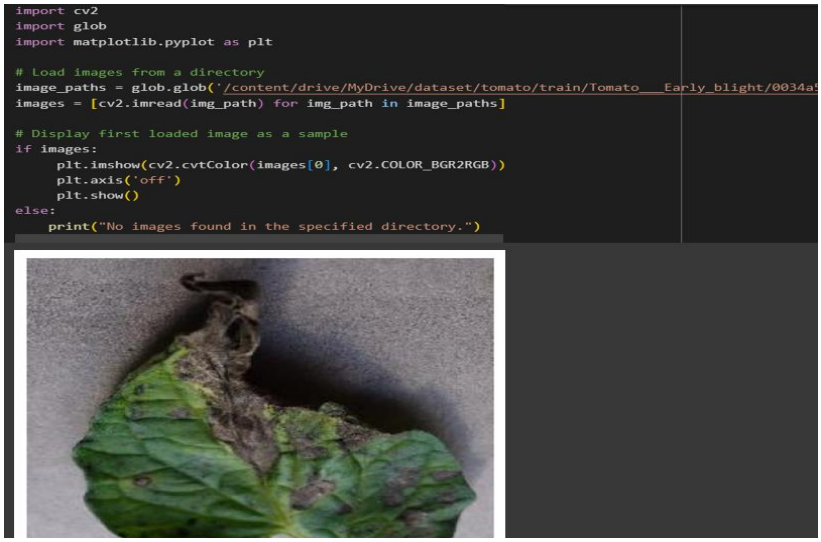
Data Collection and Preprocessing Phase

Date	11 November 2024
Team ID	team-739690
Project Title	Tomato Plant Disease Detection From Leaf Images Using Deep Learning
Maximum Marks	6 Marks

Preprocessing Template

The images will be preprocessed by resizing, normalizing, augmenting, denoising, adjusting contrast, detecting edges, converting color space, cropping, batch normalizing, and whitening data. These steps will enhance data quality, promote model generalization, and improve convergence during neural network training, ensuring robust and efficient performance across various computer vision tasks.

Section	**Description**
Data Overview	The dataset consists of images of tomato plant leaves affected by various diseases, such as bacterial spots, late blight, and healthy leaves. These images vary in resolution, lighting conditions, and orientations. The data is collected from publicly available Kaggle datasets (e.g., Tomato leaf disease detection)
Resizing	All images will be resized to a uniform target size of 224×224224 \times 224224×224 pixels to maintain consistency and compatibility with the Resnet architecture.
Normalization	Pixel values will be normalized to a range of 0 to 1 or scaled using mean subtraction and standard deviation to improve training convergence and stability.
Data Augmentation	Apply augmentation techniques such as flipping, rotation, shifting, zooming, or shearing.
Denoising	Apply denoising filters like Gaussian or median filters will be applied to reduce noise in images, improving the clarity of disease-related patterns.

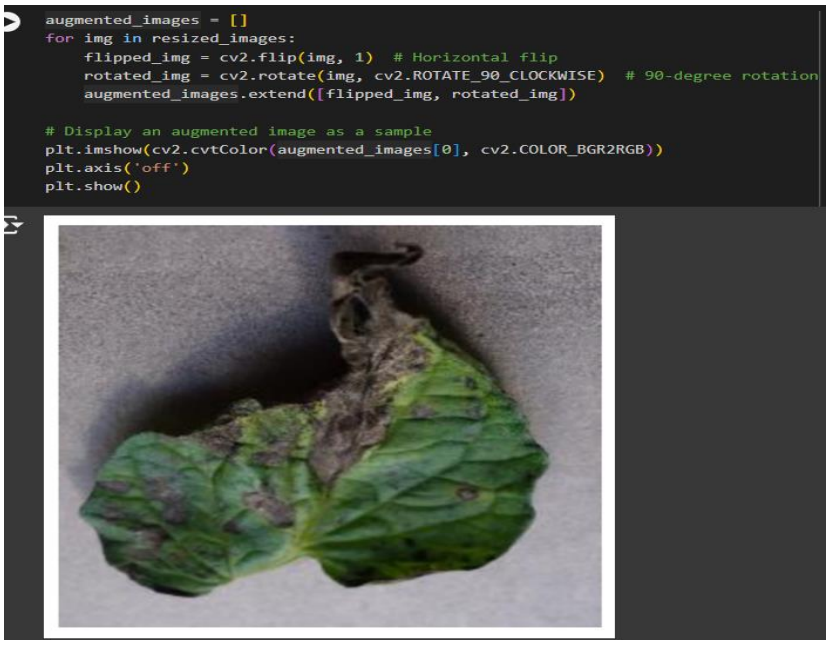
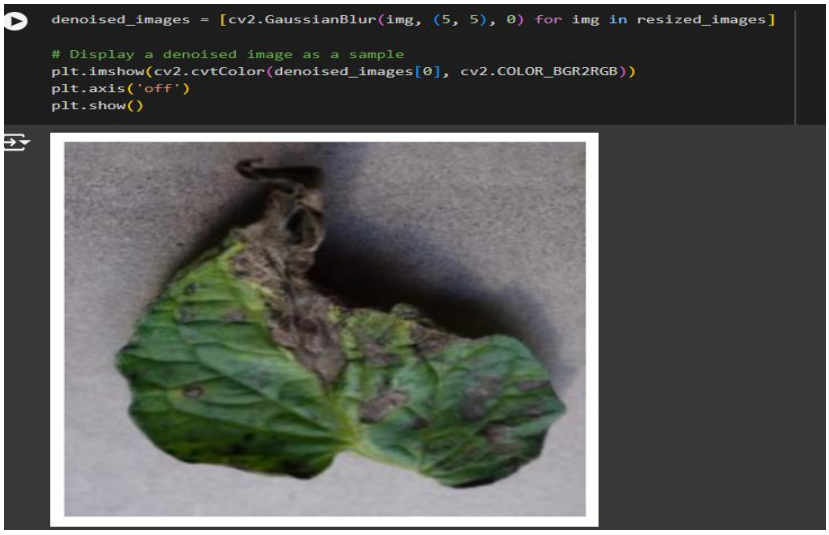
Edge Detection	Apply edge detection algorithms to highlight prominent edges in the images.
Color Space Conversion	Convert images from one color space to another.
Image Cropping	Crop images to focus on the regions containing objects of interest.
Batch Normalization	Batch normalization will be applied during training to normalize the activations within layers, improving convergence and generalization of the model.
Data Preprocessing Code Screenshots	
Loading Data	

```
import cv2
import glob
import matplotlib.pyplot as plt

# Load images from a directory
image_paths = glob.glob('/content/drive/MyDrive/dataset/tomato/train/Tomato___Early_blight/0034a5
images = [cv2.imread(img_path) for img_path in image_paths]

# Display first loaded image as a sample
if images:
    plt.imshow(cv2.cvtColor(images[0], cv2.COLOR_BGR2RGB))
    plt.axis('off')
    plt.show()
else:
    print("No images found in the specified directory.")
```

Resizing	
Normalization	

Data Augmentation	
Denoising	

Edge Detection	
Color Space Conversion	
Image Cropping	

Batch Normalization