Experiment 12

Problem Statement:

To apply autoencoders on image dataset.

GitHub & Google Colab Link:

GitHub Link: https://github.com/piyush-gambhir/ncu-lab-manual-and-end-semester-projects/blob/main/NCU-CSL312%20-%20DL%20-%20Lab%20Manual/Experiment%2012/Experiment%2012.ipynb

Google Colab Link:

Installing Dependencies:

In [ ]:
```
! pip install tabulate numpy pandas matplotlib seaborn
```

```
Requirement already satisfied: tabulate in c:\users\mainp\appdata\local\programs\python\python311\lib\site-packa
ges (0.9.0)
Requirement already satisfied: numpy in c:\users\mainp\appdata\local\programs\python\python311\lib\site-packages
(1.26.4)
Requirement already satisfied: pandas in c:\users\mainp\appdata\local\programs\python\python311\lib\site-package
s (2.2.2)
Requirement already satisfied: matplotlib in c:\users\mainp\appdata\local\programs\python\python311\lib\site-pac
kages (3.8.4)
Requirement already satisfied: seaborn in c:\users\mainp\appdata\local\programs\python\python311\lib\site-packag
es (0.13.2)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\mainp\appdata\local\programs\python\python311\
lib\site-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in c:\users\mainp\appdata\local\programs\python\python311\lib\site-p
ackages (from pandas) (2024.1)
Requirement already satisfied: tzdata>=2022.7 in c:\users\mainp\appdata\local\programs\python\python311\lib\site
-packages (from pandas) (2024.1)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\mainp\appdata\local\programs\python\python311\lib\si
te-packages (from matplotlib) (1.2.1)
Requirement already satisfied: cycler>=0.10 in c:\users\mainp\appdata\local\programs\python\python311\lib\site-p
ackages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\mainp\appdata\local\programs\python\python311\lib\s
ite-packages (from matplotlib) (4.51.0)
Requirement already satisfied: kiwisolver>=1.3.1 in c:\users\mainp\appdata\local\programs\python\python311\lib\s
ite-packages (from matplotlib) (1.4.5)
Requirement already satisfied: packaging>=20.0 in c:\users\mainp\appdata\local\programs\python\python311\lib\sit
e-packages (from matplotlib) (24.0)
Requirement already satisfied: pillow>=8 in c:\users\mainp\appdata\local\programs\python\python311\lib\site-pack
ages (from matplotlib) (10.3.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\mainp\appdata\local\programs\python\python311\lib\si
te-packages (from matplotlib) (3.1.2)
Requirement already satisfied: six>=1.5 in c:\users\mainp\appdata\local\programs\python\python311\lib\site-packa
ges (from python-dateutil>=2.8.2->pandas) (1.16.0)
```

Code

In [ ]:
```
import numpy as np
import matplotlib.pyplot as plt
import tensorflow as tf
from tensorflow.keras import layers, models
from tensorflow.keras.datasets import cifar10
```

In [ ]:
```
# Load the dataset
(x_train, _), (x_test, _) = cifar10.load_data()
x_train = x_train.astype('float32') / 255.0
x_test = x_test.astype('float32') / 255.0

# Convolutional Autoencoder


def build_autoencoder():
    input_img = layers.Input(shape=(32, 32, 3))

    # Encoder
    x = layers.Conv2D(32, (3, 3), activation='relu', padding='same')(input_img)
```

```
    x = layers.MaxPooling2D((2, 2), padding='same')(x)
    x = layers.Conv2D(32, (3, 3), activation='relu', padding='same')(x)
    encoded = layers.MaxPooling2D((2, 2), padding='same')(x)

    # Decoder
    x = layers.Conv2DTranspose(
        32, (3, 3), strides=2, activation='relu', padding='same')(encoded)
    x = layers.Conv2DTranspose(
        32, (3, 3), strides=2, activation='relu', padding='same')(x)
    decoded = layers.Conv2D(3, (3, 3), activation='sigmoid', padding='same')(x)

    # Autoencoder model
    autoencoder = models.Model(input_img, decoded)
    autoencoder.compile(optimizer='adam', loss='binary_crossentropy')

    return autoencoder


autoencoder = build_autoencoder()
autoencoder.summary()

# Training the autoencoder
autoencoder.fit(x_train, x_train,
                epochs=20,
                batch_size=64,
                shuffle=True,
                validation_data=(x_test, x_test))

# Display the results


def display_images(original, decoded, n=10):
    plt.figure(figsize=(20, 4))
    for i in range(n):
        # Display original images
        ax = plt.subplot(2, n, i + 1)
        plt.imshow(original[i].reshape(32, 32, 3))
        plt.title("Original")
        plt.gray()
        ax.get_xaxis().set_visible(False)
        ax.get_yaxis().set_visible(False)

        # Display reconstructed images
        ax = plt.subplot(2, n, i + 1 + n)
        plt.imshow(decoded[i].reshape(32, 32, 3))
        plt.title("Reconstructed")
        plt.gray()
        ax.get_xaxis().set_visible(False)
        ax.get_yaxis().set_visible(False)
    plt.show()


# Encode and decode images
decoded_imgs = autoencoder.predict(x_test)

# Display original and reconstructed images
display_images(x_test, decoded_imgs)
```

Model: "functional_1"

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	(None, 32, 32, 3)	0
conv2d (Conv2D)	(None, 32, 32, 32)	896
max_pooling2d (MaxPooling2D)	(None, 16, 16, 32)	0
conv2d_1 (Conv2D)	(None, 16, 16, 32)	9,248
max_pooling2d_1 (MaxPooling2D)	(None, 8, 8, 32)	0
conv2d_transpose (Conv2DTranspose)	(None, 16, 16, 32)	9,248
conv2d_transpose_1 (Conv2DTranspose)	(None, 32, 32, 32)	9,248
conv2d_2 (Conv2D)	(None, 32, 32, 3)	867

Total params: 29,507 (115.26 KB)

Trainable params: 29,507 (115.26 KB)

Non-trainable params: 0 (0.00 B)

```
Epoch 1/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 61s 73ms/step - loss: 0.5942 - val_loss: 0.5630
Epoch 2/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 62s 78ms/step - loss: 0.5609 - val_loss: 0.5591
Epoch 3/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 72s 91ms/step - loss: 0.5580 - val_loss: 0.5580
Epoch 4/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 113s 144ms/step - loss: 0.5572 - val_loss: 0.5570
Epoch 5/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 129s 164ms/step - loss: 0.5566 - val_loss: 0.5566
Epoch 6/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 109s 122ms/step - loss: 0.5561 - val_loss: 0.5560
Epoch 7/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 105s 134ms/step - loss: 0.5552 - val_loss: 0.5559
Epoch 8/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 105s 86ms/step - loss: 0.5546 - val_loss: 0.5557
Epoch 9/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 87s 111ms/step - loss: 0.5551 - val_loss: 0.5551
Epoch 10/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 88s 112ms/step - loss: 0.5533 - val_loss: 0.5548
Epoch 11/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 86s 110ms/step - loss: 0.5540 - val_loss: 0.5552
Epoch 12/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 79s 101ms/step - loss: 0.5537 - val_loss: 0.5546
Epoch 13/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 69s 88ms/step - loss: 0.5534 - val_loss: 0.5555
Epoch 14/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 67s 86ms/step - loss: 0.5530 - val_loss: 0.5550
Epoch 15/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 76s 98ms/step - loss: 0.5532 - val_loss: 0.5541
Epoch 16/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 108s 131ms/step - loss: 0.5538 - val_loss: 0.5543
Epoch 17/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 77s 98ms/step - loss: 0.5527 - val_loss: 0.5543
Epoch 18/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 76s 95ms/step - loss: 0.5534 - val_loss: 0.5537
Epoch 19/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 67s 85ms/step - loss: 0.5534 - val_loss: 0.5538
Epoch 20/20
782/782 ━━━━━━━━━━━━━━━━━━━━ 80s 82ms/step - loss: 0.5530 - val_loss: 0.5537
313/313 ━━━━━━━━━━━━━━━━━━━━ 7s 21ms/step
```