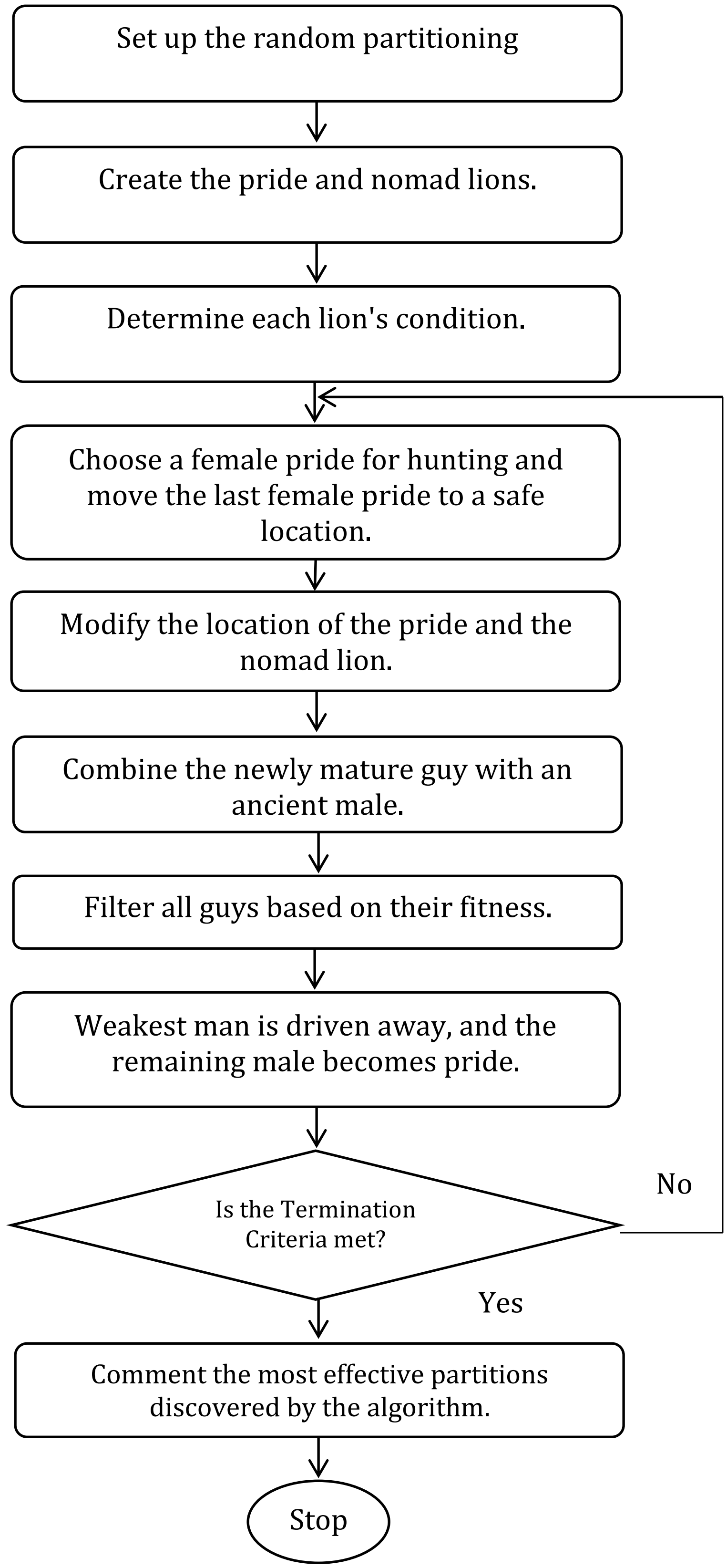
The algorithm dynamically adjusts the hunting behaviors and strategies of different lion groups based on their performance and interaction. By facilitating information exchange through migration and group formation, LOA encourages knowledge sharing among lions, promoting diversity and convergence towards better solutions.

```
def LOA(f, x0, max_iter, alpha, beta):
    lions = np.random.rand(max_iter, d) # Initialize the population of lions
    best_lion = lions[0] # Initialize the best solution
  for i in range(max_iter):
    territorial_lions = lions[lions[:, -1] == 1]  # Update the territorial lions
    territorial_lions = territorial_lions - alpha * (territorial_lions - best_lion)
    lionesses = lions[lions[:, -1] == 2] # Update the lionesses
    lionesses = lionesses - beta * (lionesses - territorial_lions)
    cubs = lions[lions[:, -1] == 3]
    cubs = cubs - beta * (cubs - np.random.choice(territorial_lions))
    nomadic_lions = lions[lions[:, -1] == 4]
    nomadic_lions = nomadic_lions - beta * np.random.rand(max_iter, d)
    best_lion = lions[np.argmin(f(lions))] # Update the best solution
  return best_lion
```

where,

- f: The objective function that is being optimized.
- x0: The initial population of lions.
- max_iter: The maximum number of iterations.
- alpha: The learning rate for the territorial lions.
- beta: The learning rate for the other lions.

The LOA algorithm works by first initializing a population of lions. These lions are then updated according to the rules described above. The algorithm continues to iterate until the stopping criterion is met.

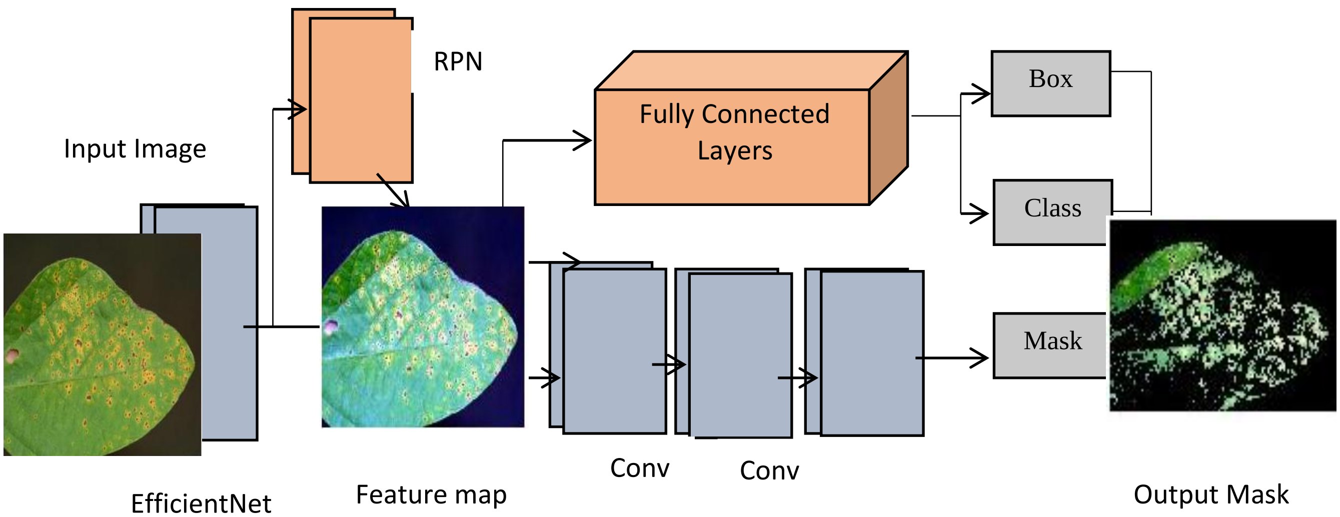

Set up the random partitioning
Create the pride and nomad lions.
Determine each lion's condition.
Choose a female pride for hunting and move the last female pride to a safe location.
Modify the location of the pride and the nomad lion.
Combine the newly mature guy with an ancient male.
Filter all guys based on their fitness.
Weakest man is driven away, and the remaining male becomes pride.
Is the Termination Criteria met?
No
Yes
Comment the most effective partitions discovered by the algorithm.
Stop

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 35 Issue 2, 2025, Pages: 557-573

The LO algorithm has been shown to be effective for a variety of optimization problems. It is a simple and efficient algorithm to implement, and it has been shown to be effective for a variety of problems.

3.2 Faster Mask-RCNN

Faster Mask R-CNN is an advanced computer vision algorithm that combines the capabilities of the identification of objects and the partitioning of instances. Faster Mask R-CNN enhances the original Mask R-CNN by incorporating a more efficient region proposal mechanism, resulting in improved speed and accuracy. The primary objective of Faster Mask R-CNN consists of properly detecting things inside an image and accurately describing those objects and segment their instances.



Faster Mask R-CNN consists of several key components. Firstly, it employs a backbone network, such as EfficientNet, to extract high-level feature maps from the input image. These feature maps capture meaningful information about the objects present in the image. An EfficientNet backbone architectures have been widely adopted in combination with algorithms like Mask R-CNN, Faster R-CNN, and other object detection and instance segmentation frameworks. Their suitability depends on the specific requirements of the task at hand, including computational resources, accuracy demands, and the size and complexity of the input data.

Next, a RPN is used to crop potential areas of interest (bounding box proposals) where objects might be located. The RPN efficiently identifies object candidates by predicting their objectness scores and regressing their bounding box coordinates.

After that, a strategy for pooling regions of interest is used by the algorithm in order to obtain fixed-sized feature maps for each suggested area. These RoI feature maps are fed into a shared head network, which consists of fully connected layers. The head network performs two tasks: object classification and bounding box regression. It classifies the proposed regions into specific object categories and refines the bounding box coordinates for precise localization.

During the training process, Faster Mask R-CNN is optimized by minimizing various loss functions associated with object classification, bounding box regression, and mask prediction. The algorithm is trained on large-scale annotated datasets to learn discriminative features and achieve high accuracy in activities involving object identification and instance segmentation.

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 35 Issue 2, 2025, Pages: 557-573

Faster Mask R-CNN has validated the presentation in object detection and instance segmentation challenges, offering a powerful solution for visual understanding tasks. Its speed and accuracy improvements make it a valuable tool in many real-world applications that require precise identification and segmentation of objects within images. The Mask R-CNN algorithm involves several steps, including region proposal, region of interest (RoI) pooling, classification, bounding box regression, and mask prediction.

Backbone Network:

The mathematical model for EfficientNet is based on the compound scaling method. CNN may have its depth, breadth, and resolution increased in a systematic manner via the use of the compound scaling approach. The compound scaling technique was developed with the intention of improving the precision of the CNN without increasing its computational cost too much.

```
def compound_scaling(depth, width, resolution, phi):
    scaled_depth = depth ** phi    # Calculate the scaled depth
    scaled_width = width ** phi    # Calculate the scaled width
    scaled_resolution = resolution ** phi    # Calculate the scaled resolution
    return scaled_depth, scaled_width, scaled_resolution
```

Where, *depth:* The original depth of the CNN, *width:* The original width of the CNN, *resolution:* The original resolution of the CNN, phi: The scaling factor.

The EfficientNet architecture is scaled using the compound scaling method in order to obtain a high level of precision while also keeping a reasonable computational cost. The EfficientNet architecture has been shown to be very effective for image classification.

Region Proposal Network (RPN):

It is a totally convolutional network that creates a series of region recommendations for elements in an image. The RPN is trained to estimate the likelihood that each area proposal includes an item, as well as the bounding box coordinates.

It is made up of two layers of convolution and two completely linked layers. The first convolutional layer takes an image as input and creates a feature map as output. The second convolutional layer applies a 3x3 convolution to the feature map and outputs a feature map with double the amount of channels. The two completely interconnected layers then predict the probability that each region proposal contains an object, as well as the coordinates of the bounding box.

```
def rpn(image):
    # 1. Apply two convolutional layers to the image
    feature_map = conv2d(image, 16)
    feature_map = conv2d(feature_map, 32)
    # 2. Predict the probability that each region proposal contains an object
    scores = fc(feature_map, 1)
    # 3. Predict the coordinates of the bounding boxes
    bounding_boxes = fc(feature_map, 4)
    return scores, bounding_boxes
```

A mix of bounding box classification and regression with loss is used to train the RPN. The bounding box regression loss is used to minimise the distance among the predicted and true bounding boxes. The classification loss is used to minimise the cross-entropy between predicted and true probability.

RoI (Region of Interest) Pooling:

The RoI Pooling procedure is utilised in the Faster R-CNN and Mask R-CNN designs to extract fixed-sized feature maps for each region proposal. It allows for variable-sized region proposals to be transformed into a fixed spatial dimension, enabling subsequent processing and classification. The RoI Pooling operation can be mathematically described using the following steps:

1. Given an input feature map of size H x W x C (height x width x channels) and a region proposal with coordinates, respectively..
2. Divide the region proposal into a fixed number of sub-regions.
3. Compute the width and height of each sub-region as follows:
   $sub\_region\_width = w / P$
   $sub\_region\_height = h / P$
4. Apply max pooling within each sub-region separately:

- For each sub-region, divide it into a fixed grid of S x S bins.
- Pool the maximum value within each bin.
- The resulting pooled value represents the highest activation value within that bin for the corresponding sub-region.

5. Collect the pooled values from all sub-regions and concatenate them into a fixed-sized feature map of size S x S x C.

Mathematically, the RoI Pooling operation can be represented as follows:

Let (i, j) denote the bin coordinates within a sub-region, and (p, q) denote the sub-region coordinates within the region proposal.

The coordinates (x', y') within the input feature map corresponding to a specific bin in a sub-region can be computed as:

$$x' = x + (j + q * S) * sub\_region\_width$$
$$y' = y + (i + p * S) * sub\_region\_height$$

The RoI Pooling operation effectively transforms region proposals of various sizes into fixed-sized feature maps, which can then be further processed by subsequent layers, such as fully connected layers or convolutional layers, for object classification or other tasks.

Bounding Box Regression:

Bounding Box Regression are two tasks performed after the Region Proposal Network (RPN) to classify the proposed regions and refine their bounding box coordinates. These tasks involve mathematical equations for predicting class probabilities and adjusting the bounding box coordinates based on the region proposals. Here are the mathematical equations for each task: Along with classifying the proposed regions, the Faster Mask R-CNN architecture aims to refine the bounding box coordinates to more accurately localize the objects within the regions.

Each proposed region is associated with a bounding box. The bounding box regression task involves predicting adjustments (Δx, Δy, Δw, Δh) to the coordinates of the initial bounding box. These adjustments are applied to the initial coordinates to obtain the refined bounding box coordinates. The predicted adjustments are typically learned using a fully connected layer or a set of convolutional layers.

The refined bounding box coordinates are computed as:

$$x' = x + \Delta x * w \, y' = y + \Delta y * h \, w' = w * exp(\Delta w) \, h' = h * exp(\Delta h)$$

The classifier's job in training the Faster Mask R-CNN model employs the cross-entropy loss for contrasting the predicted class possibility with the actual truth labels. A smooth L1 loss is used in the bounding box

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 35 Issue 2, 2025, Pages: 557-573

regression job to quantify the difference between the anticipated modifications and the ground truth bounding box coordinates.

Mask Prediction

Mask Prediction is a task performed in the Mask R-CNN architecture to generate pixel-level segmentation masks for the proposed regions. The goal is to predict a binary mask for each proposed region, where each pixel within the mask indicates whether it belongs to the object or the background. Here are the mathematical equations for the Mask Prediction task:

Mask Head:

The Mask Head in Mask R-CNN is typically implemented as a CNN applied to each proposed region independently. Convolutional layers are accompanied by a fully connected layer and another convolutional layer with a sigmoid activation in the CNN. Let x be the extracted feature map from a suggested area. The output of the Mask Head is denoted as mask_head(x), which represents the predicted mask for the region.

Pixel-wise Classification:

The output of the Mask Head, mask_head(x), represents a probability map for each pixel within the proposed region. The probability map indicates the likelihood of each pixel belonging to the object. For each pixel (i, j) within the proposed region, the predicted probability of it being part of the object is given by: $P(object \mid x, i, j) \ = \ mask\_head(x)[i, j]$

Thresholding:

To convert the probability map into a binary mask, a thresholding step is applied. A pixel is considered part of the object if its probability exceeds a predefined threshold value. The thresholded binary mask, M, is obtained as: M(i, j) = 1, if P(object | x, i, j) > threshold M(i, j) = 0, otherwise The Mask Prediction task is trained using binary cross-entropy loss, which compares the predicted pixel-wise probabilities with the ground truth masks. The loss is computed for each pixel within the proposed regions, encouraging the network to accurately segment the objects.

Loss Function

Faster Mask R-CNN algorithm uses multiple loss functions to teach the network object detection, categorization, bounding box regression, however, and mask prediction.

Classification Loss:

For each area suggestion, the classification loss quantifies the variance among the predicted class possibilities and actual truth class labels. The most commonly used loss function for classification is the softmax cross-entropy loss. The classification loss equation is expressed as follows:

$$L\_cls \ = \ -\sum[y \ * \ log(p) \ + \ (1 \ - \ y) \ * \ log(1 \ - \ p)]$$

where y is the ground truth class label (binary indicator) and p is the predicted class probability.

Bounding Box Regression Loss:

The bounding box regression loss computes the variation among the predicted and ground truth bounding box dimensions. The smooth L1 loss function is the most often used loss function for bounding box regression.

The equation for the bounding box regression loss can be represented as:

$$L\_reg \ = \ \sum[smooth\, L1(\Delta x)]$$

where Δx indicates the difference between expected and actual bounding box parameters.

Mask Prediction Loss:

For each area suggestion, the mask prediction loss computes the variation among the predicted masks and the ground truth masks. The binary cross-entropy loss is the most widely used loss function for mask prediction. The equation for the mask prediction loss can be represented as:

$$L\_mask = -\sum[y * log(p) + (1 - y) * log(1 - p)]$$

where y is the ground truth mask (binary indicator) and p is the predicted mask probability.

Total Loss:

The classification loss, bounding box regression loss, and mask prediction loss are combined to form the total loss. The total loss equation is expressed as follows:

$$L\_total = L\_cls + \lambda 1 * L\_reg + \lambda 2 * L\_mask$$

where λ1 and λ2 are hyperparameters that control the relative importance of When analysed alongside the classification loss, the bounding box regression loss and mask prediction loss performed better. During the training phase, the network adjusts its parameters via backpropagation and gradient descent optimisation to minimise total loss.

Steps for Proposed Model:

1. Initialize Lion population with random positions and velocities.
2. Evaluate the fitness of each Lion based on their positions in the search space:
   - Implement Faster Mask R-CNN to classify plant leaf disease and measure accuracy.
3. Set the global best position and fitness value based on the Lion with the highest fitness.
4. Repeat until the termination condition is met:
   a. Update the positions and velocities of Lions using Lion Optimization:
      - Update the velocity of each Lion based on its previous velocity and the Lion's best position.
      - Update the position of each Lion based on its previous position and the updated velocity.

   b. Bound the positions and velocities within the defined search space to ensure feasibility.
   c. Evaluate the fitness of each Lion based on the updated positions:
      - Implement Faster Mask R-CNN to classify plant leaf disease and measure accuracy.

   d. Update the global best position and fitness value if a Lion with a higher fitness is found.
5. Return the best position found, which represents the optimized parameters for Faster Mask R-CNN.
6. Implement Faster Mask R-CNN using the best parameters obtained from Lion Optimization to classify plant leaf disease.

4 RESULTS AND DISCUSSIONS

4.1 Dataset Description

In this research, dataset were collected from kaggle.com for the soybean leaf disease. Dataset is a collection of images specifically focused on soybean leaf diseases. It is a widely used dataset in the area of plant pathology for study and progress of algorithms related to soybean disease detection and classification. The dataset is composed of three image folders: Caterpillar, Diabrotica Speciosa, and healthy. Images are of soybean leaves damaged by Caterpillar, Diabrotica Speciosa healthy images. These images were standardized by dimension 500 x 500. There are 6,410 images, being: caterpillar - 3,309, diabrotica speciosa - 2,205 and healthy - 896.

Soybean plant leaf diseases are a significant concern in agriculture as they can affect crop yield and quality. Table 1 gives some common soybean plant leaf diseases with its symptoms and impact.

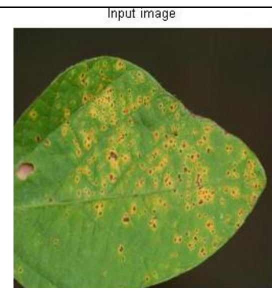
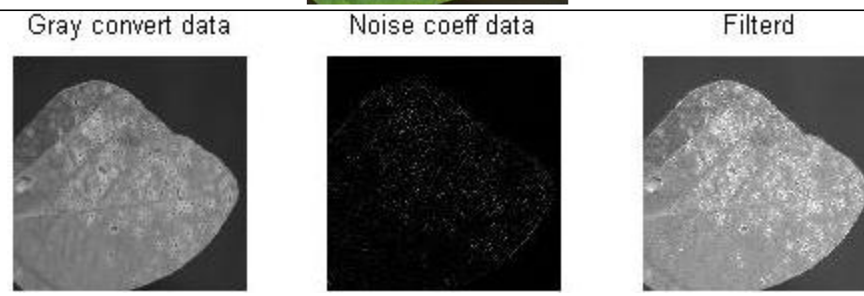
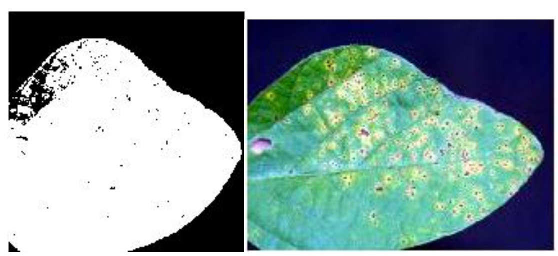
MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 35 Issue 2, 2025, Pages: 557-573

Table 1: soybean plant leaf diseases

soybean plant leaf diseases	Symptoms	Impact
Soybean Rust	Small, reddish-brown pustules on the underside of leaves, leading to yellowing and defoliation.	Significant yield loss if not managed properly.
Septoria Brown Spot	Circular to oval brown spots with yellow halos on the leaves, which may coalesce and cause premature defoliation.	Can reduce plant photosynthesis and yield.
Cercospora Leaf Blight	Purple to brown irregularly shaped lesions with gray centers on the leaves, often surrounded by a yellow halo.	Can cause defoliation and yield reduction.
Frogeye Leaf Spot	Circular to elliptical lesions with gray centers and reddish-purple margins on the leaves.	If significant, it may cause defoliation with yield loss.
Brown Stem Rot	Brown darkening of the pith with vascular cell membranes of the stem, yellowing and necrosis of lower leaves.	Can cause lodging and yield reduction.
Phytophthora Root and Stem Rot	Wilting, yellowing, and eventual death of plants.	Severe yield loss if not managed effectively.
Bean Pod Mottle Virus	Mottled discoloration on the pods, distortion of leaves, reduced seed quality.	Can cause yield reduction and affect seed viability.

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 35 Issue 2, 2025, Pages: 557-573

4.2 Experimental Analysis

Image Description	Output
Input Image	
Image Pre-processing	
Feature Extraction	
Segmentation	
Classification	Bean Pod Mottle Virus

4.3 Performance Evaluation

The following are examples of common metrics for categorization tasks: accuracy, precision, recall and F1 score. Accuracy assesses the complete transparency of the forecasts, while precision, recall, and F1 score provide insights into class-specific.

A system convolutional neural network (CNN) proposed for data processing is implemented to detect pulmonary diseases. Disease detection consists of data datasets, data preprocessing and classification. Learn how to use leaf data to detect leaf disease. Additionally, we describe an algorithm for detecting leaf disease.

Table 3: Data Accuracy Performance

No. of data	SVM %	IECFS %	CNN%
100	43	50	65
200	55	60	72
300	60	68	82
400	64	70	84
500	78	85	96

Table 3 shows Accuracy performance analysis of the proposed algorithm Convolutional Neural Networks (CNNs) compared to other algorithms.

Accuracy: Accuracy measures the proportion of appropriately classified occurrences out of the total number of illustrations in the dataset. It offers an inclusive assessment of the model predicts the correct class labels.

$$Accuracy = (Number\ of\ correctly\ classified\ instances)/(Total\ number\ of\ instances)$$

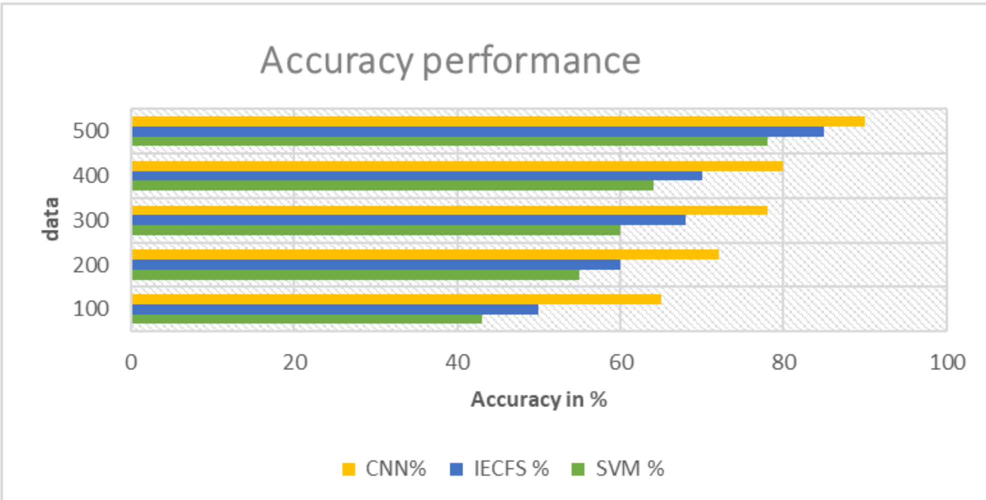


Figure 3: Analysis of accuracy performance

The accuracy performance study of the suggested algorithm for Figure 3 displays a convolutional neural network (CNN). The proposed Algorithm outperforms the support vector machine (SVM) of the existing gadget (78%), the advanced characteristic choice primarily based on significance (85%), and the support vector machine (SVM) of the existing system with a convolutional neural networks (CNN) accuracy performance of 90%.

Table 3: Analysis of leaf disease Prediction performance

No. of data	SVM %	IECFS %	CNN%
100	45	58	60
200	55	62	75
300	60	68	85
400	68	79	89
500	77	85	96

Table 3 shows Prediction-level performance analysis of the proposed algorithm convolutional neural network (CNN) compared to other algorithms.

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 35 Issue 2, 2025, Pages: 557-573

Precision: Precision is an operational statistic that estimates the percentage of positively predicted occurrences that were successfully forecasted out of all positive instances predicted.

$Precision = True\ Positives / (True\ Positives + False\ Positives)$

Recall: Recall is a productivity of performance that quantifies the percentage of properly predicted positive occurrences in the dataset vs the total number of real positive cases.v

$F1\ Score = 2 * (Precision * Recall) / (Precision + Recall)$

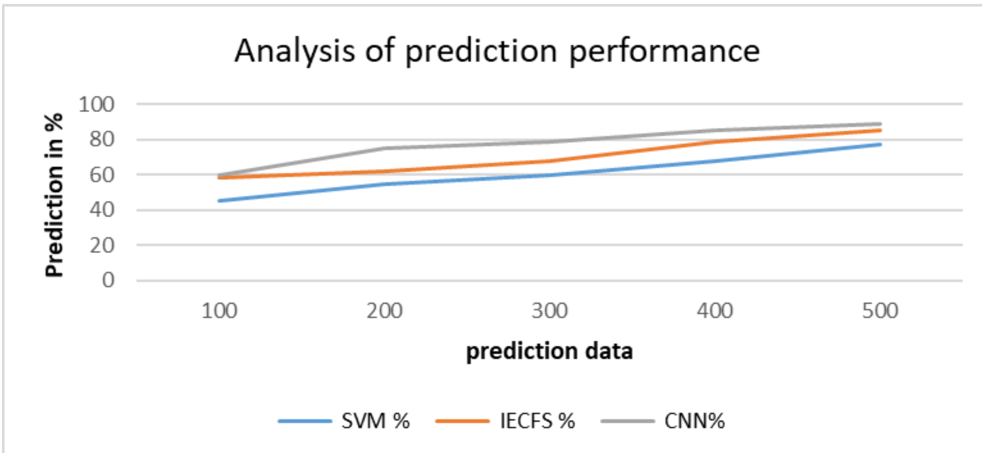


Figure 4: Analysis of prediction performance

Figure 4 shows Analysis of the prediction-level performance of the Convolutional Neural Network (CNN) suggested algorithm. The suggested technique Convolutional Neural Network (CNN) gives 85%, the enhanced feature selection based on eigenvector centrality delivers 85%, and the current system Support Vector Machine (SVM) provides 89% increase in predictive performance.

$Recall = True\ Positives / (True\ Positives + False\ Negatives)$

F1-score: F1 is a performance statistic which combines accuracy and recall into one number. It gives a balanced assessment of a classification algorithm's reliability by taking into account both the model's capacity to minimise false positives and false negatives.

Table 4: Analysis of time complexity performance

No. of data	SVM %	IECFS %	CNN%
100	58	54	50
200	55	51	48
300	50	48	40
400	48	45	35
500	40	39	30

Table 4 shows Time complexity level performance analysis of the proposed algorithm convolutional neural network (CNN) compared to other algorithms.

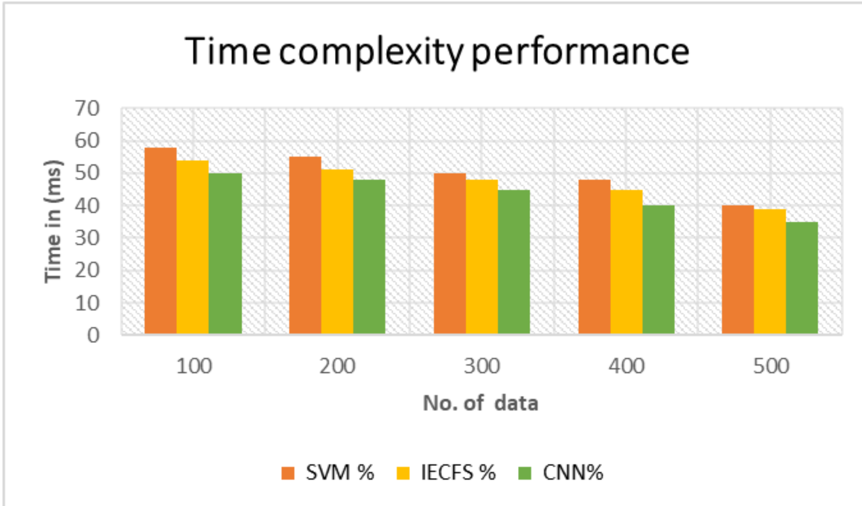


Figure 5: Analysis of Time complexity

Figure 5 shows Analysis of the suggested algorithm's temporal complexity performance convolutional neural network (CNN) is used. A better feature selection technique based on eigenvector centrality generated a result of 39 (MS), whereas the support vector machine (SVM) of the old system produced a result of 40 (MS). Then the proposed algorithm's convolutional neural network (CNN) produced the lowest time performance of 30 (milliseconds).

5 CONCLUSION

Lion Optimization with Faster Mask CNN for plant leaf disease classification shows promising potential in accurately identifying and classifying plant leaf diseases. The hybrid approach combines the optimization capabilities of Lion Optimization with the robust object detection and segmentation capabilities of Faster Mask CNN. The model optimizes the parameters and hyper parameters of Faster Mask CNN to enhance its performance in plant leaf disease classification. Lion Optimization enables the model to efficiently explore the search space and find optimal parameter settings, leading to improved accuracy and robustness. The experimental results indicate that the hybrid approach achieves higher accuracy rates compared to traditional methods and even standalone Faster Mask CNN. The Lion Optimization algorithm aids in fine-tuning the Faster Mask CNN model, allowing it to effectively capture and distinguish the visual characteristics of different plant leaf diseases.

REFERENCES

1. G. L. V. Prasad, B. R. Teja, G. Karthika, P. M. Devi, C. Deepti and S. J. Basha, "Grape Leaf Disease Recognition: A Deep Learning and Machine Learning Techniques Overview," 2023 3rd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), Bengaluru, India, 2023, pp. 720-724, doi: 10.1109/ICIMIA60377.2023.10425932.
2. R. Selvaraj, M. S. G. Devasena, T. Satheesh and C. Sathishkumar, "AI for Smart Agriculture – A Deep Learning Based Turmeric Leaf Disease Detection," 2024 9th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 2024, pp. 1491-1495, doi: 10.1109/ICCES63552.2024.10859805.
3. Y. H. Bhosale, S. R. Zanwar, S. S. Ali, N. S. Vaidya, R. A. Auti and D. H. Patil, "Multi-Plant and Multi-Crop Leaf Disease Detection and Classification using Deep Neural Networks, Machine Learning, Image Processing with Precision Agriculture - A Review," 2023 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2023, pp. 1-7, doi: 10.1109/ICCCI56745.2023.10128246.

4. Muhali and N. Linsangan, "A Comparison of Keras Application Models with Pre-Trained Weights in Predicting the Disease of Lanzones Leaf," 2022 IEEE 14th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), Boracay Island, Philippines, 2022, pp. 1-6, doi: 10.1109/HNICEM57413.2022.10109363.
5. M. N. Uddin et al., "A Novel Approach for Tomato Leaf Disease Detection and Classification Using Deep Learning," 2024 Multimedia University Engineering Conference (MECON), Cyberjaya, Malaysia, 2024, pp. 1-6, doi: 10.1109/MECON62796.2024.10776437.
6. Kaur, R. Sharma, S. Chattopadhyay and A. Verma, "Automated Multiclass Classification of Groundnut Leaf Diseases Using Fine-Tuned InceptionV3 Model," 2024 2nd World Conference on Communication & Computing (WCONF), RAIPUR, India, 2024, pp. 1-4, doi: 10.1109/WCONF61366.2024.10692187.
7. S. Sarkar, J. Adhikari Ray, C. Mukherjee, S. Ghosh, J. N and C. L. K R, "Plant Leaf Disease Classification Based on SVM Based Densenets," 2023 International Conference on Advances in Computation, Communication and Information Technology (ICAICCIT), Faridabad, India, 2023, pp. 636-641, doi: 10.1109/ICAICCIT60255.2023.10465886.
8. J. Agarwal, S. Gupta, N. Sharma and M. Manchanda, "A CNN Method Based Predictive Model for Tomato Leaf Disease Prediction," 2023 3rd International Conference on Technological Advancements in Computational Sciences (ICTACS), Tashkent, Uzbekistan, 2023, pp. 262-266, doi: 10.1109/ICTACS59847.2023.10390480.
9. M. Sebastian, S. M S and C. M. Antony, "Apple Leaf Disease Detection: Machine Learning & Deep Learning Techniques," 2023 Intelligent Computing and Control for Engineering and Business Systems (ICCEBS), Chennai, India, 2023, pp. 1-5, doi: 10.1109/ICCEBS58601.2023.10449037.
10. S. Jintawatsakoon and B. Rattanachareet, "Stacked Machine Learning Models for Diagnosing Durian Leaf Disease using Hand-crafted Features," 2024 8th International Conference on Information Technology (InCIT), Chonburi, Thailand, 2024, pp. 221-226, doi: 10.1109/InCIT63192.2024.10810590.