Aryan Neizehbaz – 400222112
Auto Encoder (practical)

1) Data Explanation

MNIST Dataset

The MNIST (Modified National Institute of Standards and Technology) dataset is a well-known dataset in the field of machine learning and computer vision. It consists of 70,000 grayscale images of handwritten digits, each of size 28x28 pixels. The dataset is divided into 60,000 training images and 10,000 test images. Each image is labeled with a digit from 0 to 9, making it a 10-class classification problem. MNIST is commonly used for benchmarking and evaluating image processing and machine learning algorithms due to its simplicity and the ease of achieving high accuracy.

CIFAR-10 Dataset

The CIFAR-10 (Canadian Institute For Advanced Research) dataset is a widely used dataset for evaluating machine learning models, particularly in the context of image classification. It consists of 60,000 32x32 color images in 10 different classes, with 6,000 images per class. The dataset is divided into 50,000 training images and 10,000 test images. The 10 classes represent various objects and scenes such as airplanes, automobiles, birds, cats, deer, dogs, frogs, horses, ships, and trucks. CIFAR-10 is more complex than MNIST due to its higher resolution and the variety of classes, making it a common benchmark for evaluating more sophisticated models and techniques.

Combined Use of MNIST and CIFAR-10

In this project, we create a novel dataset by combining the MNIST and CIFAR-10 datasets. Each MNIST image is resized to match the CIFAR-10 image size (32x32 pixels) and expanded to three color channels. The combined images are generated by averaging the pixel values of corresponding MNIST and CIFAR-10 images. This novel dataset allows for the training and evaluation of models that can handle multi-domain input data, leveraging the simplicity of MNIST and the complexity of CIFAR-10 simultaneously.

2) Auto Encoder Architecture

- **Encoder:**

  - The encoder compresses the input image into a lower-dimensional latent representation.
  - It consists of several convolutional layers followed by a fully connected layer.
  - The encoder begins with a convolutional layer that applies multiple filters to the input image, reducing its spatial dimensions while increasing its depth. This is followed by a ReLU activation function to introduce non-linearity.
  - Another convolutional layer further reduces the spatial dimensions and increases the depth, followed by another ReLU activation function.
  - The output is then flattened into a one-dimensional tensor, which is passed through a fully connected layer to reduce the dimensionality to the latent space. The final ReLU activation function is applied to introduce non-linearity.

- **Decoder for CIFAR-10:**

  - This decoder reconstructs CIFAR-10 images from the latent representation.
  - It starts with a fully connected layer that expands the latent space back to a higher-dimensional space.
  - This is followed by a ReLU activation function, and the output is reshaped into a three-dimensional tensor with the original CIFAR-10 dimensions.
  - Transposed convolutional layers (also known as deconvolution) are then used to upsample the feature maps back to the original image size.
  - Each transposed convolutional layer is followed by a ReLU activation function to introduce non-linearity.
  - The final transposed convolutional layer reconstructs the three-channel CIFAR-10 image, and a Sigmoid activation function is applied to ensure the output pixel values are between 0 and 1.

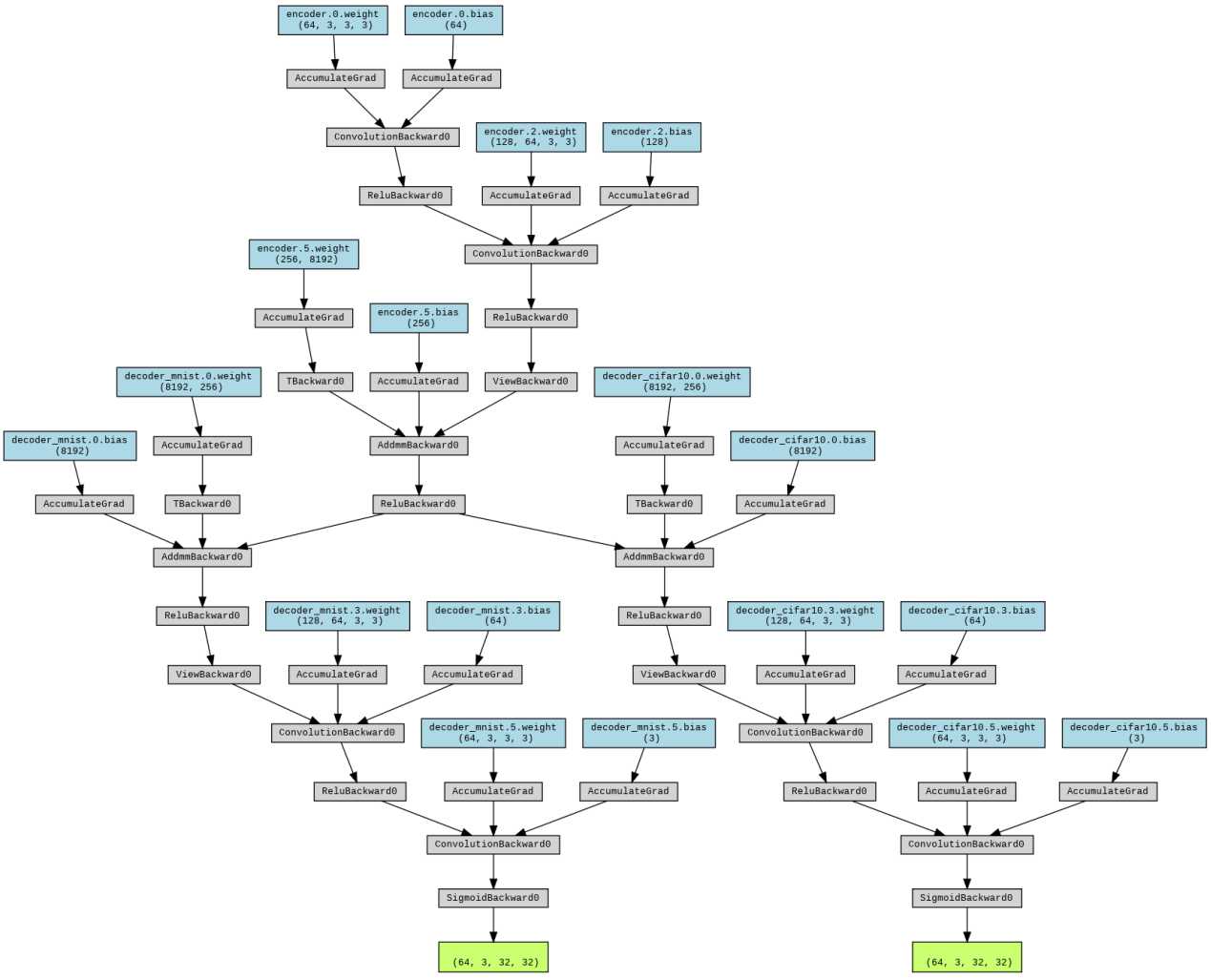
- **Decoder for MNIST:**

  - This decoder reconstructs MNIST images from the latent representation.
  - It also starts with a fully connected layer that expands the latent space.
  - This is followed by a ReLU activation function, and the output is reshaped into a three-dimensional tensor.
  - Transposed convolutional layers are used to upsample the feature maps back to the original image size.
  - Each transposed convolutional layer is followed by a ReLU activation function.
  - The final transposed convolutional layer reconstructs the MNIST image as a three-channel image, instead of the original single-channel, to match the CIFAR-10 format. A

Sigmoid activation function is applied to ensure the output pixel values are between 0 and 1.

- **Loss Function:**
  - The mean squared error (MSE) loss function is used to measure the difference between the reconstructed images and the original images. This loss function is suitable for regression tasks where the goal is to minimize the difference between predicted and actual values.

- **Optimizer:**
  - The Adam optimizer is used to update the model's parameters based on the computed gradients. Adam combines the advantages of two other popular optimization algorithms: AdaGrad and RMSProp. It is well-suited for problems with sparse gradients and noisy data.



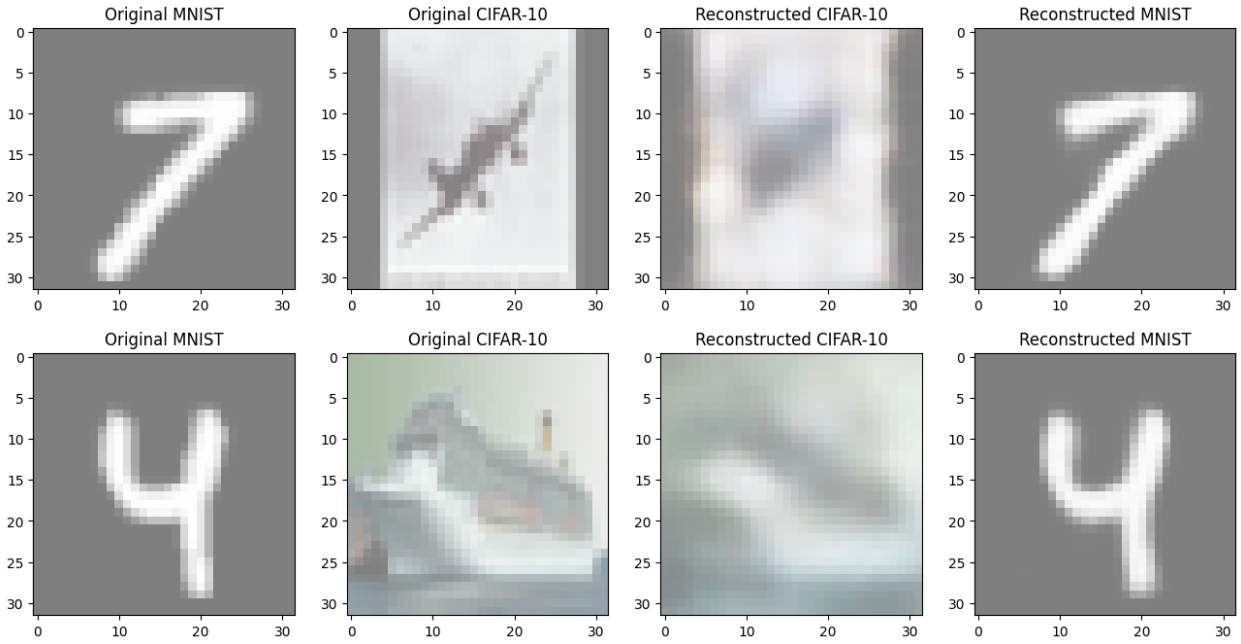
3) Evaluation Metrics

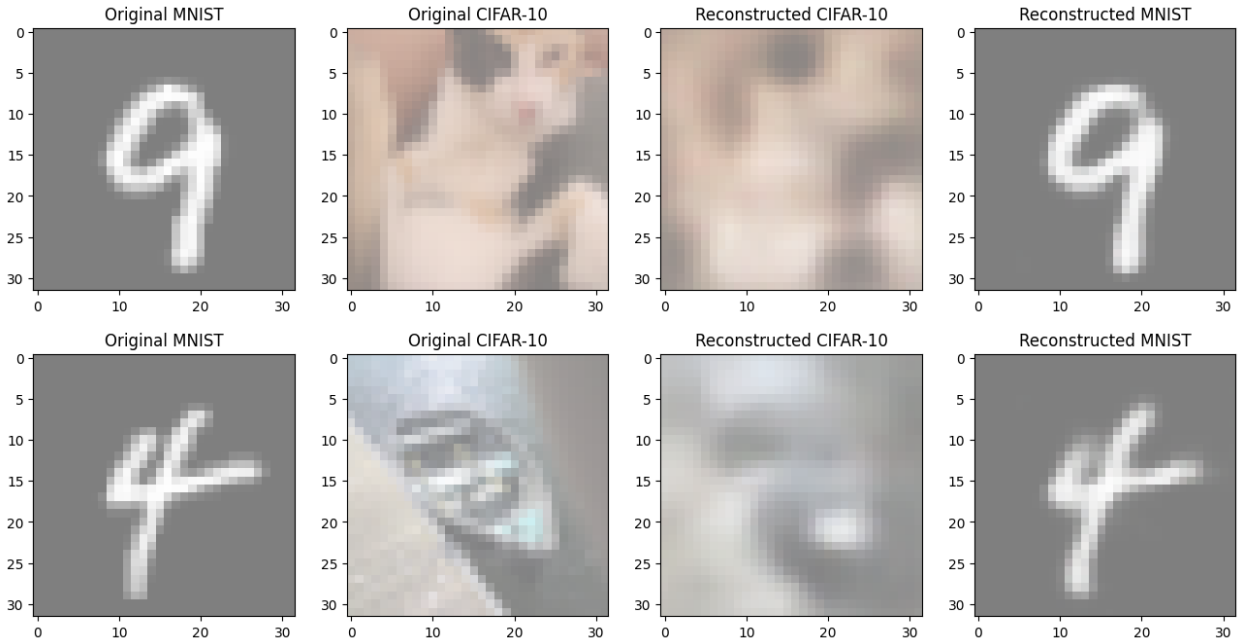
Structural Similarity Index (SSIM)

- **SSIM** is a perceptual metric that quantifies image quality degradation caused by processing such as data compression or transmission.
- It evaluates the similarity between two images based on luminance, contrast, and structure.
- The SSIM value ranges from -1 to 1, where 1 indicates perfect similarity.
- In this evaluation, the SSIM for CIFAR-10 is approximately 0.6727, and for MNIST, it is 0.9691. Higher SSIM values indicate better reconstruction quality.

Peak Signal-to-Noise Ratio (PSNR)

- **PSNR** is a widely used metric for assessing the quality of reconstructed images.
- It is the ratio between the maximum possible power of a signal (image) and the power of corrupting noise that affects the fidelity of its representation.
- The PSNR is measured in decibels (dB). Higher PSNR values indicate better image quality.
- In this evaluation, the PSNR for CIFAR-10 is approximately 18.7639 dB, and for MNIST, it is 27.9407 dB. Higher PSNR values indicate lower reconstruction error.




Original MNIST
Original CIFAR-10
Reconstructed CIFAR-10
Reconstructed MNIST
Original MNIST
Original CIFAR-10
Reconstructed CIFAR-10
Reconstructed MNIST