M.B.M UNIVERSITY , JODHPUR

NEURAL NETWORK LABORATORY REPORT

MINI PROJECT 1

TEAM NAME : INVINCIBLE

TEAM MEMBERS : HARDIKA MAKWANA , PREETI DEORA , GUNJAN AGRAWAL , NEHA SHARMA

OBJECTIVE :

Build an image dataset that contains grayscale images of leaves of various plants/trees growing in MBMU campus. Perform necessary preprocessing steps to make the dataset uniform and ready for training. Train a deep autoencoder network that can reproduce any random image of a leaf from MBMU campus.

Bonus: Can we use this network to identify if a leaf is from MBM Campus or not?

DATASET PREPARATION AND STRATEGY :

- 7 leaf species folders - Neem , Mango , Jamun , Sadabahar , Lily , Paper flower , Gulmohar collected by all team members .
- Location: MBMU campus
- Total teams: 8
- Species per team: 2 species per team
- Images per species: 100-120 images
- Species assigned to our team :
  o Crinum asiaticum (Poison Bulb / Nagadamani)
  o Catharanthus roseus(Madagascar Periwinkle / Sadabahar)
- Format: Collect images in RGB format.
- Aspect Ratio: Maintain a 1:1 aspect ratio while capturing images.
- Image format: JPG
- Lighting Conditions: Capture images in both sunlight and artificial light.

PREPROCESSING :

- Resized all images to 256 * 256 resolution .
- Converted all images to jpg format .
- Applied adaptive gaussian thresholding on all images .
- **Augmentation**:
- Rotations: 0°, 90°, 180°
- Contrast adjustments: 1.0 (original), 1.3
- Total of **6 images** per original.

- Final Dataset:

  - Train_7sp: Augmented thresholded training set contains **5 species** .
  - Test_7sp: Augmented thresholded test set contains **2 species .**

MODEL ARCHITECTURE:

Encoder Architecture:

- Conv2D layer: 32 filters, kernel size = (3,3), activation = ReLU, padding = 'same'
- MaxPooling2D: pool size = (2,2), padding = 'same'
- Conv2D layer: 64 filters, kernel size = (3,3), activation = ReLU, padding = 'same'
- MaxPooling2D: pool size = (2,2), padding = 'same'

Decoder Architecture:

- Conv2D layer: 64 filters, kernel size = (3,3), activation = ReLU, padding = 'same'
- UpSampling2D: size = (2,2)
- Conv2D layer: 32 filters, kernel size = (3,3), activation = ReLU, padding = 'same'
- UpSampling2D: size = (2,2)
- Output Conv2D layer: 1 filter, kernel size = (3,3), activation = **Sigmoid**, padding = 'same'

Output Size: 256 × 256 × 1 (Reconstructed Grayscale Image)

HYPERPARAMETERS

- **Epochs:** 20
- **Batch Size:** 10
- **Learning Rate:** Default (0.001)
- **Optimiser :** Adam
- **Loss function :** Mean squared error

Bonus Classification

- **Threshold MSE for Classification:** 0.0225
- **Class 0:** Leaf belongs to MBMU (MSE ≤ 0.0225)
- **Class 1:** Leaf does not belong to MBMU (MSE > 0.0225)

DESCRIPTION OF CODE :

1. Load Dataset from Folder :

- Reads images from each subfolder, converts to grayscale, normalizes pixel values to [0,1], and reshapes to (256, 256, 1) for model compatibility.
- x_train and x_test hold all training and test grayscale images.

2. Autoencoder Model is Defined
   - Encoder and decoder is defined
3. model.compile() - Creates and compiles model with MSE loss and Adam optimizer.
4. Performance of model is evaluated using MSE , MAE and R2 SCORE and train and test mse loss and accuracy have been calculated.
5. Train the Autoencoder in which model learns compressed encoding and reconstruction simultaneously .
6. Evaluate Model on Test Set - Predicts on test data, flattens it, and computes evaluation metrics.
7. Save Accuracy & Loss Plots and model to Google Drive .
8. Reconstructed 5 images and saved to google drive .
9. Forward pass :
   - Inputs the image and regenerates image and computes mse loss .
10. Backpropagation :
    - Gradients of MSE are propagated back from output to encoder using chain rule.
    - Weights are updated using Adam Optimizer .

PERFORMANCE EVALUATION

Evaluation Metrics Used:

- Mean Squared Error (MSE)
- Mean Absolute Error (MAE)
- R^2 Score

Training Metrics (Epoch 20):

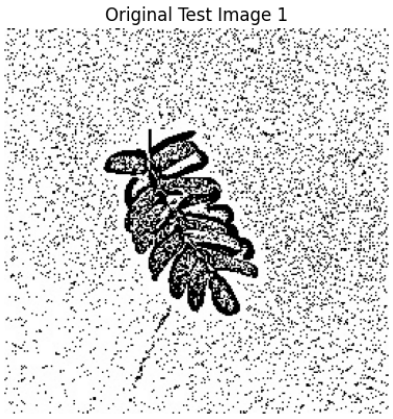
- Train MSE: 0.0241
- Train MAE: 0.0509
- Train R^2 Score: 0.8479
- Train Accuracy (MSE-based): 97.59%
- Train Accuracy (MAE-based): 94.91%

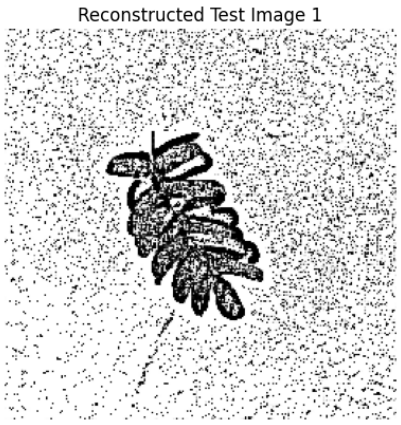
Test Set Performance:

- Test MSE: 0.0155
- Test MAE: 0.0342
- Test R^2 Score: 0.8733

Visualizations:

- Train loss curve (MSE) and train accuracy curve (1 - MAE) were plotted across epochs.
- Five random test images were selected and their reconstructions were visualized.

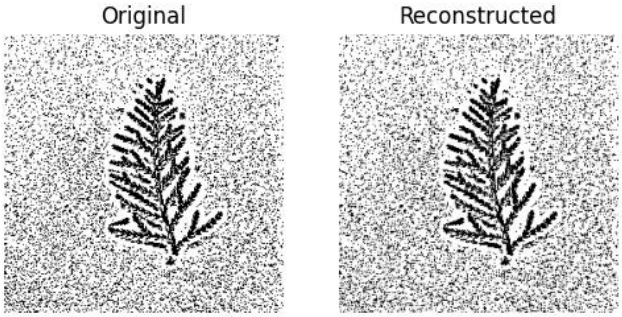




- Bonus Task :
- **Threshold MSE for Classification:** 0.0225
- **Class 0:** Leaf belongs to MBMU (MSE ≤ 0.0225)
- **Class 1:** Leaf does not belong to MBMU (MSE > 0.0225)
- **Model correctly classifies most of the test images as class 0 and class 1.**
- Sample test image :

  Reconstruction MSE: 0.022684

  Predicted Class: 1 - Non-MBMU Leaf (Class 1)



LIMITATIONS AND SCOPE OF IMPROVEMENT :

- Accuracy of model can further be improved by increasing the number of convolutional layers in model .
- Furthur we look forward to work towards improving Bonus task classification by improving dataset quality using preprocessing .
- Also we can improve data collection process while capturing the images .