**COMPARISON OF CNN ARCHITECTURES ON DIFFERENT DATASETS**

**1. LeNet-5**

**Overview:**

LeNet-5 is a pioneering convolutional neural network designed for simple tasks like handwritten digit recognition. It uses fewer parameters and is suitable for small datasets.

**Performance:**

* **MNIST**: Excellent accuracy (∼99\sim99%∼99), given its simplicity and the dataset's low complexity.
* **FMNIST**: Moderate accuracy (∼88−91\sim88-91%∼88−91); struggles slightly with the more complex patterns of FMNIST.
* **CIFAR-10**: Poor accuracy (∼70−75\sim70-75%∼70−75); limited by its shallow architecture for RGB data.

**Strengths:**

1. **Simplicity**:
   * Easy to implement and requires low computational resources.
2. **Efficiency**:
   * Lightweight and ideal for small datasets.

**Weaknesses:**

1. **Shallow Architecture**:
   * Struggles with complex datasets like CIFAR-10.
2. **Limited Scalability**:
   * Not suitable for large-scale or high-dimensional data.

**2. Alex Net**

**Overview:**

Alex Net popularized deep learning by introducing stacked convolutional layers, ReLU activations, and dropout to reduce overfitting. It’s designed for large RGB datasets.

**Performance:**

* **MNIST**: High accuracy (∼99\sim99%∼99), but overkill for a simple dataset like MNIST.
* **FMNIST**: Good accuracy (∼90−93\sim90-93%∼90−93), performs well on grayscale fashion images.
* **CIFAR-10**: Moderate accuracy (∼85−88\sim85-88%∼85−88); performs well but is less efficient compared to modern architectures.

**Strengths:**

1. **Deeper Network**:
   * Captures complex patterns effectively.
2. **Regularization**:
   * Uses dropout to mitigate overfitting.

**Weaknesses:**

1. **Computationally Intensive**:
   * Requires more memory and power compared to simpler architectures.
2. **Outdated**:
   * Struggles to compete with more efficient and deeper models like ResNet or GoogLeNet.

**3. GoogLeNet (Inception)**

**Overview:**

GoogLeNet introduced the Inception module, which allows the network to capture multi-scale features efficiently using a mix of 1×11\times11×1, 3×33\times33×3, and 5×55\times55×5 convolutions.

**Performance:**

* **MNIST**: Excellent accuracy (∼99\sim99%∼99); performs well but is computational overkill.
* **FMNIST**: High accuracy (∼91−94\sim91-94%∼91−94); excels due to its multi-scale feature extraction.
* **CIFAR-10**: High accuracy (∼90−93\sim90-93%∼90−93); efficient for RGB data due to Inception modules.

**Strengths:**

1. **Multi-Scale Features**:
   * Extracts features of varying sizes effectively.
2. **Efficiency**:
   * Uses fewer parameters compared to AlexNet or VGG.

**Weaknesses:**

1. **Complexity**:
   * The Inception module adds implementation complexity.
2. **Overkill for Small Datasets**:
   * Excessive for simple datasets like MNIST.

**4. VGG**

**Overview:**

VGG relies on deep stacks of 3×33 \times 33×3 convolutions and max-pooling. It's computationally expensive but effective for feature extraction.

**Performance:**

* **MNIST**: High accuracy (∼99\sim99%∼99); overkill for such a simple dataset.
* **FMNIST**: High accuracy (∼90−93\sim90-93%∼90−93); well-suited for fashion images.
* **CIFAR-10**: Moderate accuracy (∼85−90\sim85-90%∼85−90); performs well but struggles with computational efficiency.

**Strengths:**

1. **Thorough Feature Extraction**:
   * Deep architecture captures detailed patterns.
2. **Simplicity**:
   * Easy to implement and extend.

**Weaknesses:**

1. **Computationally Expensive**:
   * Large number of parameters and slow training.
2. **Inefficiency**:
   * Outperformed by more modern architectures like ResNet.

**5. ResNet**

**Overview:**

ResNet introduced residual connections, enabling very deep networks by solving the vanishing gradient problem. It is highly effective for complex datasets.

**Performance:**

* **MNIST**: Excellent accuracy (∼99\sim99%∼99); depth is excessive for MNIST.
* **FMNIST**: High accuracy (∼92−95\sim92-95%∼92−95); performs well due to its deep residual connections.
* **CIFAR-10**: Excellent accuracy (∼90−94\sim90-94%∼90−94); state-of-the-art for CIFAR-10.

**Strengths:**

1. **Residual Connections**:
   * Allows very deep networks without degradation.
2. **Generalization**:
   * Performs well across a variety of datasets.

**Weaknesses:**

1. **Training Complexity**:
   * Requires careful tuning of hyperparameters.
2. **High Computational Cost**:
   * More expensive than simpler architectures like LeNet or AlexNet.

**6. Xception**

**Overview:**

Xception uses depthwise separable convolutions, making it computationally efficient while retaining high accuracy.

**Performance:**

* **MNIST**: Excellent accuracy (∼99\sim99%∼99); efficient and effective.
* **FMNIST**: High accuracy (∼93−95\sim93-95%∼93−95); handles grayscale data well with efficient feature extraction.
* **CIFAR-10**: High accuracy (∼90−93\sim90-93%∼90−93); efficient and competitive for RGB datasets.

**Strengths:**

1. **Efficiency**:
   * Depthwise separable convolutions reduce computational cost.
2. **Strong Feature Extraction**:
   * Competes with state-of-the-art models like ResNet.

**Weaknesses:**

1. **Implementation Complexity**:
   * Slightly harder to implement compared to simpler architectures.
2. **Less Versatile for Small Datasets**:
   * Not designed for extremely simple tasks like MNIST.

**7. SENet**

**Overview:**

SENet adds squeeze-and-excitation (SE) blocks, recalibrating channel-wise features for better representation and improved accuracy.

**Performance:**

* **MNIST**: Excellent accuracy (∼99\sim99%∼99); handles channel recalibration well even on simple datasets.
* **FMNIST**: High accuracy (∼92−94\sim92-94%∼92−94); enhances channel-wise learning for fashion images.
* **CIFAR-10**: High accuracy (∼90−93\sim90-93%∼90−93); excels in handling RGB images.

**Strengths:**

1. **Channel Recalibration**:
   * Focuses on important feature channels.
2. **Improved Generalization**:
   * Works well on diverse datasets.

**Weaknesses:**

1. **Increased Computational Cost**:
   * SE blocks add extra operations.
2. **Complexity**:
   * Slightly harder to implement compared to simpler architectures.

**Summary Table**

| **Architecture** | **MNIST (%)** | **FMNIST (%)** | **CIFAR-10 (%)** | **Strengths** | **Weaknesses** |
| --- | --- | --- | --- | --- | --- |
| **LeNet-5** | 99 | 88-91 | 70-75 | Lightweight, easy to train | Struggles with complex datasets |
| **AlexNet** | 99 | 90-93 | 85-88 | Deep learning pioneer, strong feature extraction | Outdated, computationally expensive |
| **GoogLeNet** | 99 | 91-94 | 90-93 | Multi-scale features, efficient | Complex implementation |
| **VGG** | 99 | 90-93 | 85-90 | Simplicity, effective on small datasets | Computationally expensive |
| **ResNet** | 99 | 92-95 | 90-94 | Deep networks, state-of-the-art | High computational cost |
| **Xception** | 99 | 93-95 | 90-93 | Efficient, strong feature extraction | Harder to implement |
| **SENet** | 99 | 92-94 | 90-93 | Channel recalibration, versatile | Higher computational cost |

**Conclusion**

* For **simple datasets** like **MNIST**, **LeNet-5** or **AlexNet** is sufficient.
* For **FMNIST**, **GoogLeNet**, **ResNet**, or **Xception** performs best due to their deeper architectures and ability to handle complex patterns.

For **CIFAR-10**, **ResNet** or **Xception** is ideal for achieving high accuracy efficiently, while **SENet** adds value through channel recalibration

.