Project: Deep Learning for Image Denoising and Enhancement

Overview

This project involves building a deep learning-based image denoising model using a convolutional autoencoder architecture. The model is trained to remove noise from images and enhance their quality, making it suitable for applications like image restoration, medical imaging, and low-light photography.

Key Components

- **Dataset**: BSD500 (Berkeley Segmentation Dataset) is used for training and evaluation. The dataset consists of noisy images and corresponding ground truth clean images.
- **Model**: The model is a **denoising autoencoder** based on a convolutional neural network (CNN) architecture, which consists of:
  - **Encoder**: Extracts features from the noisy input images.
  - **Decoder**: Reconstructs the cleaned images from the extracted features.
- **Loss Function**: **Mean Squared Error (MSE)** is used to minimize the pixel-wise differences between the denoised image and the ground truth.
- **Evaluation**: The model is evaluated on a validation set using loss metrics (MSE) and visual inspection of the denoised images.

Steps Involved

1. **Data Preparation**:
   - Download the BSD500 dataset containing noisy images and corresponding ground truth images.
   - Preprocess the images (resize and normalize them) for model training.
2. **Model Architecture**:
   - Designed a convolutional autoencoder with encoder and decoder components:
     - Encoder: Series of convolutional layers to extract features.
     - Decoder: Series of transpose convolutional layers to reconstruct the image.
3. **Model Training**:
   - Trained the model on noisy images to predict clean images.
   - Used **Mean Squared Error (MSE)** as the loss function to minimize the difference between the predicted and true images.
   - Trained for multiple epochs with batch processing using a learning rate scheduler for optimization.
4. **Model Evaluation**:
   - Evaluated the model using **Validation Loss** (MSE) and visual comparison of denoised and ground truth images.
   - Achieved a **Validation Loss of 0.0117**, indicating good performance.
5. **Model Saving**:
   - The trained model was saved for future use and inference.
   - Used `torch.save()` to save the model's state dictionary.
6. **Final Results**:

- Achieved promising results with visual improvements in the denoised images.
- Further improvements can be made by experimenting with hyperparameters, architectures, and data augmentation.

Technologies Used

- **Libraries/Frameworks**: PyTorch, OpenCV, NumPy, Matplotlib
- **Tools**: Jupyter Notebook
- **Hardware**: Trained on CPU (no GPU used due to driver issues)

Evaluation

- **Validation Loss**: 0.0117
- **Visual Results**: Comparison between denoised images and ground truth images shows good restoration.

Future Improvements

- Experiment with more advanced architectures like U-Net, ResNet, or GAN-based models for better performance.
- Explore different datasets to test the generalization of the model.
- Tune hyperparameters like learning rate, batch size, etc., for optimization.
- Add data augmentation techniques to increase model robustness.

Conclusion

This project demonstrates a deep learning approach to image denoising and enhancement. The model shows promising results in removing noise while preserving image details, making it suitable for real-world applications like medical image denoising, low-light image enhancement, and more.