|  |  |
| --- | --- |
| **Ex No: 4**  **Date: 28-08-2024** | **Handwritten digit recognition using CNN** |

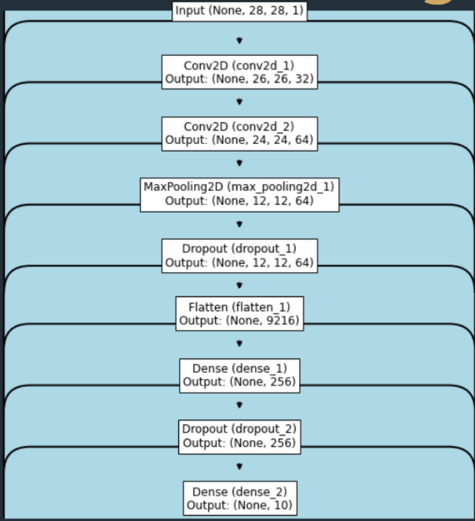
**Objective:**

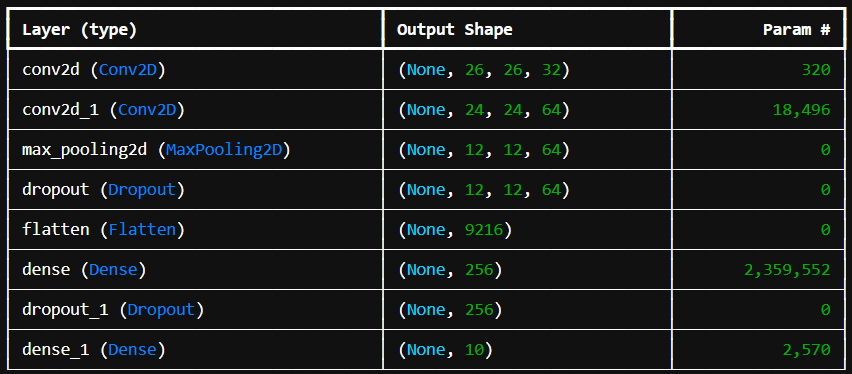
The main objective is to design and train a CNN for classifying handwritten digits from the MNIST dataset, using layers like convolutional layers, dropout, and optimizers. This helps in understanding how CNNs function in image classification tasks.

**Descriptions:**

This lab involves building a Convolutional Neural Network (CNN) for classifying handwritten digits from the MNIST dataset. The CNN architecture includes convolutional layers to extract essential features like edges and textures, and pooling layers to reduce dimensionality and computational cost. Dropout layers are used to prevent overfitting by randomly deactivating some neurons during training, enhancing the model's generalization. Flatten layers convert 2D feature maps into 1D vectors for fully connected layers, which perform high-level reasoning. The model is compiled with categorical cross-entropy as the loss function, Adadelta as the optimizer, and accuracy as the performance metric. The trained model predicts digit classes by outputting probability distributions for each input image.

**Model:**

****

****

**Building the parts of algorithm**

1. Initialize the Model: Start by creating a sequential model using Sequential() to stack layers in order.
2. Add Convolutional Layers: Add Conv2D layers with ReLU activation to extract features from the input images, followed by MaxPooling2D layers to downsample and reduce dimensionality.
3. Incorporate Dropout: Introduce Dropout layers to randomly deactivate neurons during training, helping to prevent overfitting.
4. Flatten and Connect: Use a Flatten layer to convert the 2D feature maps into a 1D vector, then add Dense layers for classification, ending with a softmax layer to output probabilities for each class.
5. Compile and Train: Compile the model with a suitable loss function, optimizer, and metrics, then train it on the MNIST dataset, adjusting weights to minimize loss and improve accuracy**.**

**GitHub Link:**

**https://github.com/amruthaa-m/DL-Lab1/tree/main/Unit-1/lab4**