Chapter 3

CHAPTER 3

RESEARCH METHODOLOGY

Discussions of the study's goals, methodology, suggested models' architectures, dataset descriptions, evaluation measures, experimental design, and supporting hardware and software are included in this chapter.

3.1 INTRODUCTION

The increased demand for crops in modern times is primarily driven by the rapid growth of the population, leading to a higher demand for agricultural fields. However, there is a shortage of cultivation areas, which necessitates the use of better techniques and technologies to improve crop yields. Crops are vulnerable to disease infections, mainly caused by aggressive environmental conditions such as soil and weather factors. Powdery mildew, late blight, rusts, leaf spot, and Eriophyid mites are only some of the leaf diseases that are constantly threatened by the introduction of new and changing pathogens and pests. Treating plants affected by specific diseases with pesticides can help reduce the spread of diseases. Another method for controlling plant leaf diseases is to evaluate them early using cutting-edge scientific methods. However, fungicides and pesticides are used extensively, leading to higher production costs and the potential for harmful residual levels in agricultural goods, which may have adverse effects on human health. Therefore, having an accurate and efficient method for assessing plant leaf diseases and recommending suitable pesticides to improve crop productivity is crucial. The use of DL models is seen as an efficient method for identifying leaf diseases and pests and suggesting appropriate pesticide applications.

3.2 SCOPE OF THE RESEARCH

The scope of the research is unique and distinct, as follows:

- The proposed model can greatly enhance the training of DATFGAN, enabling the generation of high-resolution leaf images from low-resolution ones.
- The proposed model can detect a wide range of diseases in tomato, bell pepper, and potato plant leaves.

- Method for recommending soil fertiliser types for paddy fields with several classes.
- The proposed model can accurately predict suitable pesticides for controlling pests and leaf diseases, thereby promoting efficient crop productivity.

3.3 BLOCK DIAGRAM OF RESEARCH METHODOLOGY

Four independent phases of study are advised to boost crop output via the identification of leaf diseases and pests and the selection of suitable pesticides to manage the found diseases and pests. Fig. 3.1 presents a block schematic of the research's approach.

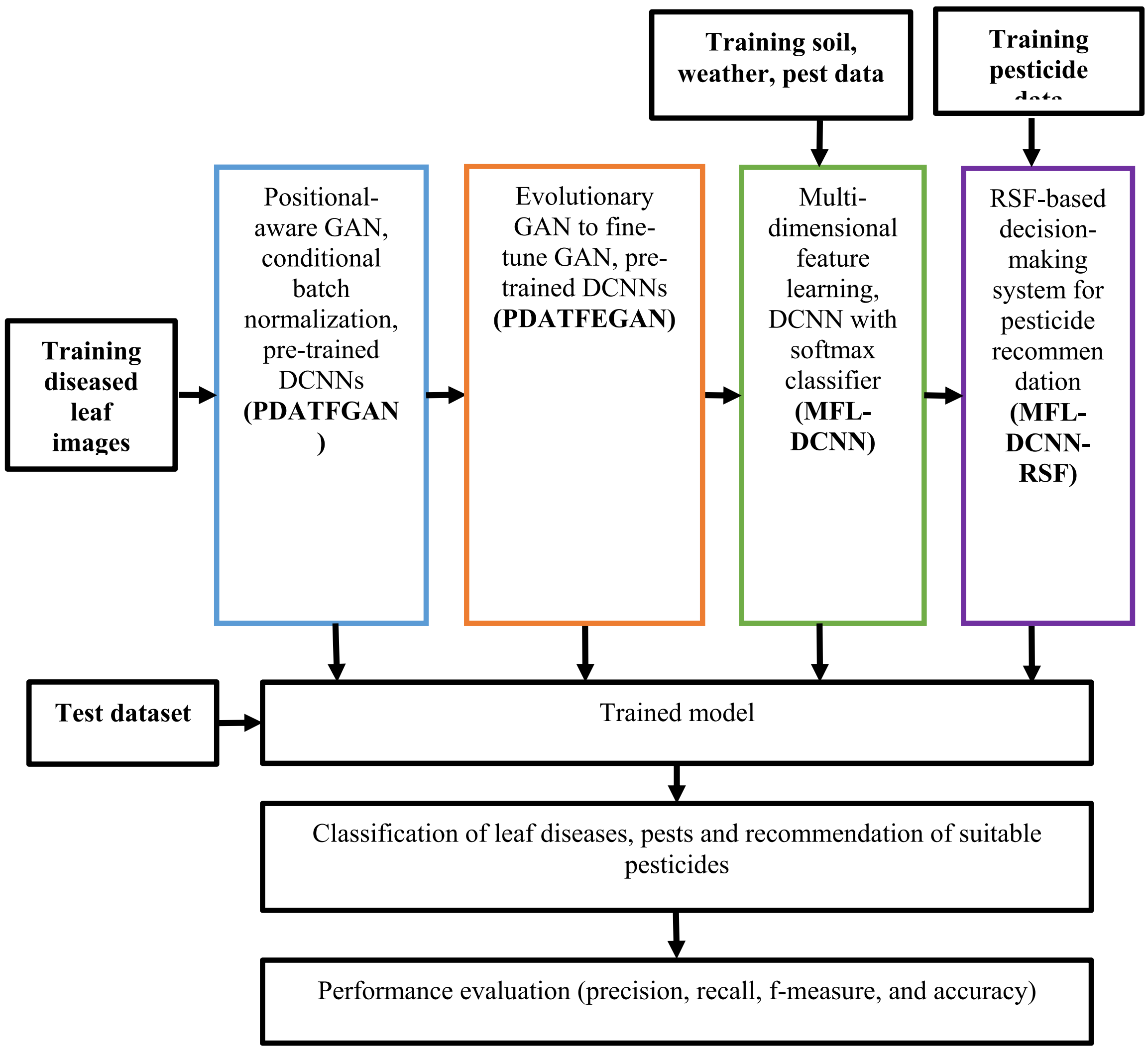

Fig. 3.1 Block Diagram of Proposed Research Methodology

3.3.1 Architecture of the Proposed Models

Four models are used in the processing of leaf disease and pest detection and in the recommendation of insecticides. In the first case, high-resolution leaf image generation is performed using a positional-aware GAN and classification is performed using three pre-learned CNN models. In the second case, high-resolution leaf image creation is performed using a positional-aware evolutionary GAN and classification is done using three pre-learned CNN models. Third, DCNN with a softmax classifier and multi-dimensional feature learning are used to detect leaf diseases and insects. In the fourth scenario, a decision support system based on RSF is created to suggest effective pesticides for known leaf diseases.

3.3.1.1 A Positional-Aware Dual-Attention and Topology-Fusion with Generative Adversarial Network-Based High-Resolution Framework for Leaf Disease Image Classification

A Positional-aware DATFGAN (PDATFGAN) model is developed in the first stages of this research to learn to identify a coordinate manifold that is analogous to the latent distribution manifold. The generator in this Positional-aware GAN (PGAN) model generates pictures of leaves using the pixel location and orientation of the leaves as the selection criterion. To generate patches at each sampled location, the generator first imposes position and orientation constraints on each pixel. The discriminator also learns to determine whether there is consistency and permanence between adjacent patches along the borders that separate them. The whole leaf image is then constructed by stitching together the high-resolution image patches. Therefore, conditional coordination in DATFGAN has a higher chance of producing images of outstanding quality than utilising simply DATFGAN. To further categorize the various leaf diseases, the collected leaf images are fed into the pre-trained DCNN models, notably ShuffleNetV3, DenseNet121, and MobileNetV3.

3.3.1.2 A Positional-Aware Dual-Attention and Topology-Fusion with Evolutionary Generative Adversarial Network for High-Resolution Diseased Leaf Image Generation and Classification

In the second phase, PDATF-Evolutionary GAN (PDATFEGAN) is built to further improve the generator and provide the high-resolution images required for classification. Because this model treats adversarial learning as an evolutionary issue, it makes use of an Evolutionary GAN (EGAN). An atmosphere may be represented by a discriminator, and a population of generators can adapt to it over time. The discriminator acquires the ability to distinguish between genuine and fake picture samples with each adversarial cycle. The generators play the role of parents and carry out various mutations in order to reproduce and adapt to the environment. To minimise losses between the produced distribution and the image distribution, multiple adversarial goal functions are explored, each of which provides a unique mutation. Next, an ideal discriminator is determined by calculating the quality and variety of images produced by the improved progeny. Thereafter, the "survival of the fittest" principle ensures that only the best-performing descendants are passed on to future generations for study. PDATFEGAN overcomes problems caused by using separate antagonistic learning goals and keeps the most desirable offspring produced by different goals. To further categorize leaf diseases, the resulting high-resolution pictures are supplied to the pre-trained DCNN models.

3.3.1.3 A Multi-Dimensional Feature Learning-Based DCNN (MFL-DCNN) for Leaf Disease and Pest Classification

In the third phase of the research, a unified pest detection and crop leaf disease classification model is proposed to improve crop productivity successfully. First, different diseased leaf images are converted into high-resolution images with the help of the PDATFEGAN model. Then, additional datasets such as pest, soil, and weather data related to leaf diseases are collected in this phase. These acquired data and images are submitted to the Multi-dimensional Feature Learning-based DCNN (MFL-DCNN) model to learn the link between pests and leaf diseases. Additionally, a softmax classifier is developed and used to categorize pests associated with certain leaf diseases. Thus, it supports farmers in efficiently recognizing pests related to specific leaf diseases.

3.3.1.4 Hybrid Rough Set with Intuitionistic Fuzzy Approximation Space (RSF)-Based Decision Support System for Pesticide Recommendation

In the last phase of the research, the suitable pesticides are proposed by combining the Multi-Functional Logistic Neural Network (MFL-DCNN) and the Rough Set (RS) on a Fuzzy approximation space (RSF)-based decision support system. By forecasting the most effective pesticide for a given crop, soil type, and climatic zone, it facilitates informed decision-making for the management of leaf diseases and pests. Using variables like leaf disease, weather, soil, and pests, the RSF method is used to develop the rule. An appropriate pesticide is suggested for managing leaf diseases and pests based on the criteria that have been developed to determine the association between these multi-dimensional data attributes. Increased crop yields are the end result of our MFL-DCNN-RSF model's ability to direct farmers towards the pesticides that will be most effective against a wide range of leaf diseases and pests.

3.4 DATASET DESCRIPTION

Kaggle users have contributed 2250 images of plant leaves affected by 15 distinct diseases to the PlantVillage Dataset (PVD). Tomatoes, peppers, and potatoes are all susceptible to these diseases. Diseases that can affect tomatoes include bacterial spot, early blight, late blight, leaf mould, septoria leaf spot, 2-spotted spider mites, target spot, yellow leaf curl virus, tomato mosaic virus, and tomato bacterial spot, early blight, late blight, leaf mould, septoria leaf spot, 2-spotted spider mites, and healthy pepper bells. One thousand five hundred (150) of these images are utilized for training (100 from each class), while seven hundred and fifty (750) are used for testing (50 from each class). It is available at https://www.kaggle.com/emmarex/plantdisease. Some actual leaf images from the PVD for different classes are shown in Fig 3.2.

Bell pepper healthy

Bell pepper bacterial spot

Potato healthy

Potato early blight

Potato late blight

Tomato healthy

Tomato target spot

Tomato mosaic virus

Tomato yellow leaf curl virus

Tomato bacterial spot

Tomato early blight

Tomato late blight

Tomato leaf mold

Tomato septoria leaf spot

Tomato two spotted spider mite

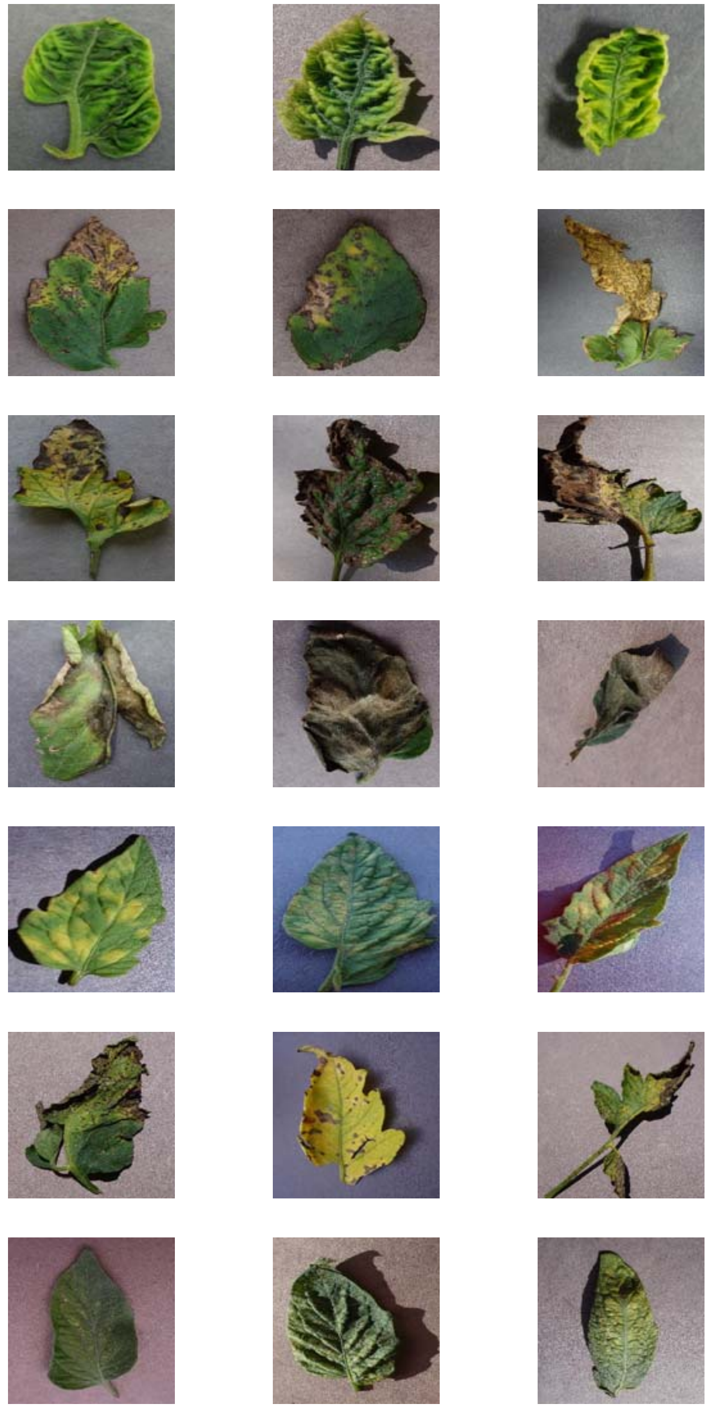

Fig 3.2 Input Samples for Various Leaf Infections images

Tomato mosaic virus, Tomato leaf curl virus, Xanthomonas campestris, Alternaria solani, Phytophthora infestans, Corynespora cassiicola, Xanthomonas gardneri, Alternaria tomatophila, Passalora fulva, Septoria lycopersici, and Tetranychidae are just a few examples of the 12 classes of leaf diseases represented in the pest dataset. In order to examine pest occurrence, scientists employ a meteorological dataset that includes time series of weather variables such as high and low temperatures, relative humidity in the morning and evening, precipitation, wind speed, and hours of daylight.

The 12 pesticides included in the dataset are: Cuprofix, Maneb, Mancozeb, Azoxystrobin, Sulfoxaflor, Pyrafluquinazon, BASF Cabriotop, Bonide Liquid Copper, Clutch, Spray Chlorothalonil, Copper Soap, and Bifenthrin.

Acidity, moisture content, and nutrient availability (such as nitrogen, phosphate, and potassium) are all part of the soil record. For these data sets, typically 70% is utilized for training and 30% is used for evaluation.

3.5 EVALUATION OF DIFFERENT MODELS

Several indicators, some of which are detailed below, are used to rate the four suggested models.

3.5.1 Performance Evaluation Metrics

Precision, recall, f-measure, and accuracy are some of the metrics used to evaluate a categorization system's effectiveness. Table 3.1 displays the categorized and actual values for all of the parameters in the confusion matrix.

Table 3.1 Confusion Matrix

		Actual	
		Positive	Negative
Classified	Positive	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

- TP stands for true positive, which shows that the actual class of the image is positive as well as the classified class, which also indicates that the image class is positive
- FP indicates that a positive class was assigned to an image while the true class is negative.
- TN means that both the actual class of the image and the class that was classified are negative
- FN refers to the fact that the actual class of the image is positive, even though the classified class is negative

Precision: As can be observed in Eq. (3.1), it is calculated by dividing the number of TPs by the sum of the TPs and FPs.

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{3.1}$$

$$\text{Precision} = \frac{\text{No.of exactly classified diseased leaves and pests}}{\text{No.of exactly classified diseased leaves and pests+No.of inexactly classified diseased leaves and pests}} \tag{3.2}$$

Recall: According to Eq. (3.3), it is the proportion of TP to the product of TP and FN. Recall might be anything from zero to one.

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \tag{3.3}$$

$$\text{Recall} = \frac{\text{No.of exactly classified diseased leaves and pests}}{\text{No.of exactly classified diseased leaves and pests+No.of inexactly classified healthy leaves and pests}} \tag{3.4}$$

F-measure: The F-measure is a DL statistic used for classification models. As shown in Eq. (3.5), the F-measure is derived by summing the accuracy and recall metrics.

$$\text{F} - \text{measure} = 2 \times \frac{\text{Precision}\cdot\text{Recall}}{\text{Precision+Recall}} \tag{3.5}$$

Accuracy: As shown in Eq. (3.6), it is calculated by dividing the entire number of genuine events (TP + TN) by the sum of the TPs, FPs, TNs, and FNs.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \tag{3.6}$$

3.5.2 Experimental Setup

In order to evaluate the suggested model, the simulation parameters batch size, optimizer, learning rate, and epochs are used.

- **Batch size:** It details the photos that are handled in a single loop. A hyperparameter is the minimum amount of data that must be analysed before adjusting the model's internal settings.

- **Optimizer:** In order to minimise the loss function, the optimizer algorithm makes changes to the network's weights and biases. Losses may be mitigated by using these optimization strategies. Adam optimizer, an alternative to Stochastic Gradient Descent (SGD) for building DL models, is employed in the research. It combines the best of AdaGrad with RMSProp for a wide range of optimization problems. It boasts various advantages, including minimal memory utilization and simplicity of use. The computer time savings is another bonus. Below, in Eqns. (3.7) and (3.8), is the mathematical description of the Adam optimizer. In Eqns. (3.7) and (3.8), α_1 and α_2 represent the average decay rates of slopes, δL is the loss function derivative, δw_t represents a time-dependent derivative of weights t, w_t stand for the weights at t, p_t is a set of gradations at t and q_t represents the square root of all previous gradients. Since p_t and q_t had both been set to zero to begin with, it is seen that they become 'biased towards 0' because $\alpha_1, \alpha_2 \approx 1$

$$p_t = \alpha_1 p_{t-1} + (1 - \alpha_1)\left[\frac{\delta L}{\delta w_t}\right] \tag{3.7}$$

$$q_t = \alpha_2 q_{t-1} + (1 - \alpha_2)\left[\frac{\delta L}{\delta w_t}\right]^2 \tag{3.8}$$

- **Learning rate:** In model training, the learning rate is a hyperparameter that controls the size of the training steps. It regulates training convergence and stability, and impacts learning rate.

- **Epochs:** It's the sum of all the times the neural network has been fed the dataset. Every single sample that was used for the training was allowed to adjust the internal model parameters once for every epoch that passed

3.5.3 Hardware and Software Used

For the implementation of the proposed methodology, Python 3.7 software containing Tensorflow and OpenCV libraries is utilized. Table 3.2 details the specifics of the system setup.

Table 3.2 System Configuration

Component	Specifications
Laptop	Processor Intel ® Core TM i5-7160U CPU @ 2.80GHz RAM 8 GB
Software	Python 3.7
Dataset collected	Kaggle dataset (PVD), pest, soil, weather and pesticide datasets

3.6 SUMMARY

This section outlined the approach used in each stage of the study and highlighted its significance.. Four models are suggested in this thesis work, and a summary of each is provided in this chapter. It also describes the evaluation metrics used for the proposed research experiments. This chapter significantly contributes to the thesis by facilitating a quick comprehension of the proposed procedures and models.