Very Deep Convolutional Networks for Large-Scale Image Recognition

Simonyan and Zisserman (2015), Reimplemented by Amit Yelin and Iking Lopez

1. Introduction

The VGG family of deep convolutional networks, introduced by Simonyan and Zisserman (2015), has been instrumental in showcasing the critical role of network depth in achieving state-of-the-art performance on large-scale image classification tasks, such as ImageNet (Deng et al., 2009). These architectures utilize small 3×3 convolutional filters, systematically stacked in increasing depths, to effectively capture spatial hierarchies and fine-grained image features. In this project, we reimplement the VGG11 and VGG16 architectures on the CIFAR-10 dataset (Krizhevsky, 2009), which comprises 60,000 32×32 images spanning 10 classes, samples of which are shown in Figure 1A. CIFAR-10 was selected as a benchmark dataset due to its manageable size and significantly reduced computational overhead compared to ImageNet, allowing for faster experimentation and model evaluation while maintaining a challenging classification task.

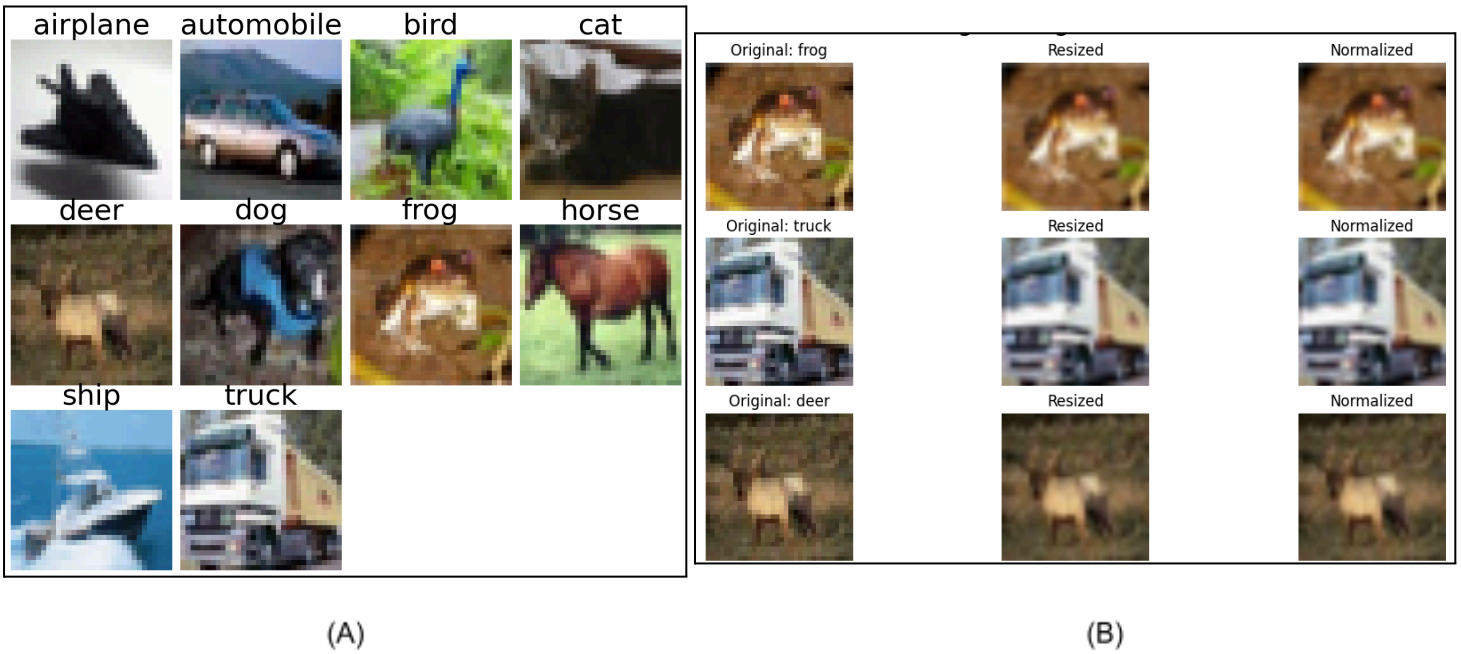

Figure 1: (A) Sample images from 10 classes of CIFAR-10. (B) Sample resizing and normalization preprocessing steps explored during the experiment

Crucially, as our dataset is different from that of the original paper, which used ImageNet, our research objective was simply to demonstrate that **classification accuracy improves with model depth,** albeit on a different dataset. To this end, we adjusted our data by resizing CIFAR-10 images to 224×224 and normalizing them to align with the VGG input requirements (Figure 1). Unlike the original paper, which primarily presented results in tabular form, this work importantly enhances model interpretability by incorporating comprehensive visualization components to analyze performance and decision-making processes.

2. Methods

2.1 **Model Architectures**

The architectures for VGG11 and VGG16 were implemented in alignment with the original paper's design, incorporating modifications to tailor them for the CIFAR-10 dataset. Both architectures feature a sequential arrangement of convolutional and pooling layers, designed to efficiently extract hierarchical features from the input images. These models use 3×3 convolutional filters throughout the network, enabling precise feature extraction while maintaining computational efficiency. VGG16, as an extension of VGG11, includes additional convolutional layers in certain blocks, allowing it to capture more intricate patterns and achieve improved performance compared to the shallower VGG11.

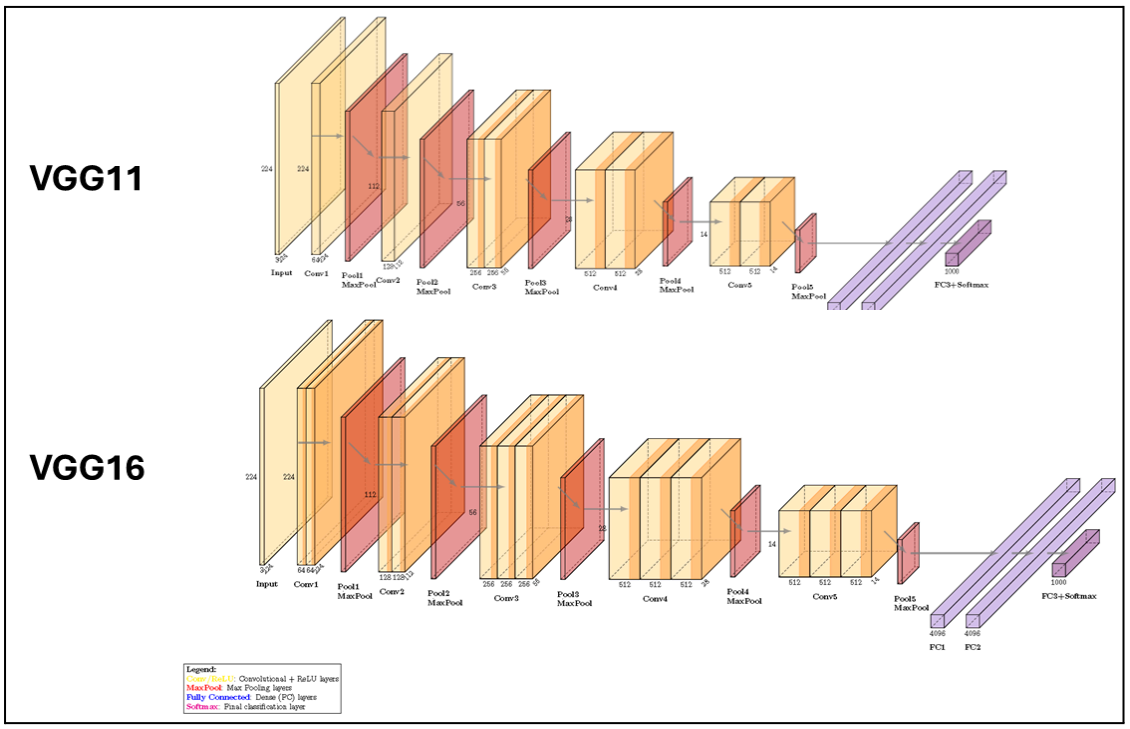

Figure 2: Our visualization of VGG-11 (top) and VGG-16 (bottom) architectures, featuring convolutional layers with ReLU, max-pooling for down-sampling, and fully connected layers for classification. VGG-16 includes more convolutional layers for enhanced feature extraction

In both models, the convolutional layers are grouped into blocks, with each block containing one or more convolutional layers followed by a max-pooling layer for downsampling. VGG11 contains fewer convolutional layers per block, making it more lightweight and faster to train but potentially less capable of capturing complex features. In contrast, VGG16 introduces extra convolutional layers in the later blocks, increasing the network's depth and enhancing its ability to learn detailed hierarchical features. For instance, while VGG11 has one convolutional layer in some blocks, VGG16 has two or three layers in corresponding blocks, making it more computationally intensive but also more accurate.

Max-pooling layers are strategically placed after each block in both architectures to progressively reduce the spatial dimensions of feature maps. Each pooling operation reduces the dimensions by a factor of 2, facilitating the compact representation of features while preserving critical spatial information. The difference in the number

of convolutional layers between VGG11 and VGG16 does not affect the placement of pooling layers, as both models adhere to a consistent structure for spatial downsampling.

At the end of the convolutional and pooling stages, three fully connected layers map the extracted features into class predictions. For CIFAR-10, the final fully connected layer is adjusted to produce an output of 10 neurons, corresponding to the dataset's 10 classes. Both models employ Rectified Linear Unit (ReLU) activations after each convolutional and fully connected layer to introduce non-linearity, enabling the network to learn complex patterns. Additionally, dropout with a rate of 0.5 is incorporated in the fully connected layers to reduce overfitting and improve generalization.

Despite their differences, both architectures are designed with parameter efficiency in mind, relying on small convolutional filters and systematic layer arrangements to balance representational power with computational manageability. These modifications optimize the models for CIFAR-10, which features smaller image dimensions (32×32 pixels) and fewer classes compared to the original ImageNet dataset. A visualization of the architectures for both VGG11 and VGG16 is shown in Figure 2, highlighting the structural differences between the two models and illustrating the hierarchical flow of feature maps through convolutional, pooling, and fully connected layers. This design ensures that both models effectively capture low-level and high-level features, with VGG16 offering greater representational power at the cost of increased computational complexity.

2.2 **Training and Evaluation**

The training and evaluation process involved several carefully designed steps to optimize the performance of VGG11 and VGG16 on the CIFAR-10 dataset. During preprocessing, the images were resized to 224×224 pixels and normalized using the CIFAR-10 mean and standard deviation to ensure compatibility with the model architecture and to facilitate effective learning. We also divided the data in accordance with the following train/validation/test split (see Figure 3A).

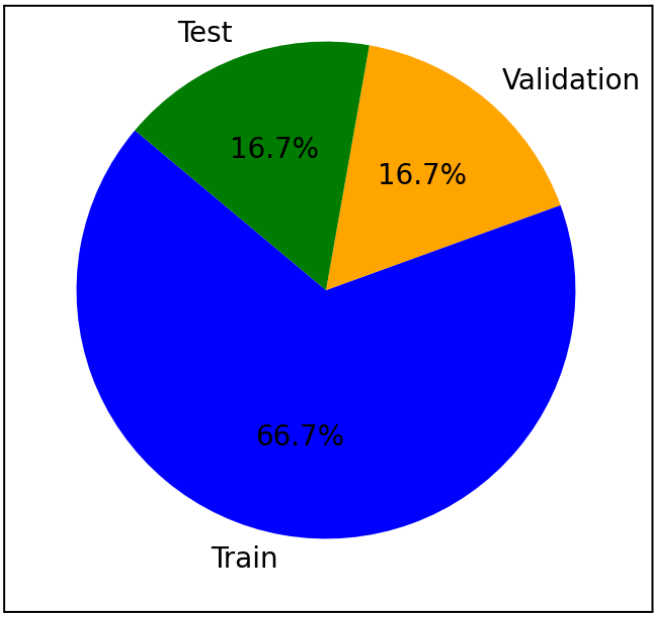

Figure 3: Train/validation/test proportions we chose for VGG model development on CIFAR-10

The Stochastic Gradient Descent (SGD) optimizer with a momentum of 0.9 was employed to update the model weights, enabling efficient convergence. An initial learning rate of 0.01 was used and dynamically reduced upon validation loss plateaus to fine-tune the training process. Regularization techniques, including an L2 weight decay of 5×10^{-4} and a dropout rate of 0.5 in fully connected layers, were applied to prevent overfitting and enhance generalization. Training was conducted with a batch size of 64, utilizing early stopping with a patience of five epochs to halt training if no improvement in validation performance was observed. Each model was trained independently, and due to differences in their architectures and convergence rates, VGG11 and VGG16 required varying numbers of epochs to reach optimal performance (see Figure 4). After training, both models were evaluated on the test set to measure their accuracy and overall performance.

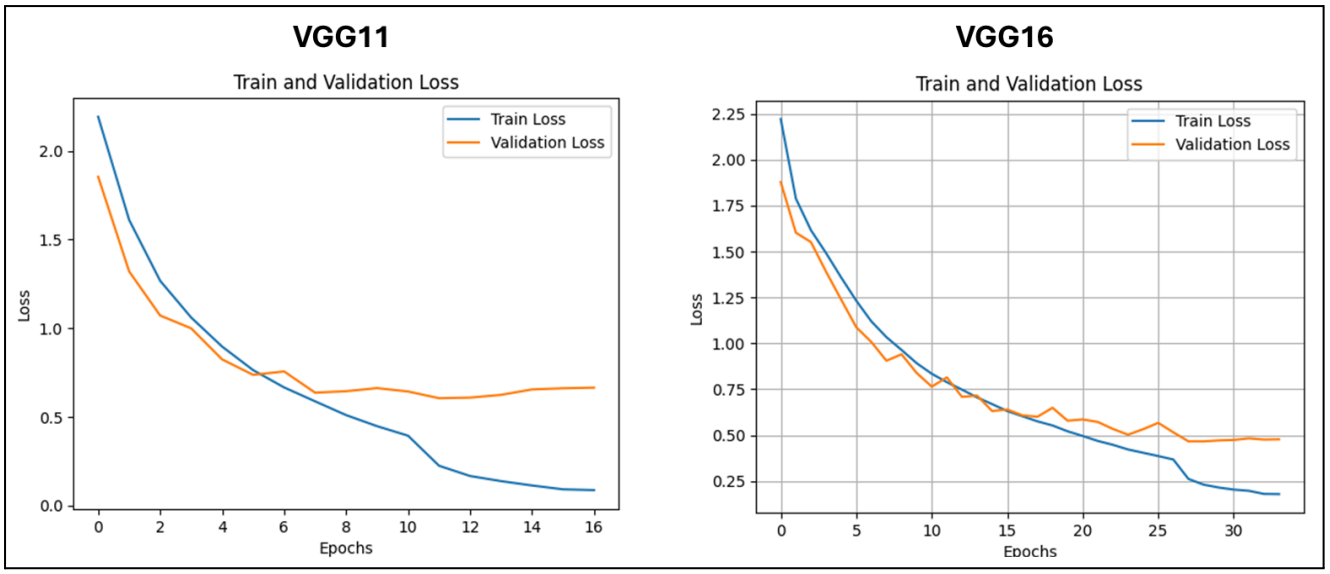

Figure 4: Train and validation losses of VGG models over number of epochs

3. Results

3.1 **Performance Metrics**

The performance of the VGG11 and VGG16 models was assessed using key evaluation metrics, including accuracy, precision, recall, and F1-score, as summarized in Table 1. VGG16 consistently outperformed VGG11 across all metrics, achieving an accuracy of 84.74% compared to VGG11's 81.61%. This difference highlights VGG16's enhanced capacity to effectively learn and classify features within the CIFAR-10 dataset. In addition to accuracy, VGG16 demonstrated superior precision, recall, and F1-score values, signifying its improved balance between false positives and false negatives.

Table 1. Model Skill Metrics of VGG11 vs VGG16

Model	Accuracy	Precision	Recall	F1
VGG11	0.8161	0.8176	0.8161	0.8161
VGG16	0.8474	0.8476	0.8474	0.8473

The ROC-AUC analysis, illustrated in Figure 4, further underscores the multi-class classification effectiveness of the VGG16 model. The ROC curves for individual classes reveal high true positive rates, with AUC values exceeding 0.96 for all categories. The "Automobile" and "Ship" classes achieved perfect AUC scores of 1.00, reflecting VGG16's exceptional discrimination capabilities for these categories. However, the "Cat" class exhibited the lowest AUC value at 0.96, indicating relatively lower discrimination performance for this class compared to others. These results reinforce the advantages of deeper architectures like VGG16 in achieving higher classification performance on diverse datasets such as CIFAR-10, while also highlighting potential areas for improvement in specific classes.

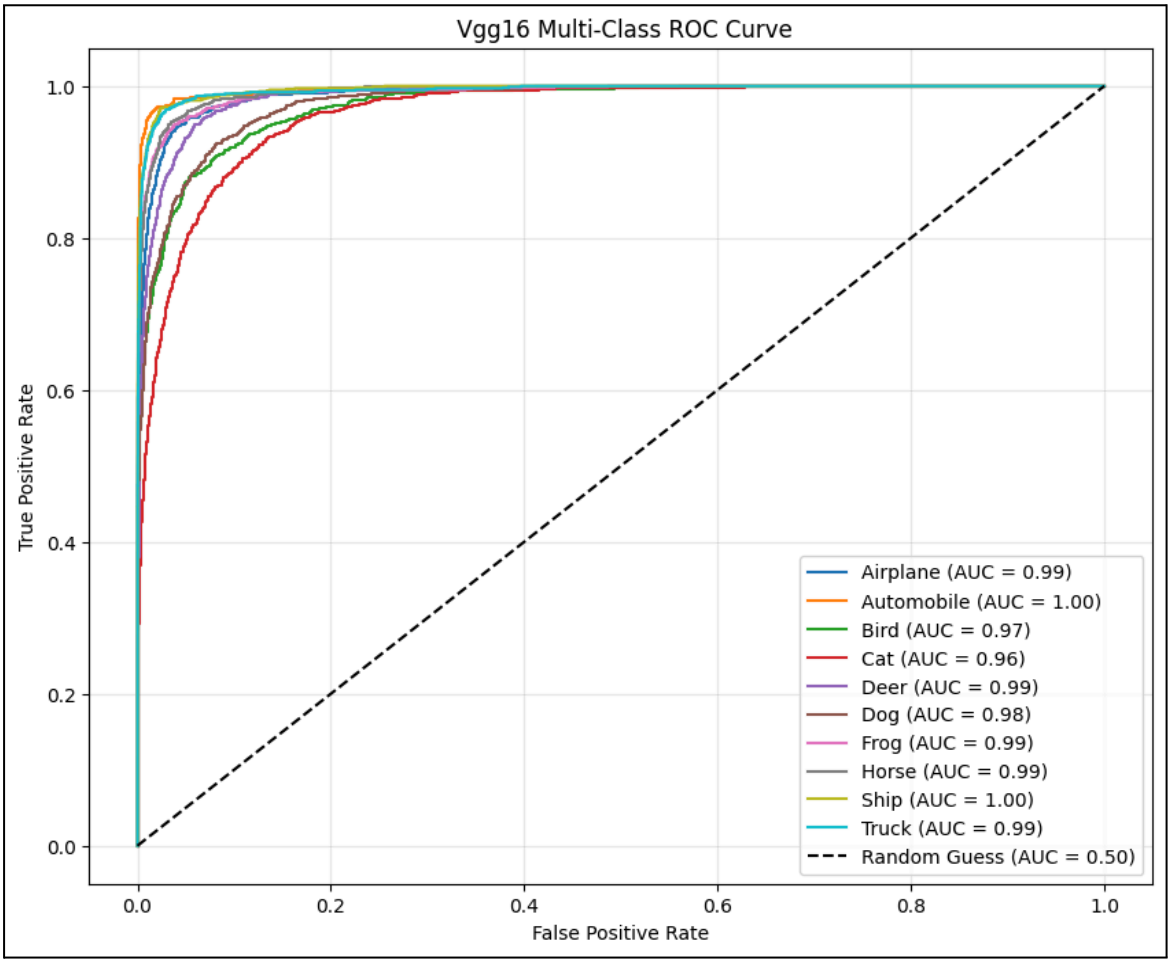

Figure 4: Multi-class ROC curves for VGG16 across CIFAR-10 classes, showcasing high AUC values for all categories, indicating strong classification performance.

3.2 Confusion Matrix

The confusion matrices for VGG11 and VGG16, presented in Figure 3, provide a detailed breakdown of the classification performance across all CIFAR-10 classes. These matrices reveal key insights into model behavior, particularly with respect to misclassification trends. VGG16 demonstrates improved overall accuracy and reduced misclassification rates compared to VGG11, as evidenced by higher true positive counts across most classes. However, both models show notable errors between visually similar classes, such as "Cat" and "Dog," where the models frequently confuse one for the other. This trend is particularly evident in VGG16, where the top misclassification pairs include "Cat → Dog" and "Dog → Cat" (Figure 5). Such errors may be attributed to

overlapping features in these classes, such as fur textures and similar shapes, making them challenging to differentiate.

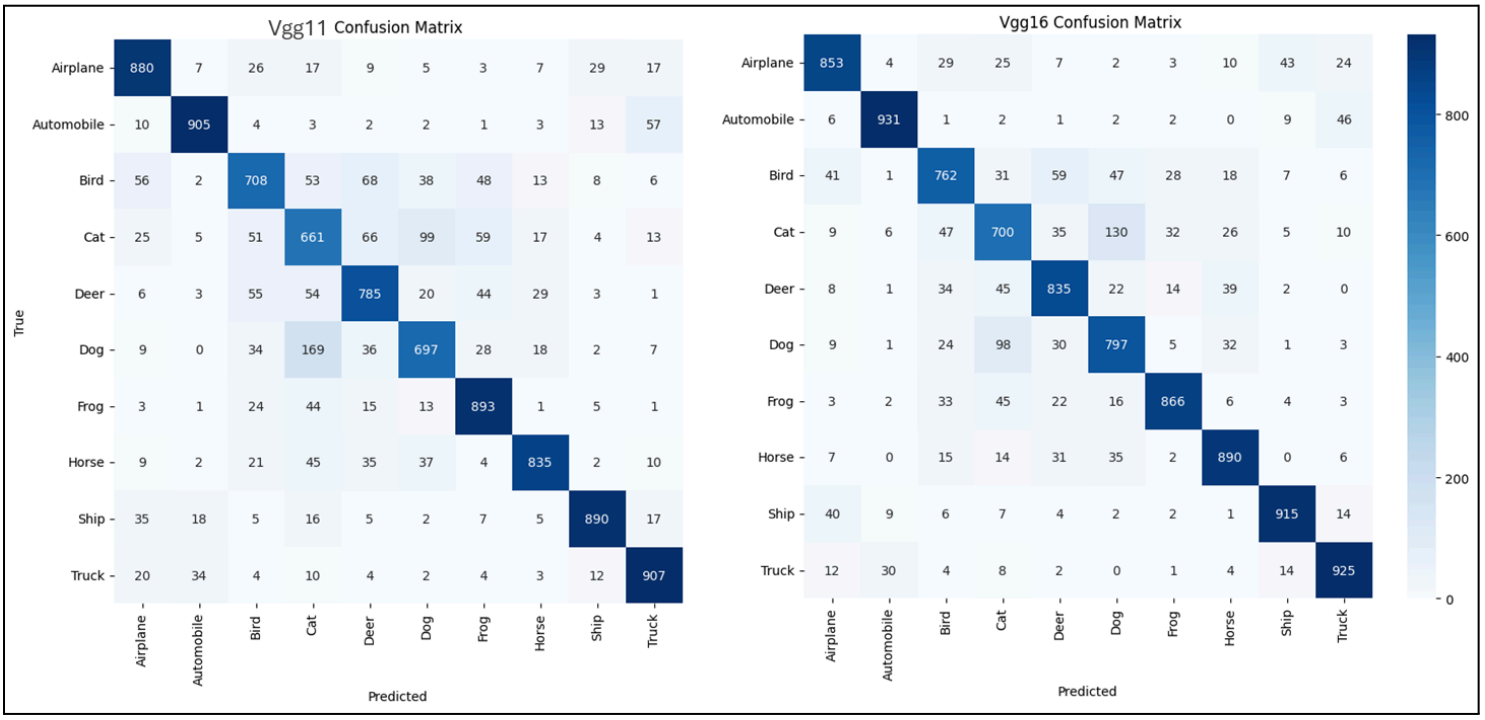

Figure 5: Confusion matrices for VGG-11 (left) and VGG-16 (right) models, showing the classification performance across CIFAR-10 classes. Each matrix illustrates true labels versus predicted labels, with darker shades indicating higher counts.

The analysis also highlights other prominent misclassification patterns, such as confusion between "Bird → Deer" and "Bird → Dog," which further underscore the models' limitations in learning distinct features for specific classes. Interestingly, the confusion between "Cat → Bird," though less frequent, also points to reliance on background features during prediction. These patterns suggest that while VGG16 exhibits better performance overall, further optimization or alternative feature extraction strategies might be necessary to address specific inter-class ambiguities.

The inclusion of a bar chart in Figure 5 emphasizes the most frequent errors for VGG16, with "Cat → Dog" dominating as the leading source of misclassification. This visual representation provides a clear and concise summary of where the model struggles most, paving the way for targeted interventions, such as data augmentation or specialized loss functions, to mitigate these issues. Overall, the confusion matrices and error analysis reveal not only the strengths of the VGG models but also the challenges in achieving perfect classification, particularly for visually similar categories.

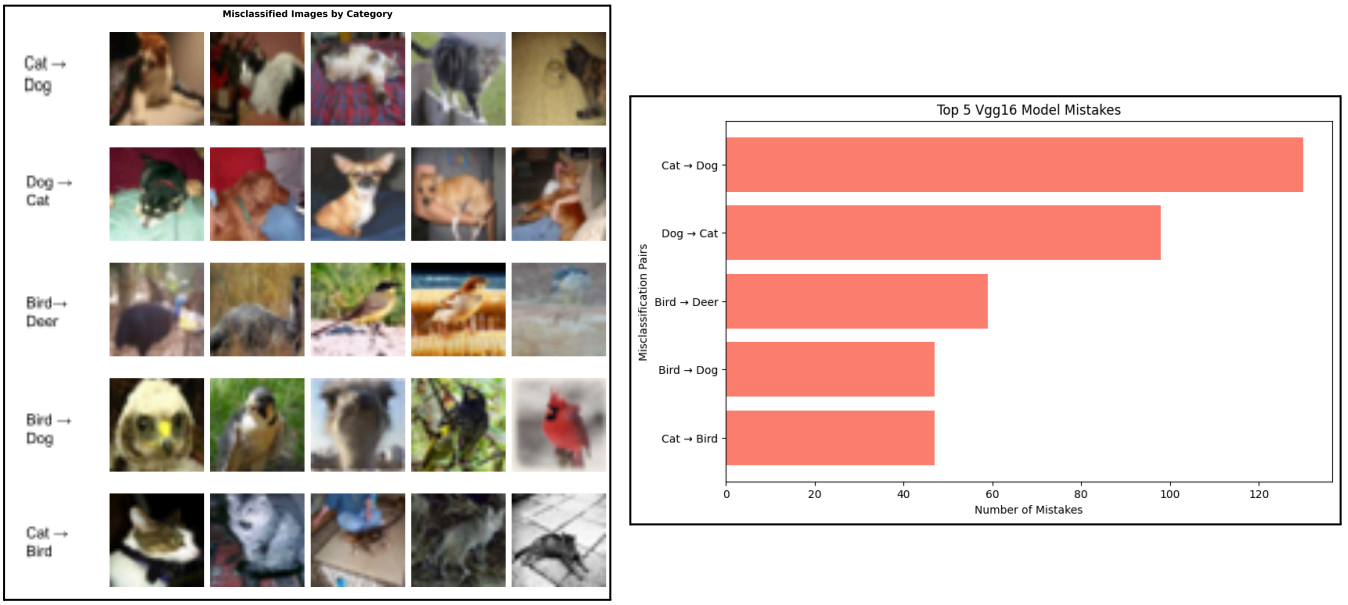

Figure 6: Top 5 misclassification pairs by the VGG16 model on the CIFAR-10 dataset. The most frequent mistakes include confusing "Cat" with "Dog" and vice versa, reflecting similarity in visual features between these classes. (A) shows sample images and (B) shows frequency of each errorr

3.3 **Activation Maps Visualization**

Activation maps generated using Grad-CAM (Selvaraju et. al, 2017) provide an interpretative lens into the decision-making processes of the VGG11 and VGG16 models, highlighting the areas of input images that contributed most significantly to the classification outcomes. These visualizations, as shown in Figure 6, reveal that both models generally focus on semantically meaningful regions of the images, reinforcing their ability to extract discriminative features effectively.

For instance, in correctly classified "Dog" images, the models primarily highlight regions around the face and body, suggesting that these are the most informative features for classification. Similarly, in "Airplane" images, attention is concentrated on the wings and fuselage, which are distinctive for identifying this class. These results confirm that the models have learned to leverage spatial hierarchies to focus on relevant object features, aligning well with the architectural goals of the VGG networks.

However, the activation maps also reveal important limitations in the models. In cases of misclassification, the highlighted regions are often diffuse or irrelevant, with attention sometimes drawn to background elements or unrelated features. For example, in a misclassified "Bird" image, the model might focus on tree branches or the surrounding foliage instead of the bird itself, demonstrating a reliance on context rather than the object. Similarly, some "Cat → Dog" misclassifications show activation across the entire image, lacking a concentrated focus on critical features like the head or body.

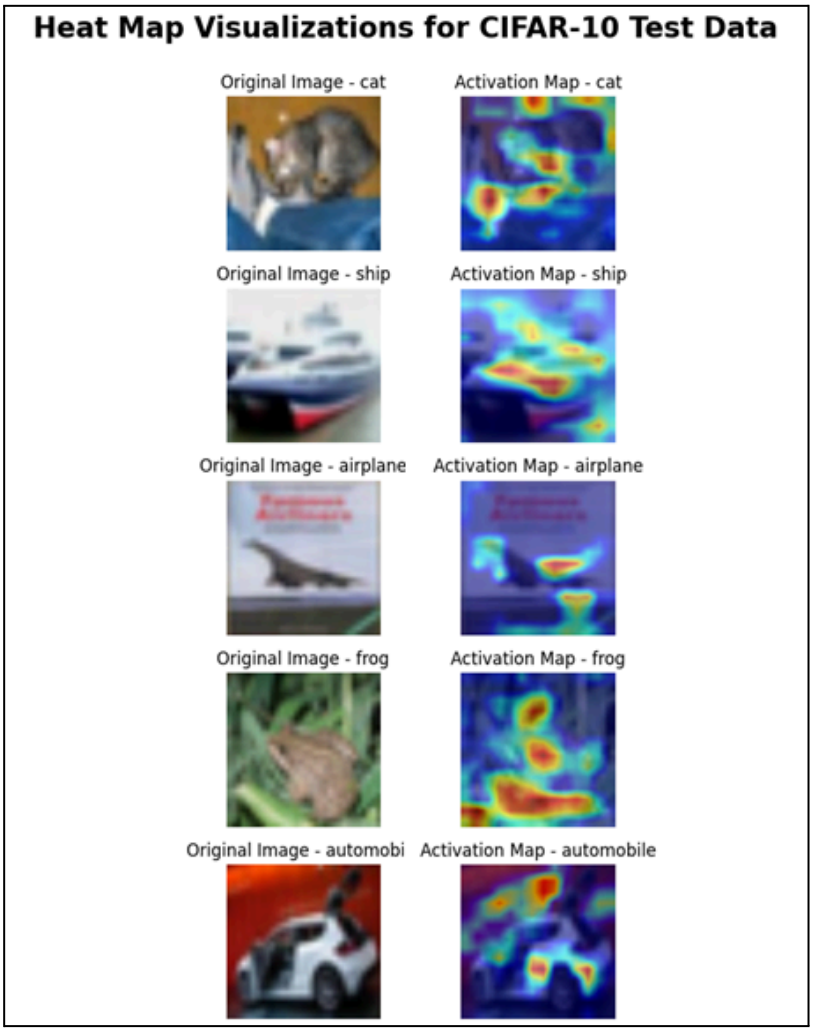

Figure 7: Heatmap visualizations of activation maps for selected CIFAR-10 test images using the VGG16 model. The left column shows the original images, while the right column displays the corresponding activation maps, highlighting regions of interest that contributed most to the model's predictions

These findings highlight potential areas for improvement. The presence of diffuse activation in certain cases suggests a need for enhanced data preprocessing or augmentation strategies, such as random cropping or background removal, to reduce reliance on irrelevant features. Regularization techniques, such as stronger dropout or adversarial training, could further encourage the model to focus on essential regions. Additionally, increasing the diversity of training samples for challenging classes could help the model generalize better and avoid biases stemming from specific contexts in the dataset.

Overall, the Grad-CAM visualizations serve as a powerful tool to interpret the inner workings of the VGG models. While they validate the models' ability to focus on relevant regions for many correctly classified images, they also expose weaknesses in misclassified cases, providing actionable insights for refining the models and improving their robustness. These findings underscore the importance of combining quantitative metrics with qualitative analysis to gain a comprehensive understanding of model behavior.

4. Discussion

This study highlights the strengths and limitations of VGG11 and VGG16 on CIFAR-10, a smaller dataset chosen over ImageNet due to reduced computational demands, allowing rapid prototyping. VGG16 outperformed VGG11 across all metrics, achieving higher accuracy (84.74% vs. 81.61%), precision, recall, and F1-score, attributed to its greater depth enabling better feature extraction. Unlike the original VGG study, which trained models on ImageNet for 72 epochs, this study implemented early stopping, resulting in significantly shorter training times.

Despite its strengths, VGG16 exhibited limitations, with confusion matrices revealing frequent misclassifications between similar classes like "Cat" and "Dog," often due to reliance on background features. Grad-CAM visualizations showed both models focused on relevant regions (e.g., wings for "Airplane," faces for "Dog"), but misclassified images displayed diffuse activations, indicating areas for improvement. Enhancing data augmentation and regularization could address these issues.

VGG16's ROC-AUC analysis showed robust performance, with AUC scores exceeding 0.96 for all classes and perfect scores for "Automobile" and "Ship." However, the "Cat" class had a slightly lower AUC of 0.96, reflecting challenges in distinguishing it from similar classes. Future work could test these models on more complex datasets like ImageNet or CIFAR-100 to validate findings and explore advanced techniques, such as attention mechanisms and tailored augmentations, to further enhance performance and generalization.

References:

Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. *International Conference on Learning Representations (ICLR)*. https://arxiv.org/abs/1409.1556

Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 248–255. https://doi.org/10.1109/CVPR.2009.5206848

Krizhevsky, A. (2009). *Learning Multiple Layers of Features from Tiny Images* (Technical Report). University of Toronto. https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). *Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization.* In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 618–626. DOI: 10.1109/ICCV.2017.74

.