USN NUMBER:1RVU22CSE071
NAME:Joselyn Riana Manoj

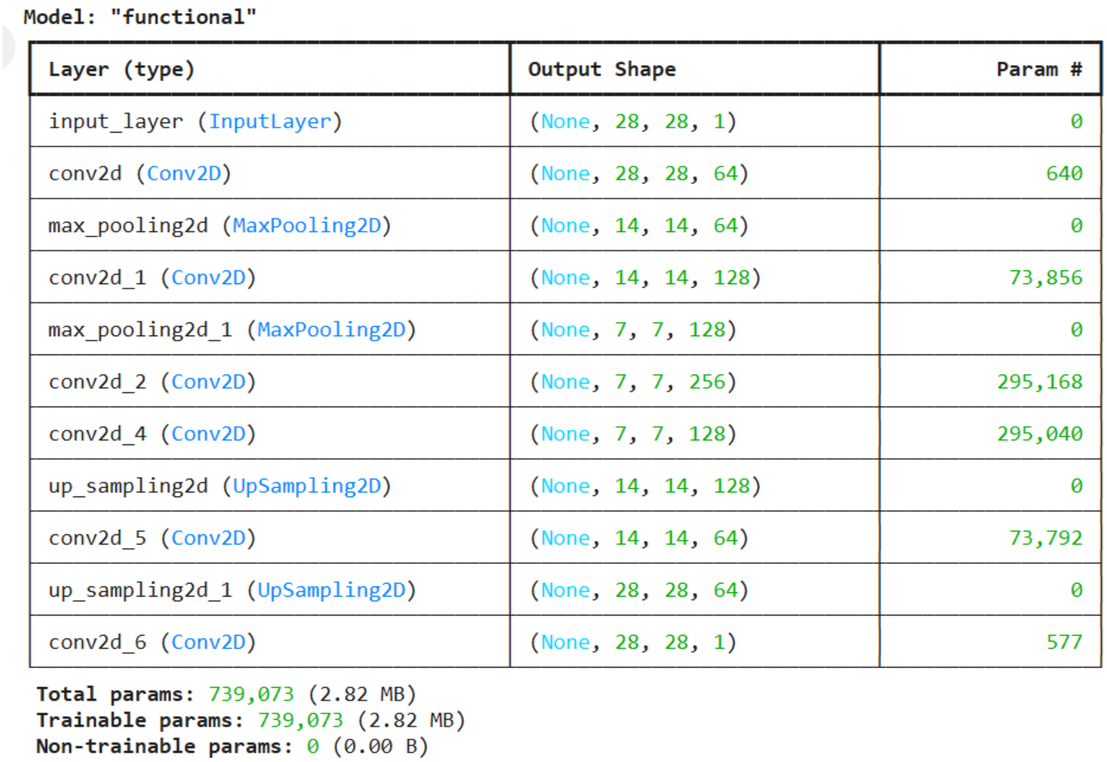
Ex No: 6.4 **Date: 11-09-2024**	**Lab 6.4: Denoising with a CNN Autoencoder**

Objective:

The goal of this lab is to develop a Convolutional Autoencoder model to denoise images from the Fashion MNIST dataset. An autoencoder is a type of neural network used for learning efficient codings of unlabeled data. This project aims to remove noise from the dataset images by training a model that can take noisy images as input and output clean images.

Descriptions:

The model architecture comprises an encoder, a bottleneck, and a decoder. The encoder compresses the input image into a lower-dimensional representation, the bottleneck further processes this representation, and the decoder reconstructs the image from the compressed data. During training, the model uses noisy images to learn how to produce noise-free outputs, thereby improving its ability to generalize to unseen noisy data.



Model: "functional"

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	(None, 28, 28, 1)	0
conv2d (Conv2D)	(None, 28, 28, 64)	640
max_pooling2d (MaxPooling2D)	(None, 14, 14, 64)	0
conv2d_1 (Conv2D)	(None, 14, 14, 128)	73,856
max_pooling2d_1 (MaxPooling2D)	(None, 7, 7, 128)	0
conv2d_2 (Conv2D)	(None, 7, 7, 256)	295,168
conv2d_4 (Conv2D)	(None, 7, 7, 128)	295,040
up_sampling2d (UpSampling2D)	(None, 14, 14, 128)	0
conv2d_5 (Conv2D)	(None, 14, 14, 64)	73,792
up_sampling2d_1 (UpSampling2D)	(None, 28, 28, 64)	0
conv2d_6 (Conv2D)	(None, 28, 28, 1)	577

```
Total params: 739,073 (2.82 MB)
Trainable params: 739,073 (2.82 MB)
Non-trainable params: 0 (0.00 B)
```

USN NUMBER:1RVU22CSE071
NAME:Joselyn Riana Manoj

Steps to Build the Model:

Import Libraries:

Import necessary libraries such as TensorFlow, TensorFlow Datasets, NumPy, and Matplotlib.

Data Preparation:

Load the Fashion MNIST dataset using tensorflow_datasets.
Create a function map_image_with_noise that normalizes the images and adds random noise to the input images.
Prepare training and testing datasets using map, shuffle, batch, and repeat methods to preprocess the images.

Define Encoder:

Create an encoder function using two Conv2D layers followed by MaxPooling2D layers to downsample the input images and capture relevant features.

Define Bottleneck:

Create a bottle_neck function using a Conv2D layer to further process the encoded image features. Add another Conv2D layer with a sigmoid activation to visualize the encoded representation.

Define Decoder:

Create a decoder function to upsample the encoded images back to the original size using Conv2D and UpSampling2D layers, aiming to reconstruct the input image.

Build the Convolutional Autoencoder:

Define the convolutional_auto_encoder function to combine the encoder, bottleneck, and decoder to form the complete model.
Create two models: the full autoencoder and a separate encoder model for visualization.

Compile and Train the Model:

Compile the autoencoder model using the Adam optimizer and binary cross-entropy loss.
Train the model on the noisy Fashion MNIST dataset for 40 epochs using the fit method, with training and validation steps defined.

Display Results:

Define helper functions display_one_row and display_results to visualize the input images,

encoded representations, and reconstructed outputs.
Test the model on a batch of test images, display the noisy input, encoded output, and the denoised reconstruction side-by-side.

Evaluate Model Performance:

Take a sample of test images and feed them through the trained encoder and autoencoder models.
Use the display_results function to visually assess how well the model removes noise from the test images.

GitHubLink:
https://github.com/joselynrianaaa/DeepLearning_Labs/blob/main/Lab_Encoder_4_Denoising_DISTRI.ipynb