**HW4 Report**

**Comparison and Conclusions**

**1. Methodology**

This study compares the performance of a custom Convolutional Neural Network (CNN) and a pre-trained ResNet18 model using transfer learning on the Fashion-MNIST dataset.

* **Dataset:** Fashion-MNIST, containing 70,000 grayscale images across 10 categories.
* **Preprocessing:**
  + Images resized to 224x224 pixels.
  + Converted to RGB using grayscale-to-RGB transformation.
  + Normalized using ImageNet mean and standard deviation.
  + Augmentation techniques: random horizontal flips and rotations.
* **Model Implementation:**
  + **Custom CNN:** Designed with convolutional layers, max-pooling, dropout, and fully connected layers, trained from scratch.
  + **ResNet18:** A pre-trained model with frozen convolutional layers initially, followed by fine-tuning.

**2. Results and Key Comparisons**

|  |  |  |
| --- | --- | --- |
| **Metric** | **Custom CNN** | **ResNet18 (Transfer Learning)** |
| **Accuracy (%)** | 91.84 | 93.52 |
| **Precision** | 0.919 | 0.937 |
| **Recall** | 0.916 | 0.935 |
| **F1-Score** | 0.919 | 0.935 |
| **Training Time (Epochs)** | ~10 epochs | ~5 epochs (FC only), ~10 epochs (fine-tuning) |

* **Accuracy:** ResNet18 outperformed the custom CNN by achieving a higher test accuracy of 93.52%.
* **Training Time:** ResNet18’s transfer learning approach required fewer epochs to converge during the initial training phase.
* **Generalization:** ResNet18 exhibited better precision, recall, and F1-scores, especially in challenging categories.

**3. Visualizations**

**a. Training and Validation Metrics**

A line graph with blue and orange lines

Description automatically generatedThe **CNN** model demonstrated a rapid decline in training and validation losses, indicating faster convergence initially.

A graph with blue and orange lines

Description automatically generatedThe **ResNet18** model exhibited a slower initial decrease in losses due to frozen layers, but after fine-tuning, it achieved better generalization with smoother validation loss

**b. Confusion Matrix**

* **Custom CNN:** Misclassifications were more prominent in categories such as "Shirt" vs. "T-shirt/top." A graph of a number of blue squares

  Description automatically generated with medium confidence
* **ResNet18:** Demonstrated improved performance in 9 out of 10 categories, with fewer misclassifications overall.

A graph of a number

Description automatically generated with medium confidence

The confusion matrices for both the CNN and ResNet18 models reveal strong performance on the Fashion-MNIST dataset. However, the ResNet18 model demonstrated superior classification accuracy, outperforming the CNN in 9 out of 10 classes and showing fewer overall misclassifications. The only exception was the "Coat" class, where the CNN achieved slightly higher accuracy. Nevertheless, the ResNet18 model exhibited more consistent and balanced performance across all other categories, highlighting its enhanced generalization capabilities.

**4. Conclusions on Transfer Learning Effectiveness**

**Advantages of Transfer Learning:**

* **Efficiency:** Reduced training time by leveraging pre-trained weights for feature extraction.
* **Improved Accuracy:** Higher performance metrics compared to training a model from scratch.
* **Robust Generalization:** Better at distinguishing between visually similar classes.

**Limitations:**

* Requires careful adjustment of learning rates during fine-tuning.
* Computational demands increase during fine-tuning compared to training only the fully connected layers.

**Scenarios Where Transfer Learning Excels:**

* **Small Datasets:** When training data is limited, transfer learning captures general patterns effectively.
* **Time Constraints:** Faster results due to pre-trained feature extraction.
* **Complex Tasks:** Datasets with intricate patterns benefit from pre-trained models like ResNet.

**Summary of Pre-trained ResNet Impact:**

* Enhanced accuracy, precision, recall, and F1-scores.
* Allowed faster convergence during the initial training phase.
* Achieved better generalization across challenging categories.

**Challenges:**

* Initial training process lacked proper storage for loss metrics, requiring code adjustments and re-training.
* Fine-tuning required precise hyperparameter tuning to prevent overfitting.