| |
|---|
| Experiment No. 6 |
| Design and implement a CNN model for digit recognition application. |
| Date of Performance: |
| Date of Submission: |

**Aim:** Design and implement a CNN model for digit recognition application.

**Objective:** Ability to design and implement a CNN model that can accurately recognize handwritten digits (0–9) from image data.

**Theory:**

A **Convolutional Neural Network (CNN)** is a deep learning model particularly effective for image-based tasks such as classification, object detection, and segmentation. It leverages the concept of **convolution**, which captures spatial hierarchies and patterns in image data.

The CNN architecture is made up of multiple layers that help automatically extract relevant image features:

1. **Convolutional Layer:**
   Applies filters to input images to extract spatial features such as edges, textures, and shapes.
2. **Activation Function (ReLU):**
   Introduces non-linearity and ensures that the network can learn complex mappings. 3. **Pooling Layer (Max Pooling):**
   Reduces the spatial size of feature maps, minimizing computation while retaining essential features.
4. **Fully Connected (Dense) Layers:**
   Combine extracted features to perform classification tasks. 5. **Output Layer:**
   Uses a **Softmax** activation to predict probabilities for each class (digits 0–9).

## Working Principle:

The CNN model is trained on labeled image data. During training:

- The network extracts hierarchical features from the images.
- These features are fed into dense layers that classify the image into one of the digit categories (0–9).

- The model is optimized using backpropagation with a loss function such as **CrossEntropyLoss**.

## Applications of CNN:

1. **Digit Recognition:** Used in postal automation, bank check processing, and digital handwriting input systems.
2. **Image Classification:** Identifying objects or patterns within images.
3. **Medical Imaging:** Detecting anomalies or diseases from medical scans.
4. **Computer Vision Systems:** Used in autonomous vehicles, face recognition, and surveillance.

**Conclusion:**

The designed Convolutional Neural Network effectively recognized handwritten digits from the MNIST dataset with high accuracy. The hierarchical feature extraction capability of CNN layers enables robust learning of visual patterns, making CNNs highly efficient for image-based classification tasks. This experiment demonstrates the effectiveness of CNN architectures for digit recognition applications such as OCR systems, form digitization, and automated number reading.

**1. Summary of the Experiment**

The experiment involved designing and training a **Convolutional Neural Network (CNN)** to recognize handwritten digits from the **MNIST dataset**. Key observations:

- The CNN achieved **high accuracy**, demonstrating strong generalization on unseen test data.
- The network effectively learned **hierarchical visual features**, starting from low-level edges to high-level digit structures.
- The experiment highlights the **strength of CNNs in image-based classification tasks**.

---

**2. Critical Analysis**

**Merits:**

1. **Hierarchical Feature Learning:**
   - Convolutional layers automatically learn spatial hierarchies of features (edges → curves → shapes → digits).
   - Reduces the need for manual feature engineering compared to traditional machine learning methods.
2. **High Accuracy & Robustness:**

- CNNs are resilient to variations in handwriting styles and distortions in the MNIST dataset.
    - Pooling layers provide translation invariance, improving generalization.
3. **Wide Applicability:**
    - Optical Character Recognition (OCR) systems for automated digit reading.
    - Form digitization in administrative tasks.
    - License plate recognition or automated meter reading systems.

**Limitations / Challenges:**

1. **Computational Resources:**
    - Training deep CNNs can be resource-intensive, especially with large datasets or high-resolution images.
2. **Overfitting Risk:**
    - Despite high training accuracy, overfitting may occur if the network is too deep relative to dataset size.
    - Requires regularization techniques such as dropout or data augmentation.
3. **Interpretability:**
    - CNNs are often “black boxes”; understanding exactly which features lead to a particular classification is difficult.
4. **Dataset Dependence:**
    - Performance is excellent on MNIST, but may drop on more complex or noisy real-world datasets.