Saudi Digital Academy

AI Bootcamp

Convolutional Neural Networks and Transfer Learning

Team Members:

1. Ahmed Alqarni
2. Dana Saeed
3. Hanan Alnbhani
4. Raghad Alkhudair

Project Number: 2 Group Number: 5

Table of Contents:

1.Introduction

Convolutional Neural Networks (CNNs) have revolutionized image classification tasks, providing state-of-the-art performance in computer vision. In this project, we implement and evaluate CNN models, including custom architectures and pretrained models like VGG16 and ResNet50, on the CIFAR-10 dataset. This report details the methodology, preprocessing, training, evaluation, and key insights.

2. Chosen CNN Architecture

Two CNN architectures were explored:

- **Custom CNN Model**: A sequential CNN built from scratch.
- **Pretrained CNN Models (VGG16 & ResNet50)**: Transfer learning using pre-trained networks.
-

CNN Training Process

The CNN training process involves:

- Extracting low-level features (edges, corners) in early layers.
- Detecting high-level patterns (shapes, objects) in deeper layers.
- Using backpropagation to optimize weights and improve accuracy.
- Classifying the image into one of the predefined categories

Custom CNN Model

The custom model is designed with the following components:

- **Convolutional Layers**: Feature extraction using Conv2D with ReLU activation.
- **Pooling Layers**: MaxPooling2D to downsample feature maps.
- **Batch Normalization**: Stabilizes training and speeds up convergence.
- **Dropout Layers**: Reduces overfitting.
- **Fully Connected Layers**: Dense layers with softmax activation for classification.

Pretrained CNN Models

- **VGG16**: A 16-layer deep model known for its strong feature extraction.
- **ResNet50**: A 50-layer deep residual network that mitigates vanishing gradient issues using skip connections.

Transfer Learning Training Process

The Transfer learning training process involves:

- **Selecting a Pre-trained Model:** Choosing a CNN model trained on a large dataset like ResNet50, VGG16, ImageNet
- **Freezing Early Layers:** Keeping initial layers unchanged as they contain general image features.
- **Fine-tuning Final Layers:** Training only the last few layers to adapt to the new dataset.
- **Replacing the Fully Connected Layer:** Customizing it to match the number of target classes.

3. Data Preprocessing

Dataset

The CIFAR-10 dataset consists of 60,000 color images (32x32 pixels) across 10 categories:
['Airplane', 'Automobile', 'Bird', 'Cat', 'Deer', 'Dog', 'Frog', 'Horse', 'Ship', 'Truck']

Preprocessing Steps

- **Normalization**: Pixel values scaled to the [0,1] range.
- **One-Hot Encoding**: Converts labels into categorical format.
- **Data Augmentation**: Used to increase dataset diversity:
    - Random rotations
    - Width and height shifts
    - Horizontal flips

4. Training Process

Hyperparameters

Parameter	Value
Optimizer	Adjusts model weights to minimize loss.
Learning Rate	Controls step size in weight updates
Batch Size	Number of samples processed at once
Epochs	Full passes through the training dataset
Loss Function	Measures error between predictions and actual values.

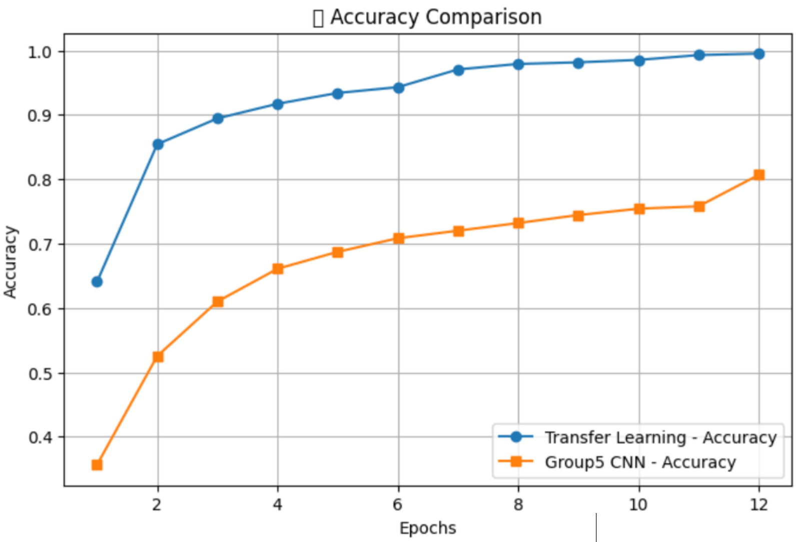
Callbacks Used

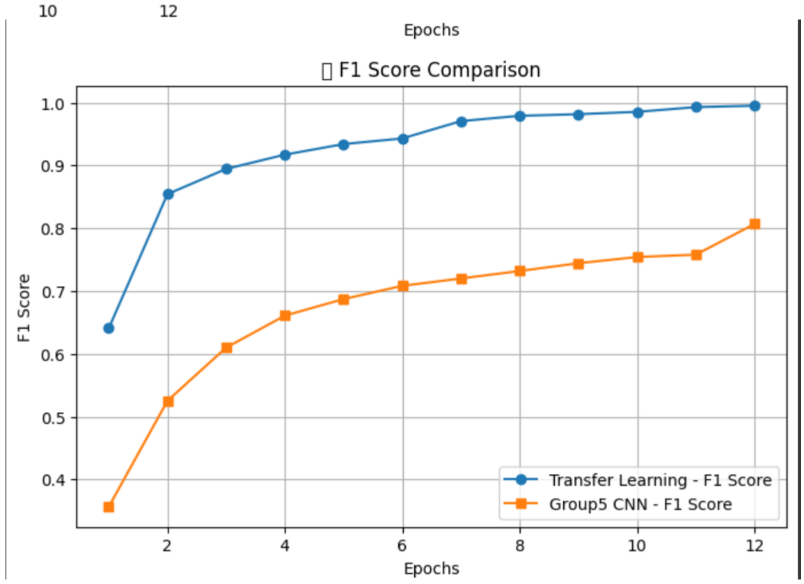
- **Early Stopping**: Stops training if validation loss doesn't improve.
- **ReduceLROnPlateau**: Lowers learning rate when validation loss plateaus.

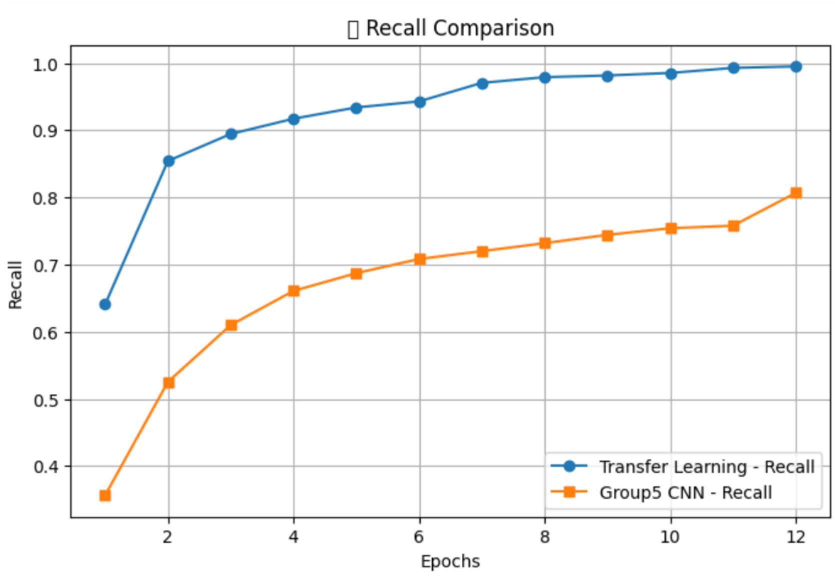
5. Results and Performance Analysis

Evaluation Metrics

- **Accuracy**
- **Loss**
- **Confusion Matrix**







Comparison of Models

Model	Test Accuracy
Custom CNN	86.60%
Custom CNN2	91.58%
VGG16	92.21%
ResNet50	91.31%

6. Best Model and Justification

Best Model: VGG16

After testing both Custom CNN models and Transfer Learning, we found that VGG16 delivered the best performance, achieving the highest test accuracy (92.21%). While our Custom CNN models showed promising results, they lacked the deep feature extraction capabilities of VGG16.

Although ResNet50 is a more advanced model with residual connections that help prevent vanishing gradients, it did not outperform VGG16 in our case, achieving a slightly lower accuracy (91.31%). This suggests that for our specific dataset, VGG16's structured architecture provided better feature extraction and classification performance, while also being easier to fine-tune compared to ResNet50's deeper and more complex network.

Additionally, VGG16 has been widely used in image classification tasks and is known for its reliability and consistency in extracting robust features.

7. Insights and Key Learnings

Custom vs. Pretrained Models

- Custom CNN models showed good performance (86.60% and 91.58% accuracy), but they struggled to match the feature extraction capabilities of pretrained models.
- Transfer learning significantly improved accuracy by leveraging pretrained knowledge, reducing the need for extensive training from scratch.

VGG16 vs. ResNet50

- While ResNet50 is deeper and incorporates skip connections to handle vanishing gradients, it did not outperform VGG16 in our case.
- VGG16 achieved the highest accuracy (92.21%), making it the most effective model for our dataset.

Importance of Data Preprocessing

- Normalization, one-hot encoding, and data augmentation played a crucial role in improving generalization and preventing overfitting.

- Data augmentation helped increase dataset diversity, leading to better model robustness.

Effectiveness of Training Techniques

- Early Stopping helped prevent unnecessary overfitting and reduced computation time.
- ReduceLROnPlateau improved optimization by adjusting the learning rate dynamically when performance plateaued.

8. Conclusion

Through this study, we demonstrated the effectiveness of Convolutional Neural Networks (CNNs) and Transfer Learning in image classification tasks. Our experiments showed that while custom CNN architectures can perform well, pretrained models like VGG16 offer superior feature extraction, leading to higher accuracy and better generalization.

Among the tested models, VGG16 emerged as the best-performing architecture (92.21% accuracy), outperforming ResNet50 (91.31%) and custom models. The structured nature of VGG16 made it easier to fine-tune while maintaining high reliability and consistency.

In future work, further improvements could be explored through hyperparameter tuning, more advanced augmentation techniques, or testing newer architectures like EfficientNet or Vision Transformers (ViTs) for potentially better results.