KLE Technological University | Creating Value, Leveraging Knowledge

IMAGE CLASSIFICATION USING CNN AND TRANSFER LEARNING

Team Members:

NAME	SRN	DIVISION
Gouri N Banapurmath	01FE23BCI079	G
Tanuja D Ratan	01FE23BCI076	G
Swajeet Chikhali	01FE23BCI070	F
Parth H	01FE23BCS201	E

PROBLEM STATEMENT

- Manual image classification is time-consuming and inefficient for large, multi-class datasets.
- Objective: Build a CNN-based model to automatically classify images into 19 predefined categories.
- Compare the performance of a custom CNN with a VGG16 transfer learning model.

DATASET DESCRIPTION

- Includes 19 categories (e.g.,bed, bottles, car, cellphone, chair, college, furniture, indoor, monitor, nameboard, office, railwaystations, road, school, signboard, sofas, toys, trafficsignal, tv.).
- Each category has multiple images for training and testing.
- Total images: 4,640 for training and 1,147 for testing.
- Images are resized to 224x224 and normalized for better model performance.

CUSTOM CNN ARCHITECTURE

Convolutional Layers:

- Conv1: 3x3, 64 filters, BatchNorm → ReLU → MaxPooling
- Conv2: 3x3, 128 filters, BatchNorm → ReLU → MaxPooling
- Conv3: 3x3, 256 filters, ReLU → MaxPooling
- Conv4: 3x3, 512 filters, ReLU → MaxPooling

Fully Connected Layers:

- Flatten → FC1: 512 units + ReLU + Dropout (0.5)
- FC2: Output layer for 19 categories

Training Setup:

- Activation: ReLU
- Loss Function: CrossEntropyLoss
- Optimizer: Adam (learning rate = 0.001)
- Epochs: 10 (trained on GPU)

Training & Performance

Training Configuration:

- Epochs: 10
- Batch Size: 32
- Learning Rate: 0.0001
- Optimizer: Adam
- Loss Function: Cross Entropy Loss
- Regularization: Dropout (0.5), Batch Normalization
- Device: GPU (if available)

Performance:

- Training Accuracy (Final Epoch): 98.03%
- Test Accuracy: 73.03%

Remarks:

- Achieved high training accuracy but showed signs of overfitting.

VGG16 Transfer Learning

Base Model: Pretrained VGG16 from torchvision.models
Modifications:

- Feature extractor layers frozen.
- Final classifier layer replaced with Linear(4096 → 19).
- Training Setup:
- Loss: Cross Entropy Loss
- Batch Size: 32 | Epochs: 10
- Device: GPU (if available)
- Optimizer: Adam (lr = 0.0001)

Result:

- Test Accuracy: 90.19%
- Improved generalization and reduced overfitting compared to custom CNN
- Slower training time due to large model size

Comparison – CNN vs VGG16

Feature	Custom CNN	VGG16(Transfer Learning)
Test Accuracy	73.03%	90.19%
Training Time	Faster	Slightly Longer
Overfitting	Moderate	Minimal
Model Complexity	Lightweight	High(pretrained)
Adaptability	Built from scratch	Uses generalized features

Advantages & Limitations

Advantages:

- Custom CNN achieved high training accuracy with relatively few layers.
- Transfer learning with VGG16 improved test accuracy by ~17%.
- Dropout and batch normalization improved performance stability.

Limitations:

- Custom CNN suffered from overfitting due to small dataset size.
- No early stopping or learning rate

Conclusion & Future Work

Conclusion:

- Built and trained two models: a Custom CNN and a VGG16-based transfer learning model.
- VGG16 significantly outperformed Custom CNN in accuracy and generalization (90.19% vs 73.03%).
- Custom CNN was simpler and faster but prone to overfitting.

Future Work:

- Add early stopping and learning rate schedulers.
- Apply K-Fold Cross Validation for robust evaluation.
- Expand dataset for better generalization.
- Visualize predictions using Grad-CAM for interpretability.

Thank you