Plant Image Classification Report

Introduction

In this project, we aim to develop a plant image classification model using convolutional neural networks (CNN). The purpose of this project is to classify plant images into two categories (P1 and P2) based on their visual features. This classification task can be valuable in various domains such as agriculture, environmental monitoring, and botany.

Dataset

We used a custom dataset of plant images located in the '/content/Plants' directory. The dataset contains images of plants from two different classes (P1 and P2). It consists of a total of 1253 images, with 1003 images for training and 250 images for testing. The dataset was manually labeled with the corresponding class labels.

Data Preprocessing

Before training the model, we performed the following data preprocessing steps:
- Video collection: Separate videos of the plants were collected and they were converted into different images.
- Image resizing: All images were resized to a uniform size of 224x224 pixels to ensure compatibility with the CNN model.
- Image normalization: Pixel values of the images were rescaled to the range of 0 to 1 by dividing each pixel value by 255.
- Data augmentation: We used the ImageDataGenerator class from Keras to apply data augmentation techniques such as rotation, zooming, and horizontal flipping. This helps in increasing the diversity of the training data and improving the model's generalization capability.

Model Architecture

The CNN model architecture used for plant image classification is as follows:
- Convolutional layer with 1 filter and a 3x3 kernel size, taking input images of size 224x224x3.
- Flatten the layer to convert the 3D feature maps into a 1D feature vector.
- Dense layer with 256 neurons and a ReLU activation function.
- Dense output layer with the number of classes (2) and a softmax activation function.

Model Training

The model was trained using the training generator generated by the ImageDataGenerator.
The following details summarize the training process:
- Optimizer: The model was compiled using the categorical cross-entropy loss function.

- Batch size: We set the batch size to 5 images per batch.
- Number of epochs: The model was trained for 1 epochs, iterating over the entire training dataset.
- Training performance: The model was trained on 1003 training images, and the training accuracy and loss were monitored throughout the training process.

Evaluation and Results

The trained model was evaluated on the testing dataset to measure its performance. The following steps were taken:
- Predictions: The model made predictions on the testing generator using the predict() method, producing a probability distribution for each image.
- Class labels: The class labels were determined by selecting the class with the highest predicted probability for each image.
- Performance metrics: Metrics such as accuracy and loss were calculated to evaluate the model's performance.

Based on the evaluation results, the model achieved test accuracy of 0.9520 with a loss of 0.1619

Conclusion

In conclusion, we have developed a plant image classification model using a CNN architecture. The model demonstrates promising performance in classifying plant images into two categories (P1 and P2). This project highlights the potential of CNNs in solving image classification tasks. Further improvements can be made by exploring different model architectures, hyperparameter tuning, and expanding the dataset.

References

Machine learning and deep learning by Achuthsankar