Pest Classification Using Efficientnetv2 and XG Boost Algorithms

Presentation By:

- Yeswanth Koti- 402834689
- Dhruvi Desai - 403309306
- Kris Kajar – 403327415
- Alma Campos – 307114150
- Gauri Joshi - 403333408

Table of Contents:

Abstract:

Introduction

- Pest classification plays a crucial role in agriculture and environmental management by enabling early detection and targeted control measures. However, accurately identifying pests from images poses significant challenges due to variations in species, life stages, and environmental conditions. Traditional methods often struggle with the complexity and variability of pest images, leading to suboptimal classification accuracy.

Problem Statement

The conventional methods of pest classification face several challenges:

- Variability: Pest species exhibit diverse morphological features and color patterns, making it difficult to develop robust classification models.

- Image Quality: Images captured under different lighting conditions and angles may vary in quality, affecting the performance of traditional image processing techniques.

- Scalability: With the increasing volume of image data, manual classification becomes impractical, necessitating automated and scalable solutions.

Abstract:

Solution Approach

To address the challenges in pest classification, we propose a novel approach leveraging state-of-the-art deep learning and machine learning algorithms:

- EfficientNetV2L: Utilizing the EfficientNetV2L architecture for feature extraction, which is known for its efficiency and effectiveness in handling diverse image data.

- XGBoost: Employing the XGBoost algorithm for classification, which excels in handling large-scale, high-dimensional datasets and provides robust performance across different domains’.

Tools and technologies Used:

1. Programming Languages

- Python: Main language for coding.

2. Libraries

- Tkinter: GUI development.
- NumPy: Numerical operations.
- Matplotlib: Data visualization.
- OpenCV: Image processing.
- Scikit-learn: Machine learning algorithms.
- Keras: Deep learning framework.
- XGBoost: Gradient boosting algorithm.

Dataset

Source: The dataset was collected from Kaggle, a popular platform for data science and machine learning enthusiasts.

Contents: The dataset consists of images depicting various types of pest insects commonly found in agricultural settings.

Total Images: The dataset comprises a total of 5494images.

Classes: There are 12 different classes of pest insects represented in the dataset:

1.Ants (499 images) 2.Bees (500 images) 3.Beetle (416 images) 4. Caterpillar (434 images) 5.Earthworms (323 images) 6.Earwig (466 images) 7.Grasshopper (485 images) 8.Moth (497 images) 9.Slug (391 images) 10.Snail (500 images) 11.Wasp (498 images) 12.Weevil (485 images)

Link: The dataset can be accessed on Kaggle via the following link(https://www.kaggle.com/code/vencerlanz09/pests-classification-using-efficientnetv2-l/notebook)

Modelling:

1. Data Collection and Preprocessing

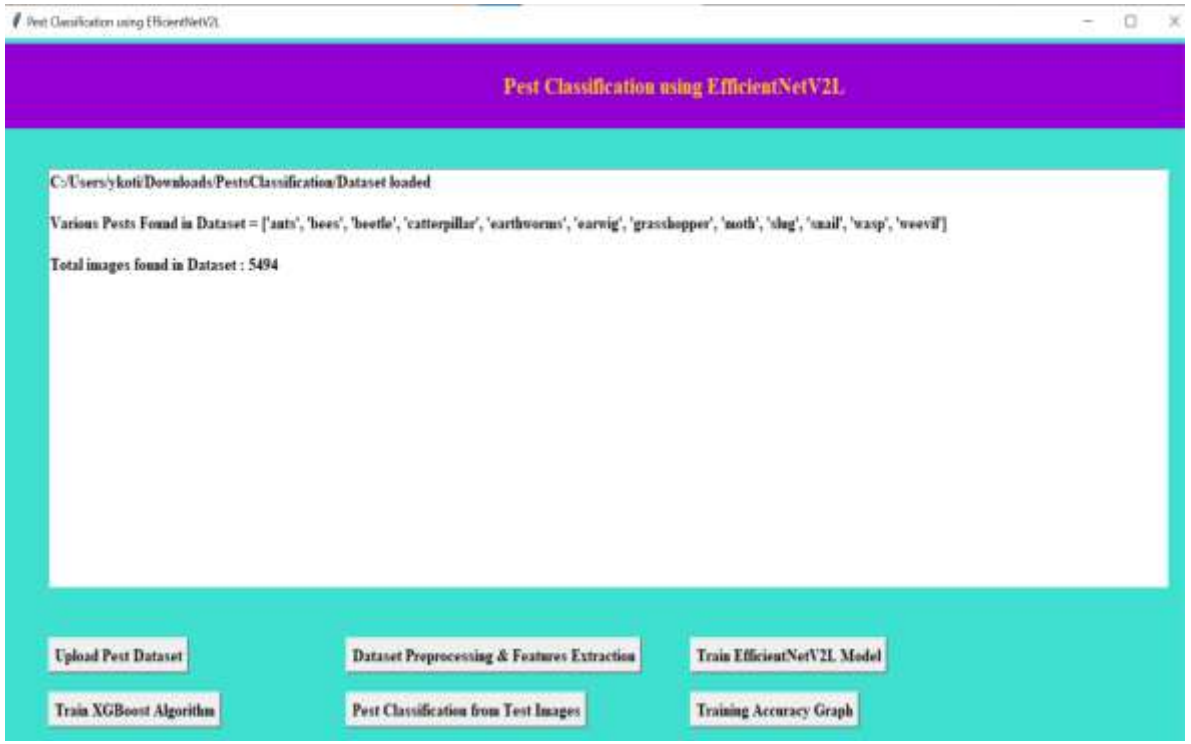
- Uploading and preprocessing the dataset.

```
def uploadDataset():
    global filename, X, Y, labels
    labels = []
    filename = filedialog.askdirectory(initialdir=".")
    # Code for displaying dataset loading confirmation and class labels
    ...
    # Preprocessing code (resizing, normalization) can be found in datasetPreprocessing function
    datasetPreprocessing()

def datasetPreprocessing():
    global X, Y
    # Data normalization
    X = X.astype('float32')
    X = X / 255
    # Shuffling and splitting dataset into train and test sets
    indices = np.arange(X.shape[0])
    np.random.shuffle(indices)
    X = X[indices]
    Y = Y[indices]
    Y = to_categorical(Y)
    X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.2)
    ...
```

Output:


Pest Classification using EfficientNetV2L
Pest Classification using EfficientNetV2L
Dataset Shuffling & Normalization Completed
Dataset train & test split as 80% dataset for training and 20% for testing
Training Size (80%): 4395
Testing Size (20%): 1099
Upload Pest Dataset
Dataset Preprocessing & Features Extraction
Train EfficientNetV2L Model
Train XGBoost Algorithm
Pest Classification from Test Images
Training Accuracy Graph


Pest Classification using EfficientNetV2L
Pest Classification using EfficientNetV2L
C:/Users/ykoti/Downloads/PestsClassification/Dataset loaded
Various Pests Found in Dataset = ['ants', 'bees', 'beetle', 'catterpillar', 'earthworms', 'earwig', 'grasshopper', 'moth', 'slug', 'snail', 'wasp', 'weevil']
Total images found in Dataset : 5494
Upload Pest Dataset
Dataset Preprocessing & Features Extraction
Train EfficientNetV2L Model
Train XGBoost Algorithm
Pest Classification from Test Images
Training Accuracy Graph

2. Model Training:

- EfficientNetV2L architecture for feature extraction.
- XGBoost classifier for comparison.

```
from keras.applications import EfficientNetV2S

def trainModel():
    global X_train, X_test, y_train, y_test, efficient_model, xg_cls
    # Training EfficientNetV2L model
    efficient_model = EfficientNetV2S(input_shape=(X_train.shape[1], X_train.shape[2], X_train.shape[3]), include_top=False,
    weights='imagenet')
    for layer in efficient_model.layers:
        layer.trainable = False
    model = Sequential()
    model.add(efficient_model)
    model.add(Flatten())
    model.add(Dense(units = 256, activation = 'relu'))
    model.add(Dense(units = y_train.shape[1], activation = 'softmax'))
    model.compile(optimizer = 'adam', loss = 'categorical_crossentropy', metrics = ['accuracy'])
    hist = model.fit(X_train, y_train, batch_size=32, epochs=35, validation_data=(X_test, y_test), verbose=1)

    # Training XGBoost classifier
    xg_cls = XGBClassifier()
    xg_cls.fit(X_train.reshape(X_train.shape[0], -1), np.argmax(y_train, axis=1))
```

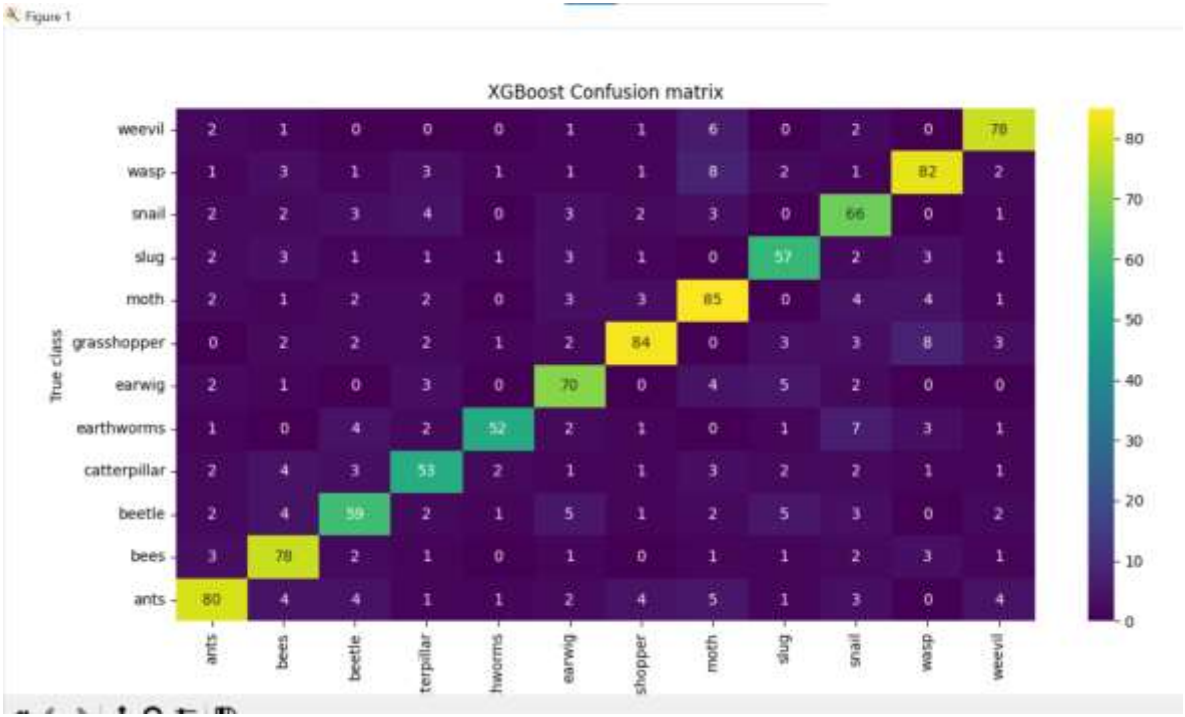
3.Evaluation Metrics

- Accuracy, precision, recall, and F1-score calculations.
- Confusion matrix visualization.

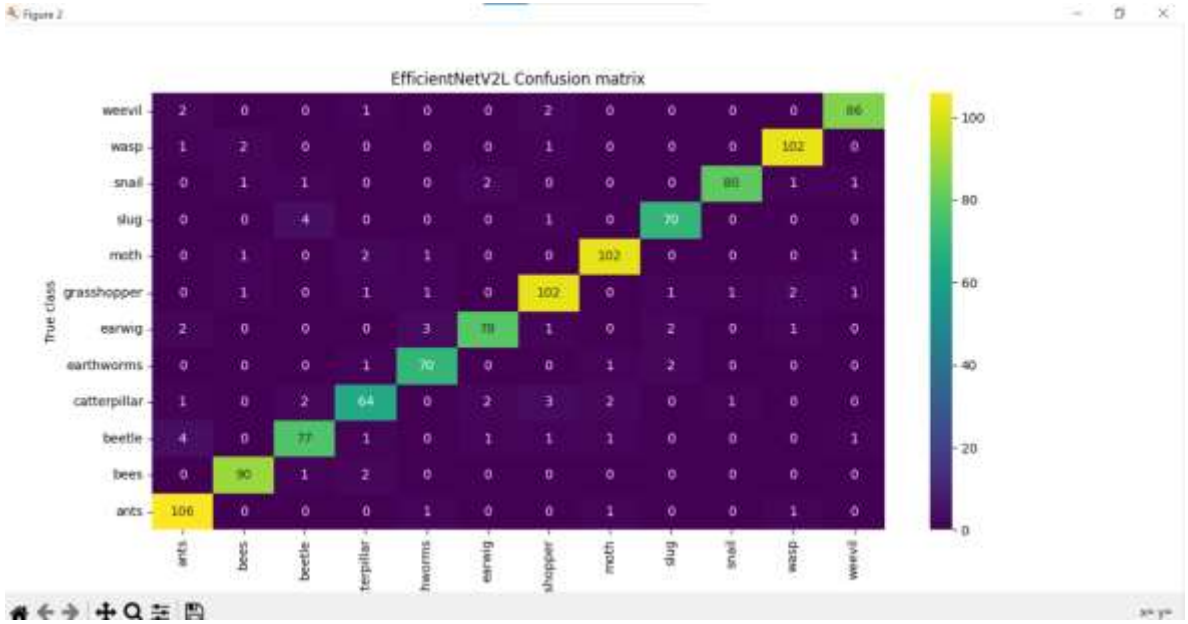
```
#function to calculate various metrics such as accura
def calculateMetrics(algorithm, predict, testY):
    global labels
    global accuracy, precision, recall, fscore
    p = precision_score(testY, predict,average='macro
    r = recall_score(testY, predict,average='macro')
    f = f1_score(testY, predict,average='macro') * 10
    a = accuracy_score(testY,predict)*100
    accuracy.append(a)
    precision.append(p)
    recall.append(r)
    fscore.append(f)
    text.insert(END,algorithm+' Accuracy  : '+str(a)+
    text.insert(END,algorithm+' Precision : '+str(p)+
    text.insert(END,algorithm+' Recall    : '+str(r)+
    text.insert(END,algorithm+' FSCORE    : '+str(f)+
    conf_matrix = confusion_matrix(testY, predict)
    plt.figure(figsize =(8, 5))
    ax = sns.heatmap(conf_matrix, xticklabels = label
    ax.set_ylim([0,len(labels)])
    plt.title(algorithm+" Confusion matrix")
    plt.ylabel('True class')
    plt.xlabel('Predicted class')
    plt.show()
```

Confusion Matrix visualization:

XGBoost



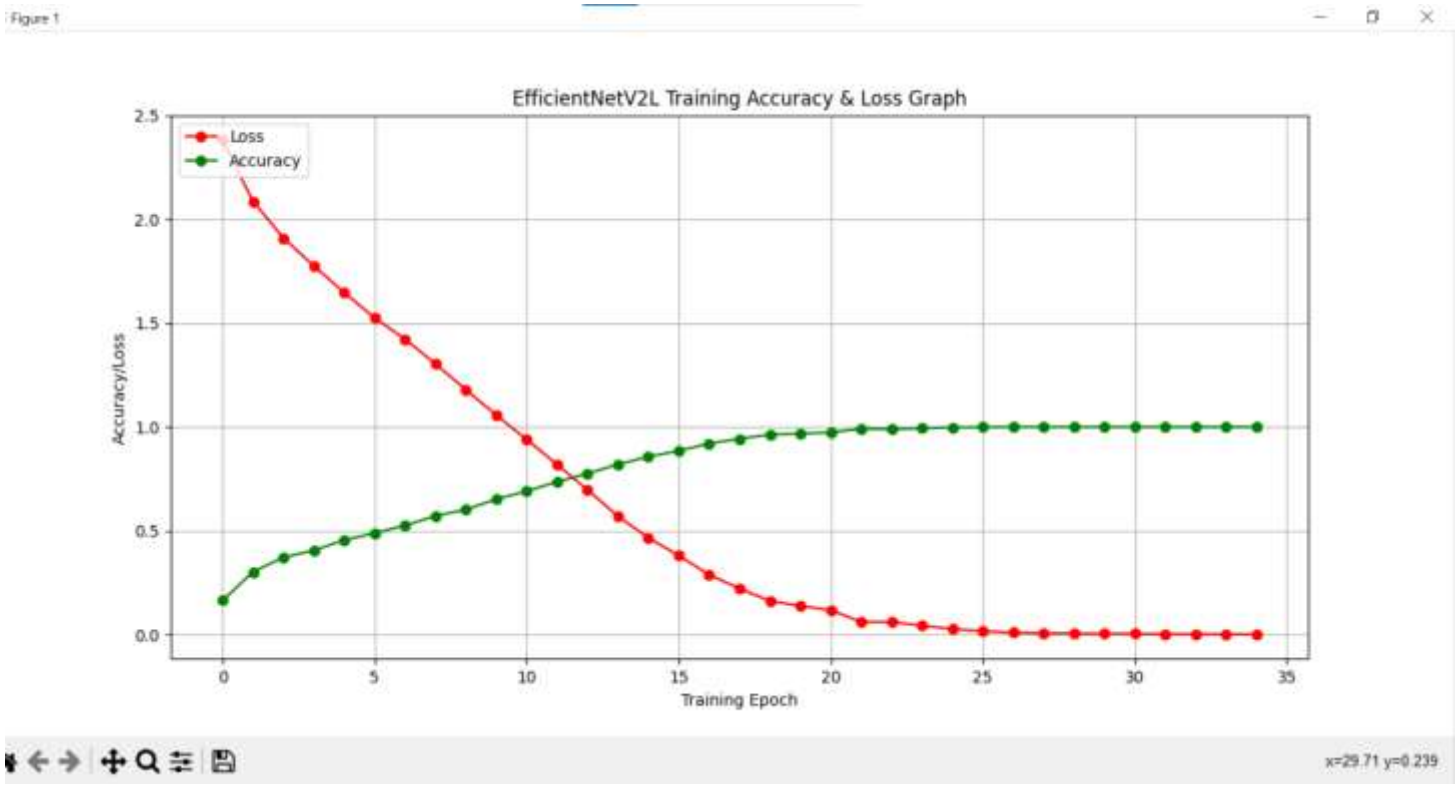
Efficient Net V2L



Results:

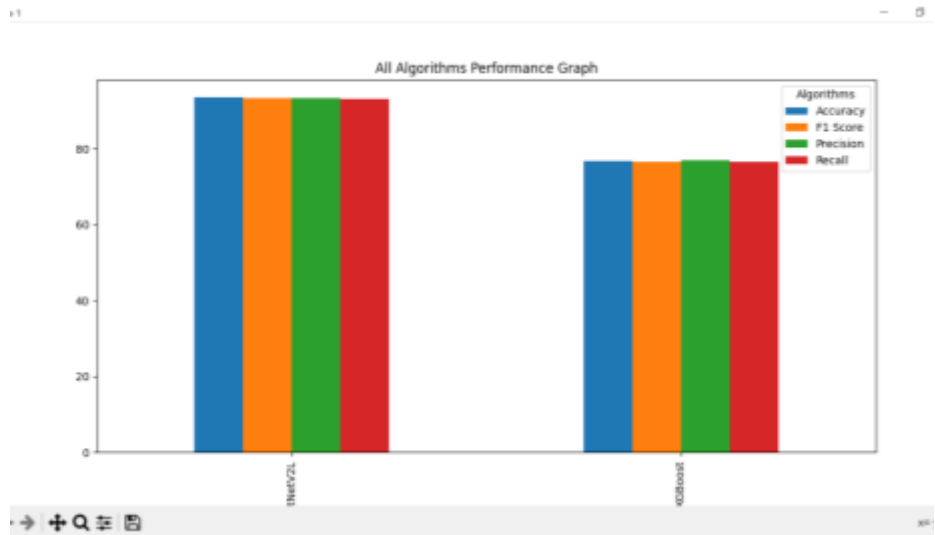
1. Training Progress:

- Graph showing training accuracy and loss.



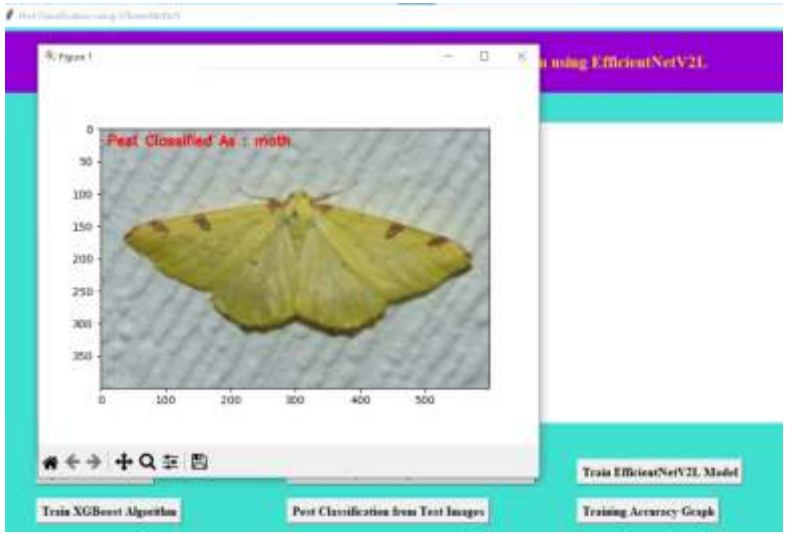
2.Model Performance

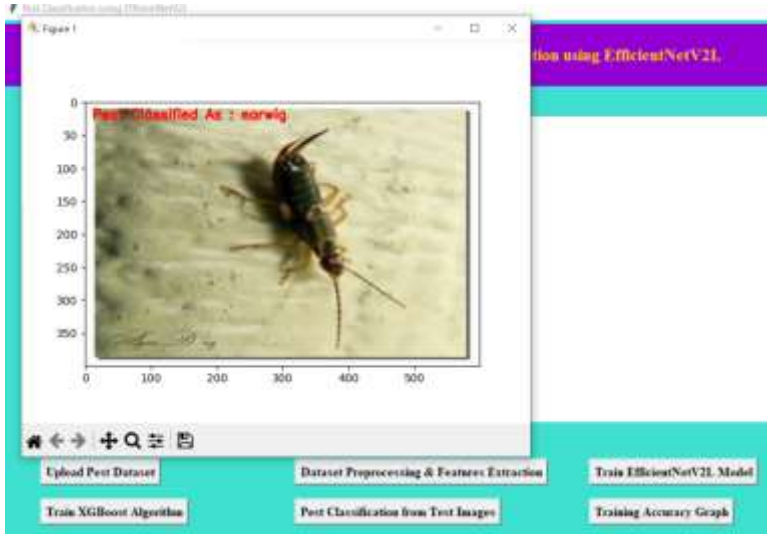
- Comparative analysis of EfficientNetV2L and XGBoost.
- Accuracy, precision, recall, and F1-score results.



3. Pest Classification

- Demonstration of classifying pests in test images.
- Displaying predicted pest class labels.





Conclusion:

- The integration of EfficientNetV2L and XGBoost has significantly improved the accuracy and efficiency of pest classification tasks. By leveraging the capabilities of deep learning and gradient boosting algorithms, we have achieved a high level of accuracy in distinguishing between different pest species, even in challenging environmental conditions.

- Moving forward, there is immense potential to integrate our pest classification system with existing pest management systems. By incorporating real-time pest detection capabilities into agricultural practices, we can enhance decision-making processes and optimize resource allocation. Furthermore, the scalability of our approach opens up opportunities for deployment in large-scale agricultural settings.

-Accurate pest classification is paramount for ensuring crop health, minimizing yield losses, and preserving the environment. Our project underscores the importance of leveraging advanced technologies to address complex challenges in agriculture and environmental sustainability. Through continued research and innovation, we can further enhance the effectiveness of pest management strategies and contribute to global food security efforts.

Thank you