Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

AY: 2025-26

Class:	AI	Semester:	VII
Course Code:		Course Name:	DL

Name of Student:	BARI ANKIT VINOD
Roll No. :	61
Assignment No.:	4
Title of Assignment:	
Date of Submission:	
Date of Correction:	

Evaluation

Performance Indicator	Max. Marks	Marks Obtained
Completeness	5	
Demonstrated Knowledge	3	
Legibility	2	
Total	10	

Performance Indicator	Exceed Expectations (EE)	Meet Expectations (ME)	Below Expectations (BE)
Completeness	5	3-4	1-2
Demonstrated Knowledge	3	2	1
Legibility	2	1	0

Checked by

Name of Faculty : Raunak Joshi

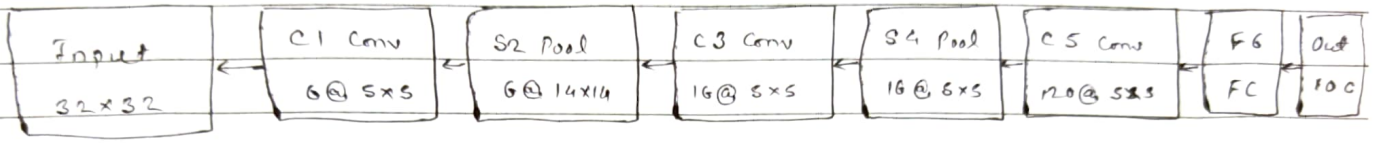
Signature :

Date :

Assignment No. - 4

1) Imagine you are working on a project to develop a handwritten digit recognition system for a small-scale banking application. The goal is to accurately classify digits (0-9) from images of handwritten checks, with appropriate diagram show how LeNet architecture is used to perform this tasks.

→ import matplotlib.pyplot as plt



- Input - greyscale image, typically 32×32 pixels.
  - If your captured digits are 28×28 or arbitrary size change.
- C1 - Convolutional layer
  - 6 filters, kernels - 5×5, produces 6 feature map of 28×28.
  - original used tanh activation, in practice use ReLU.
- S2 - Subsampling / pooling.
  - Average pooling with 2×2 subsamples → 6 maps of 14×14
  - modern practice - use max-pooling.
- C3 - Convolutional layer
  - 16 filters of size 5×5 → yields 16 maps of 10×10.
- S4 - Subsampling / pooling
  - 2×2 pooling → 16 maps of 5×5
- C5 - Convolutional layer (acts like FC)
  - 120 filters of 5×5 → each produces a 1×1 map → flattened to a 120-d vector
- F6 - fully connected
  - 84 units, then
- Output layer - 10 units with s/w for class probability.

- LeNet-5 is compact, fast, and works well on small greyscale digit images.

Q. 2) Consider a convolutional NN with two convolutional layers. Solve the following: considering input to be 10 × 10 × 1 applied with 3 × 3 × 5 kernels, calculate the dimension of the first convolution layer among with number of activations. Calculate the number of trainable parameters for the same.

→ given, input size = 10 × 10 × 1

kernel size = 3 × 3

no. of filters = 5

stride = 1

padding = 0

S1 - dim^n of the first convolution layer -

$H_{out} = \frac{H_{in} - K}{S} + 1$ $\quad W_{out} = \frac{W_{in} - K}{S} + 1$

$= \frac{10 - 3}{1} + 1$ $\quad = \frac{10 - 3}{1} + 1$

$= 8$ $\quad = 8$

∴ No. of activations = 8 × 8 × 5 = 320

S2 - each filter has

∴ K × K × C_{in} = 3 × 3 × 1

= 9 weights

∴ So per has, each filter also has 1 bias term

∴ So, per filter,

∴ 9 + 1 = 10 parameters

∴ for 5 filters

∴ 5 × 10 = 50 parameters

∴ Total trainable parameters = 50