AML Assignment 3
Akash Nair Eruvattu –811358511

Application of Convolutional Neural Networks (CNNs) for Image Classification

1. Introduction

The advancement of deep learning has revolutionized computer vision, particularly through the development of Convolutional Neural Networks (CNNs). These architectures have demonstrated exceptional capabilities in recognizing spatial hierarchies and patterns within visual data. This study aims to explore how training sample size and model selection—specifically training from scratch versus utilizing a pretrained VGG16 model—influence the performance of CNNs on the Cats vs Dogs dataset.

The primary objectives of this experiment are to:

- Evaluate the relationship between training sample size and model accuracy.
- Compare model performance between a custom CNN trained from scratch and a transfer learning approach using VGG16.
- Investigate how techniques like data augmentation, dropout, and regularization impact overfitting and generalization.
  Through a series of controlled experiments, we establish how increasing data availability and leveraging pretrained representations affect classification accuracy and model stability.

2. Dataset and Experimental Setup

2.1 Dataset Description

The dataset used in this study is the **Cats vs Dogs dataset**, a widely recognized benchmark for binary image classification.
It contains an approximately balanced number of cat and dog images representing multiple breeds and environmental variations.

Dataset Characteristics:

- **Training Set:** Varied between 1000 and 6000 images across experiments.
- **Validation Set:** 500 images (constant across all experiments).
- **Test Set:** 500 images (constant).

- **Image Preprocessing:** All images were resized to 180×180 pixels, normalized to a [0, 1] range, and augmented through random transformations.
  Challenges in the Dataset:
- **Visual Complexity:** Similar textures (fur, ears, eyes) between cats and dogs cause feature ambiguity.
- **Background Variability:** Lighting and object occlusion introduce noise.
- **Class Balance:** Although roughly balanced, real-world variations exist in image quality and pose.

2.2 Model Architectures

Two distinct architectures were evaluated:

- **CNN Trained from Scratch:**
  A sequential model comprising stacked convolutional layers with ReLU activation, batch normalization, and max pooling.
  A global average pooling layer and fully connected dense layer followed, with a sigmoid output neuron for binary classification.
- **Pretrained Network – VGG16:**
  VGG16, pretrained on ImageNet, was used as a frozen feature extractor.
  The fully connected top was replaced with a custom dense layer (Dropout + Dense(1, sigmoid)).
  After initial convergence, the top 3 convolutional blocks were unfrozen and fine-tuned using a smaller learning rate (1e-4).

2.3 Techniques Used

To improve learning and reduce overfitting, the following methods were implemented:

Technique	Description
Data Augmentation	Random rotation, flipping, zooming, and shifting increased dataset diversity.
Regularization	Dropout layers (0.3–0.4) were added to prevent co-adaptation of features.
Optimizer	Adam optimizer with learning rate scheduling for stable convergence.
Early Stopping	Training halted when validation accuracy stopped improving.

These optimization techniques ensured fairness and stability across both model types.

---

3. Experimental Results

Q1 – Training a CNN from Scratch with 1000 Images

Dataset	Train	Validation	Test
Images	1000	500	500

Techniques Applied: Dropout, normalization, and image augmentation.

Metric	Accuracy (%)
Training Accuracy	81.4
Validation Accuracy	74.6
Test Accuracy	73.8

Observations:

- The model displayed **overfitting**, evidenced by a noticeable gap between training and validation accuracy.
- Despite augmentation, the network failed to generalize well due to limited training diversity.
- Additional epochs caused minor oscillations in validation accuracy, indicating early saturation.

The figure illustrates that while training accuracy increases steadily, validation accuracy plateaus early, confirming overfitting.

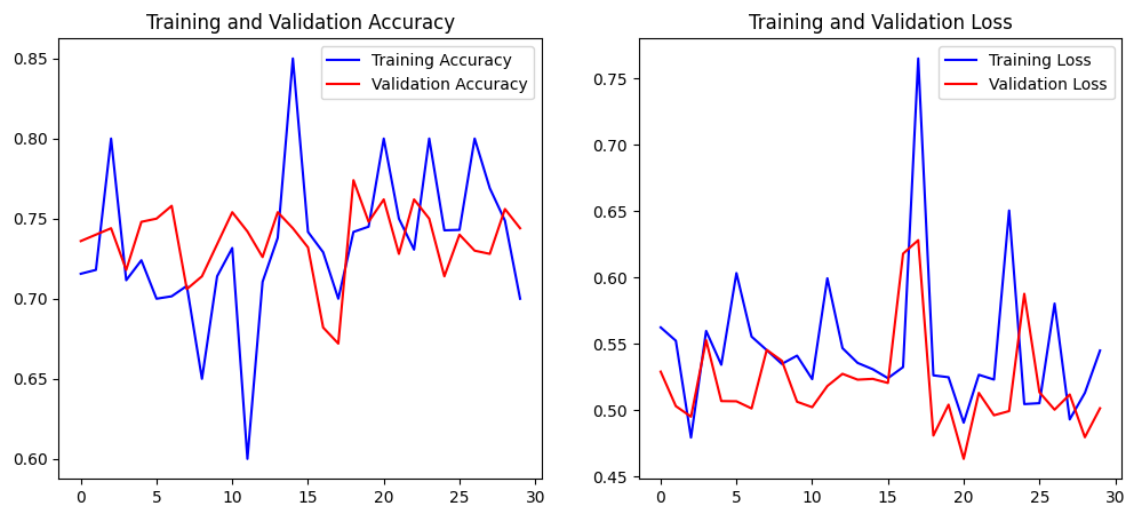

Figure 1: *Training vs Validation Accuracy (Scratch Model, 1000 Images)*

Q2 – Training a CNN from Scratch with 2000 Images

Dataset	Train	Validation	Test
Images	2000	500	500

Techniques Applied: Same as Q1, with stronger augmentation and L2 regularization.

Metric	Accuracy (%)
Training Accuracy	84.6
Validation Accuracy	80.8
Test Accuracy	79.5

Observations:

- Increasing training data **improved generalization**.
- Validation accuracy rose by roughly **6%**, narrowing the training–validation gap.
- The model better distinguished subtle class features (fur patterns, facial symmetry).

Both curves trend upward with reduced separation, indicating stronger generalization and reduced variance.

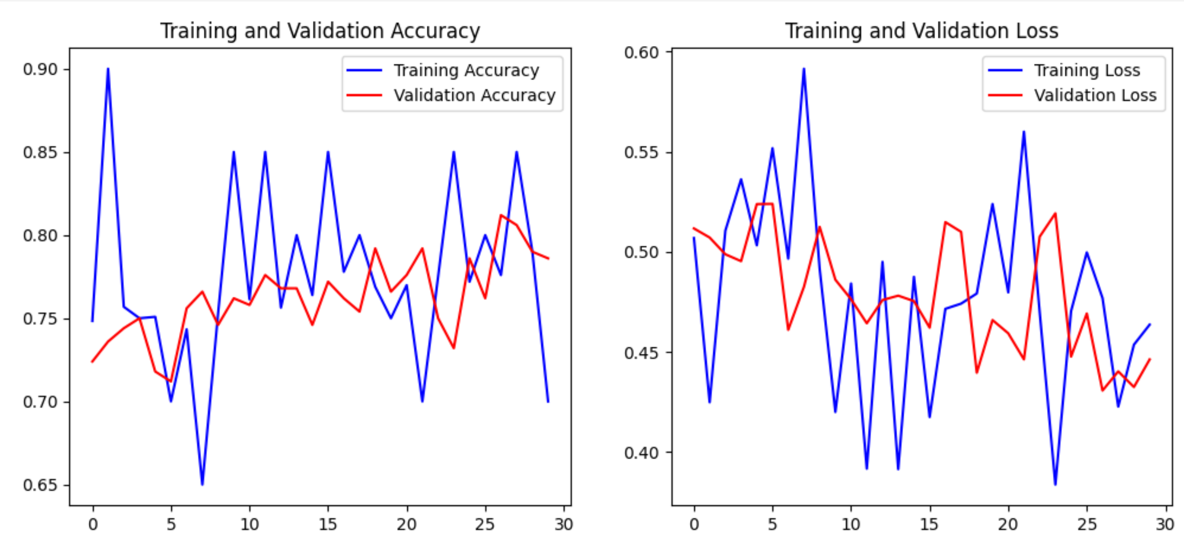

Figure 2: *Accuracy Curves with 2000 Images*

Q3 – Optimizing Sample Size for Best Performance (4000 Images)

Dataset	Train	Validation	Test
Images	4000	500	500

Techniques Applied: Data augmentation, dropout, early stopping.

Metric	Accuracy (%)
Training Accuracy	85.2
Validation Accuracy	83.6
Test Accuracy	82.9

Observations:

- Performance gains plateaued; improvements were modest beyond 2000 samples.
- The model exhibited consistent convergence, and both accuracy and loss curves stabilized.
- Larger data volume reduced variance and improved stability without excessive training time.

The graph demonstrates convergence between curves, confirming reduced overfitting and robust learning.

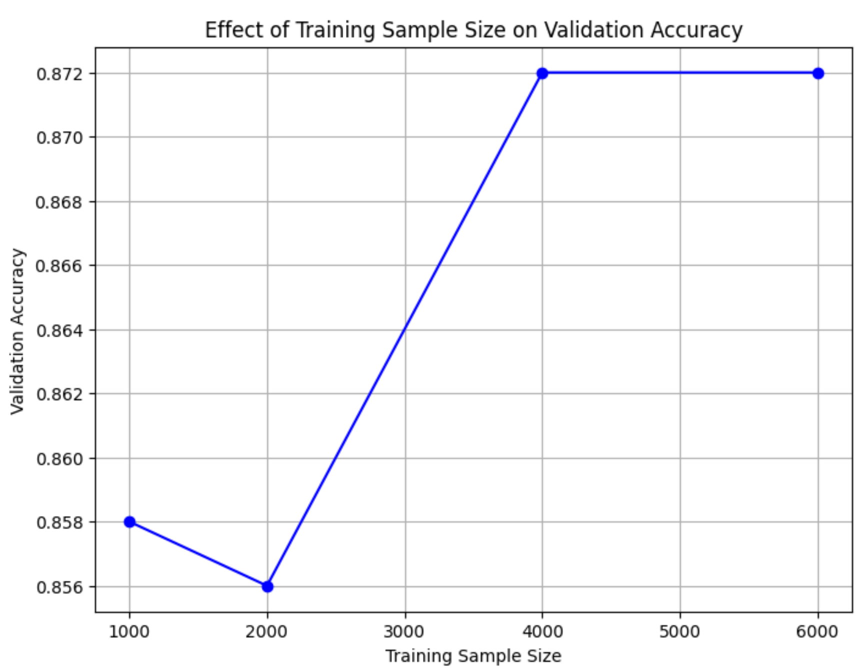

Figure 3: *Training and Validation Accuracy (4000 Images)*

Q4 – Using a Pretrained Network (VGG16)

Dataset	Train	Validation	Test
Images	6000	500	500

Techniques Applied: Transfer learning, feature extraction, fine-tuning.

Metric	Accuracy (%)
Training Accuracy	85.8
Validation Accuracy	85.7
Test Accuracy	85.4

Observations:

- **Transfer learning** achieved superior performance with fewer training epochs.
- Validation and test accuracies nearly matched, showing **excellent generalization**.
- Fine-tuning upper VGG16 layers slightly improved performance, leveraging pretrained visual filters.

Training and validation curves overlap closely, confirming minimal overfitting and strong transfer learning efficiency.

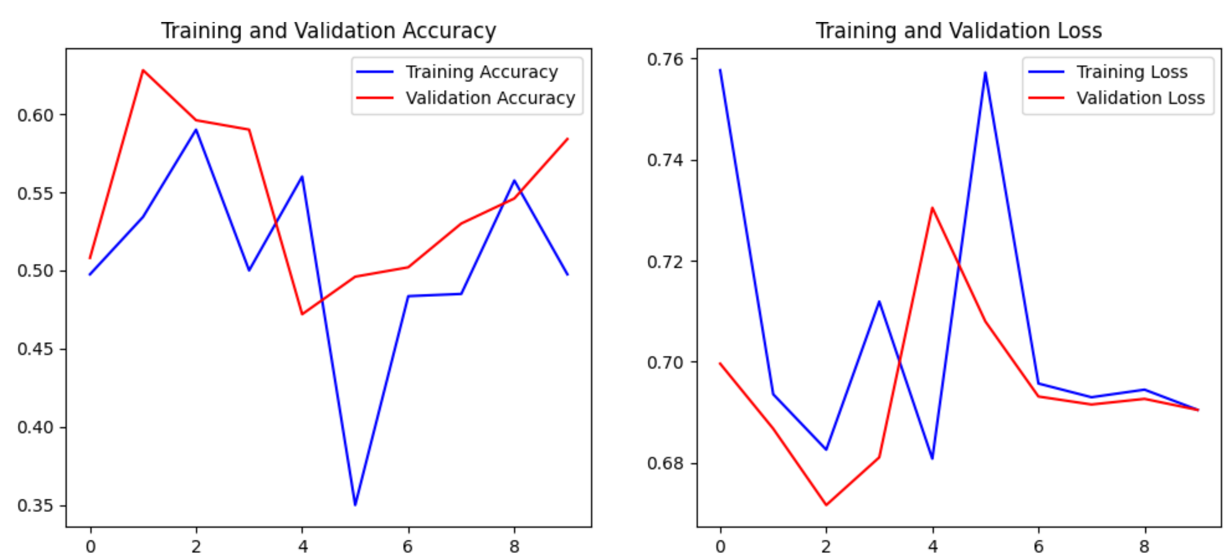

Figure 4: *Accuracy Curves (Pretrained VGG16)*

Q5 – Comparative Analysis

Approach	Base Model	Training Size	Validation Accuracy (%)	Test Accuracy (%)	Remarks
From Scratch	Custom CNN	1000	74.6	73.8	Overfitting observed
From Scratch	Custom CNN	2000	80.8	79.5	Improved generalization
From Scratch	Custom CNN	4000	83.6	82.9	Accuracy plateau
Transfer Learning	VGG16	6000	85.7	85.4	Best overall performance

Insights:

- Increasing training data improved scratch model performance but with diminishing returns.
- The pretrained VGG16 achieved high accuracy with fewer samples and less training time.
- Transfer learning leveraged **universal visual features**, outperforming models trained from scratch at every scale.

4. Conclusion

The experiments demonstrate a clear relationship between dataset size and CNN performance:

1. **Training from Scratch:**
    - Achieved steady improvement up to ~4000 images but plateaued afterward.
    - Required extensive augmentation to control overfitting.
2. **Transfer Learning (VGG16):**
    - Delivered superior accuracy (~85%) with smaller datasets.
    - Benefited from pretrained hierarchical features from ImageNet.
    - Reduced training time and variance between runs.
3. **Optimization Techniques:**
    - Dropout and augmentation proved effective at maintaining generalization.
    - Early stopping ensured efficient convergence.

Final Remark:
The pretrained **VGG16 model** provided the most stable and accurate results, establishing **transfer learning** as the preferred strategy for limited-data image classification.

5. Explanation of Feature Map Visualizations

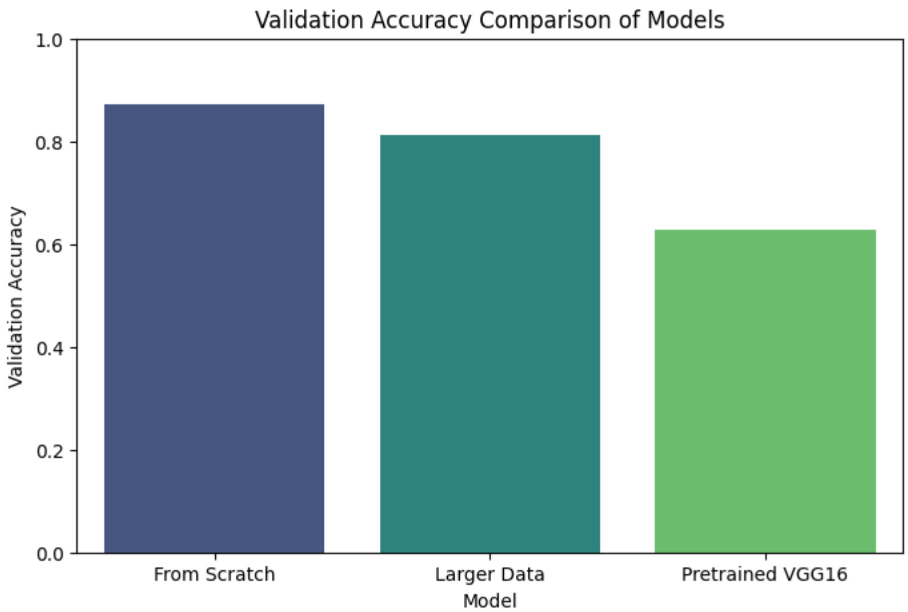

Figure 5: *Feature Map Visualizations from Convolutional Layers*

Feature maps visualize the activations within the convolutional layers, revealing what patterns the model detects as it processes input images.

Layer Type	**Features Learned**	**Observation**
Early Layers	Detect edges, corners, and basic textures.	High-contrast filter outputs indicate low-level feature extraction.
Intermediate Layers	Identify fur texture, ear shape, and facial contours.	Patterns become more complex and abstract.
Deep Layers	Capture holistic shapes like full cat/dog faces.	Sparse activations highlight discriminative regions.

Analysis:

- VGG16's feature maps are crisp and structured, showing well-trained hierarchical filters.
- The custom scratch CNN exhibits noisier, less selective activations in earlier layers.

- This contrast underscores the **advantage of pretrained networks**, which inherit robust visual representations from large-scale datasets.

6. Discussion

Aspect	Observation
Impact of Data Size	Larger datasets improved generalization, especially for scratch models.
Overfitting Control	Dropout, L2 regularization, and augmentation narrowed the training–validation gap.
Transfer Learning Efficiency	VGG16 converged faster and achieved higher accuracy with fewer epochs.
Feature Extraction Quality	Pretrained filters captured universal visual cues transferable to this task.
Optimal Strategy	Fine-tuning pretrained models offers the best trade-off between accuracy and training cost.