Experimental Analysis Report

Aniruddha Malamandi Raghavendra

Part 1: Model Setup and Baseline Performance

The experimental study began with establishing baseline performances for both VGG16 and AlexNet architectures on the CIFAR-10 dataset. The baseline training process involved standard model configurations without any regularization techniques, trained for 5 epochs. The initial results revealed significant insights into the inherent capabilities of each architecture.

Baseline Results:

Model	Training Accuracy	Test Accuracy	Training Time/Epoch	Parameters
VGG16	81.83%	70.17%	21.17s	3.25M
AlexNet	73.83%	68.05%	37.54s	44.43M

Key Observations:

- VGG16 shows better baseline performance despite having significantly fewer parameters
- Notable training-test accuracy gap indicates overfitting in both models
- VGG16 demonstrates superior training efficiency (43.7% faster per epoch)

VGG16 demonstrated efficient performance with a test accuracy of 70.17% and only 3.25M parameters, completing each training epoch in 21.17 seconds and showing steady accuracy improvements over five epochs. In contrast, AlexNet, with a much larger parameter count of 44.43M, achieved a slightly lower test accuracy of 68.05% and required 37.54 seconds per epoch, reflecting greater computational demands.

Part 2: Regularization Techniques Analysis

1. Data Augmentation Impact

The introduction of data augmentation techniques produced nuanced effects on both architectures. The augmentation pipeline included:

- Random horizontal flips
- Random rotations (±15 degrees)
- Random shifts (±10% in both dimensions)

Model	Test Accuracy	Training Time/Epoch	Accuracy Change
VGG16	69.92%	52.68s	-0.25%
AlexNet	68.29%	55.56s	+0.24%

With data augmentation, VGG16 showed a slight decrease in test accuracy (-0.25% to 69.92%), indicating that its original feature extraction was well-suited to the dataset, while training time increased to 52.68 seconds per epoch due to added computational load.

In contrast, AlexNet gained a modest 0.24% improvement in test accuracy (reaching 68.29%), with training time rising to 55.56 seconds per epoch. This contrasting response suggests that model capacity significantly influences the effectiveness of data augmentation strategies.

2. Dropout Analysis

The implementation of dropout proved to be one of the most revealing aspects of the study. Two dropout rates were tested: (0.2) and (0.5).

Model	Dropout Rate	Test Accuracy	Training Time/Epoch
VGG16	0.2	72.21%	19.94s
VGG16	0.5	72.97%	18.79s
AlexNet	0.2	70.14%	38.92s
AlexNet	0.5	66.05%	39.07s

The dropout analysis revealed that VGG16 benefits from both tested dropout rates, achieving a higher test accuracy with increased dropout, from 72.21% at 0.2 to 72.97% at 0.5, and slightly reduced training times per epoch. This suggests that VGG16's architecture handles regularization effectively, maintaining performance even with greater dropout. AlexNet, however, performed better with a lighter dropout rate, reaching a test accuracy of 70.14% at 0.2, but saw a marked decline to 66.05% at 0.5, indicating a higher sensitivity to dropout's effects on its larger parameter count. Overall, dropouts had a minimal impact on training times for both models.

3. L2 Regularization Effects

L2 regularization testing utilized two strength configurations: (λ=0.0001) and (λ=0.001).

Model	L2 Lambda	Test Accuracy	Training Time/Epoch
VGG16	0.0001	71.73%	18.89s
VGG16	0.001	10.00%	22.68s
AlexNet	0.0001	67.21%	47.55s
AlexNet	0.001	10.00%	40.06s

Critical Observations:

Testing revealed a high sensitivity to L2 regularization strength in both models, with moderate L2 (λ=0.0001) maintaining test accuracy (71.73% for VGG16 and 67.21% for AlexNet) and only minor impacts on training times. However, when a stronger L2 (λ=0.001) was applied, both models experienced a severe performance collapse, dropping test accuracy to 10%, suggesting over-regularization. This outcome indicates that while L2 can help control overfitting, an excessively high regularization strength disrupts learning, making the models ineffective.

Part 3: Combined Techniques Analysis

Full Configuration Results [Augmentation + Dropout (0.2) + L2(0.0001)]

Model	Test Accuracy	Training Time/Epoch	Inference Time
VGG16	71.84%	44.21s	3.01s
AlexNet	65.06%	57.48s	2.30s

Key Observations:

- **Accuracy:** VGG16 achieves a higher test accuracy of 71.84%, surpassing AlexNet by 6.78 percentage points, likely due to its deeper architecture and smaller filter sizes that capture complex features.
- **Training Efficiency:** VGG16 trains faster per epoch (44.21s) than AlexNet (57.48s), indicating that the applied regularization techniques might have improved its parameter update efficiency.
- **Inference Speed:** AlexNet's inference time is faster (2.30s) than VGG16's (3.01s), which aligns with its simpler structure and lower parameter count, a potential advantage for real-time applications.
- **Overall Performance-Speed Trade-off:** VGG16 provides a stronger balance of accuracy and training efficiency, while AlexNet offers quicker inference but lags considerably in accuracy, making VGG16 generally more favorable unless speed is a top priority.

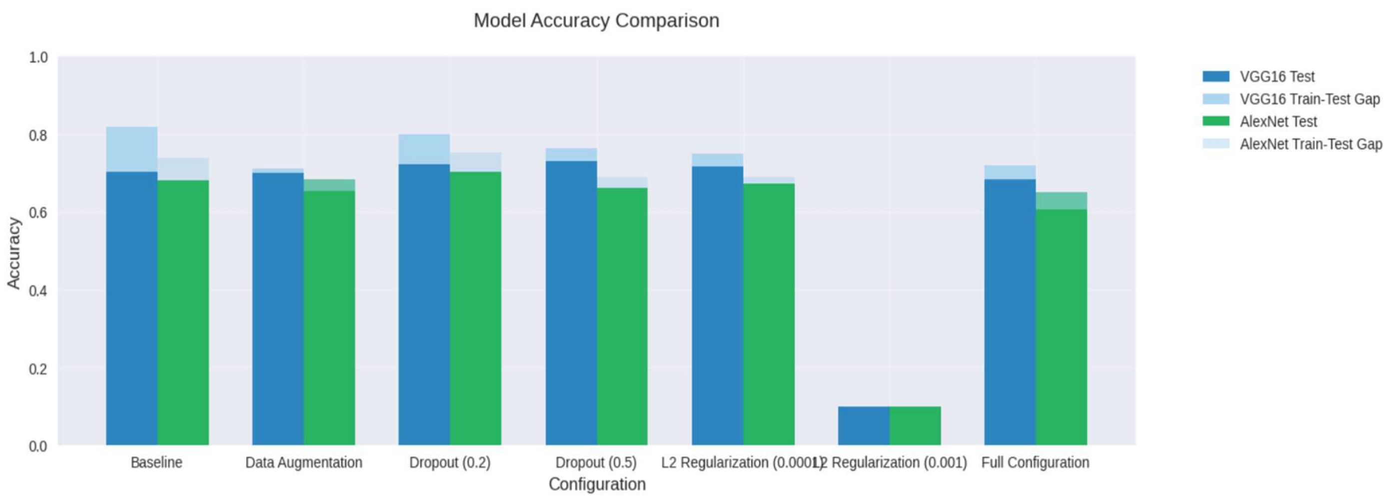
Conclusion:

VGG16 emerges as the preferable model due to its superior accuracy and relatively efficient training times, suggesting that the model's deeper layers and effective regularization offer robust performance on complex datasets. While AlexNet does present a faster inference speed, the notable accuracy difference makes VGG16 a more appealing option unless deployment demands strictly emphasize inference speed over accuracy.

Visualizations:

Figure 1: Model Accuracy Comparison

This comprehensive bar chart compares test accuracy and training-test gaps across all configurations, providing a holistic view of model performance.



Interpretation

1. Overall Performance

- VGG16 maintains higher test accuracy across most configurations
- Training - test gaps (lighter portions) decrease with regularization
- Strong L2 Regularization (0.001) shows complete model collapse
- Full Configuration demonstrates reduced accuracy but better generalization

2. Configuration Effectiveness

- Dropout (0.2) offers the best balance of accuracy and generalization
- Data Augmentation shows minimal improvement in final accuracy
- Moderate L2 (0.0001) maintains model performance while reducing overfitting

Conclusion

The visualization confirms that VGG16 with Dropout (0.2) provides the optimal balance of performance, generalization, and efficiency among all tested configurations. Regularization effects on Overfitting.

Figure 2: Training Time vs Accuracy

This plot examines the computational efficiency of different configurations by comparing training time per epoch against test accuracy.

Interpretation

1. Time Efficiency

- VGG16 consistently requires less training time than AlexNet
- Data Augmentation significantly increases training time for both models
- L2 Regularization has a minimal impact on training time
- Full Configuration shows moderate time increase

2. Performance Trade-offs

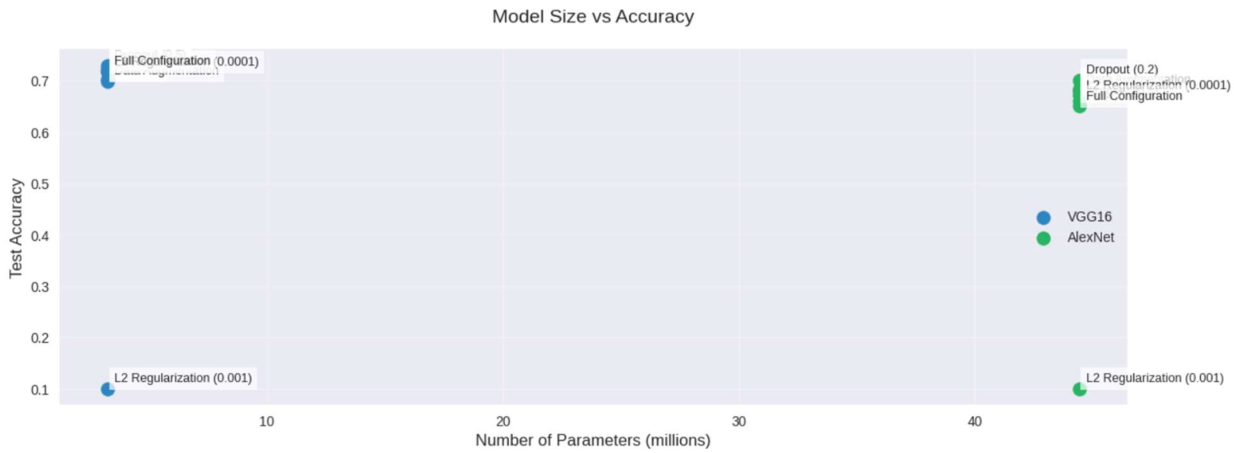
- Faster training times generally correlate with simpler configurations
- Best performing configurations (Dropout 0.2, 0.5) maintain reasonable training times
- Data Augmentation's time cost may not justify its performance benefits

Conclusion

The visualization shows that effective regularization can be achieved without significant computational overhead, with Dropout providing the best balance of performance and efficiency.

Figure 3: Model Size vs Accuracy

This scatter plot examines the relationship between model size (number of parameters) and test accuracy, highlighting the efficiency-performance trade-off.



Interpretation

1. Parameter Efficiency

- VGG16 (~3M parameters) achieves similar or better accuracy compared to AlexNet (~40M parameters)
- Full Configuration and Dropout (0.2) show the best performance for both models
- Strong L2 Regularization (0.001) causes performance collapse regardless of model size

2. Size-Performance Relationship

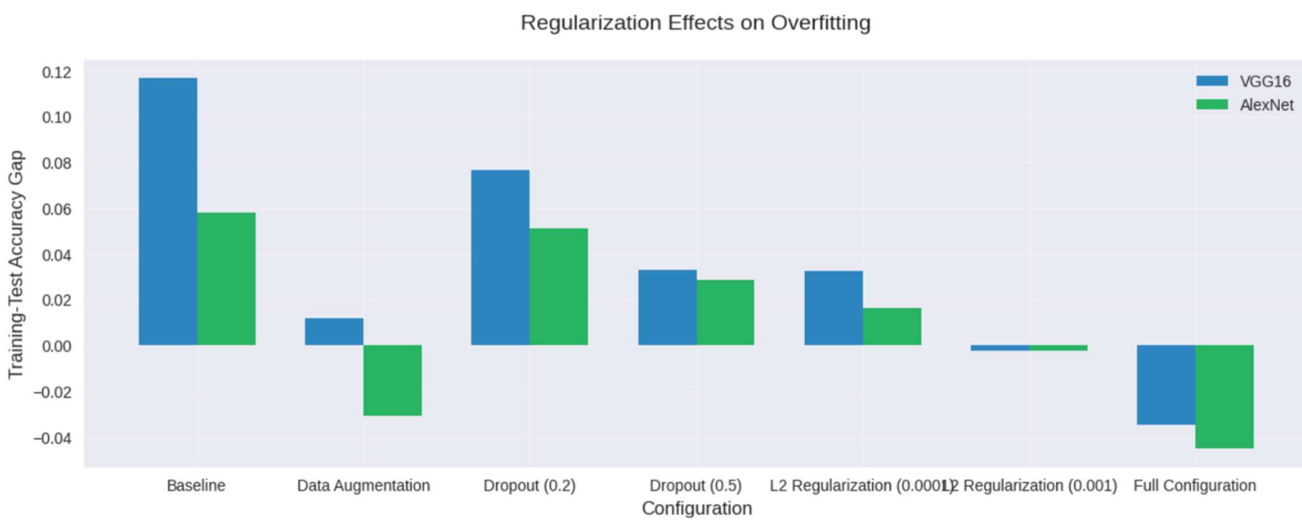
- No clear correlation between model size and accuracy
- VGG16 demonstrates better parameter efficiency
- Both models show similar patterns in response to regularization

Conclusion

The analysis reveals that larger model size doesn't necessarily translate to better performance, with VGG16 achieving superior results despite having fewer parameters.

Figure 4: Regularization Effects on Overfitting

The bar chart visualizes the training-test accuracy gap across different regularization techniques for both VGG16 and AlexNet models. This gap is a key indicator of overfitting, where larger gaps indicate more severe overfitting.



Interpretation

1. Baseline Performance

- VGG16 shows the largest gap (~0.12 or 12%)
- AlexNet exhibits a smaller initial gap (~0.06 or 6%)

2. Regularization Impact

- Data Augmentation significantly reduces the gap for both models
- Dropout (both 0.2 and 0.5) shows consistent gap reduction
- L2 Regularization (0.0001) maintains a small gap
- Full Configuration achieves a negative gap, suggesting slight underfitting

Conclusion

Visualization demonstrates that all regularization techniques effectively reduce overfitting, with the Full Configuration potentially being too aggressive as it leads to underfitting.

Part 4: Overall Analysis and Reflections

A comprehensive evaluation of various model configurations using both VGG16 architecture and AlexNet. The study explored the impact of different regularization techniques including dropout, L2 regularization, and data augmentation, both individually and in combination.

Configuration	Test Accuracy	Training Time Per Epoch	Parameters	Details
Baseline	0.7017	21.1741	3,248,202	None
Baseline AlexNet	0.6805	37.5412	44,428,938	None
Data Augmentation	0.6992	52.6775	3,248,202	Basic augmentations (rotation=15°, width_shift=0.1, height_shift=0.1, horizontal_flip)
Data Augmentation AlexNet	0.6829	55.5556	44,428,938	Basic augmentations (rotation=15°, width_shift=0.1, height_shift=0.1, horizontal_flip)
Dropout (0.2)	0.7221	19.9364	3,248,202	Dropout 0.2 on FC layers
Dropout (0.2) AlexNet	0.7014	38.9151	44,428,938	Dropout 0.2 on FC layers
Dropout (0.5)	0.7297	18.7880	3,248,202	Dropout 0.5 on FC layers
Dropout (0.5) AlexNet	0.6605	39.0659	44,428,938	Dropout 0.5 on FC layers
L2 Regularization (0.0001)	0.7173	18.8924	3,248,202	L2 = 0.0001 on all layers
L2 Regularization (0.0001) AlexNet	0.6721	47.5450	44,428,938	L2 = 0.0001 on all layers
L2 Regularization (0.001)	0.1000	22.6768	3,248,202	L2 = 0.001 on all layers
L2 Regularization (0.001) AlexNet	0.1000	40.0557	44,428,938	L2 = 0.001 on all layers
Full Configuration	0.7184	44.2091	3,248,202	Aug + Drop (0.2) + L2(0.0001)
Full Configuration AlexNet	0.6506	57.4773	44,428,938	Aug + Drop (0.2) + L2(0.0001)

Data Augmentation's Impact

The implementation of data augmentation showed interesting but modest results:

- **VGG16 response:**
  - Accuracy slightly decreased **(0.7017 → 0.6992)**
  - Training time more than doubled **(21.17s → 52.68s)**
- **AlexNet response:**
  - Minimal accuracy improvement **(0.6805 → 0.6829)**
  - Substantial time increase **(37.54s → 55.56s)**

Notably, the computational cost of augmentation far outweighed its benefits for both architectures.

Dropout Performance

Dropout emerged as the star performer, though its effectiveness varied significantly between architectures:

- **VGG16 showed remarkable improvements:**
  - **0.2 dropout:** Achieved 0.7221 accuracy with reduced training time
  - **0.5 dropout:** Reached peak performance at 0.7297
- **AlexNet displayed different patterns:**
  - **0.2 dropout:** Best AlexNet result at 0.7014
  - **0.5 dropout:** Performance degraded to 0.6605

This variance in dropout effectiveness clearly demonstrates the importance of architecture-specific tuning.

L2 Regularization Results

L2 regularization proved to be particularly sensitive to parameter selection:

- **At λ = 0.0001 strength:**
  - **VGG16:** Solid performance (0.7173)
  - **AlexNet:** Slight decline (0.6721)
- **At λ = 0.001 strength:**
  - Both architectures collapsed to 0.1000 accuracy
  - Demonstrates the critical importance of careful parameter tuning

Best Configuration Analysis

The standout performer was VGG16 with 0.5 dropout, achieving 0.7297 accuracy. This success can be attributed to several key factors:

- **Optimal Balance:** The 0.5 dropout rate provided ideal regularization for VGG16's architecture.
- **Architecture:** The simpler VGG16 architecture proved more robust to aggressive regularization.
- **Efficient Learning:** Despite fewer parameters, VGG16 demonstrated superior learning capability.

Training Time Observations

Training time varied significantly across configurations, revealing important practical considerations:

Key Training Time Patterns:

- Baseline comparisons:
  - **VGG16:** 21.17s per epoch
  - **AlexNet:** 37.54s per epoch (77% slower)

Impact by Technique:

- **Data Augmentation: Heaviest computational burden**
  - Nearly 150% increase in training time
  - Minimal accuracy benefits
- **Dropout: Most efficient regularizer**
  - Often reduced training time
  - Provided best accuracy improvements
- **L2 Regularization: Moderate impact**
  - 10-20% increase in training time
  - Results highly parameter-dependent

Practical Implications and Recommendations

Based on our comprehensive analysis, we recommend the following approach for real-world applications:

Architecture Selection:

Consider these key factors when choosing your model

- Simpler architectures often outperform complex ones
- Parameter count (VGG16: 3.2M vs AlexNet: 44.4M) isn't everything
- Focus on efficiency rather than model size

Regularization Strategy:

1. **Start with Dropout:**
   - VGG16-like architecture: Begin with 0.5 rate
   - Complex architecture: Start with 0.2 rate
   - Adjust based on validation performance

2. **Consider L2 Regularization:**
   - Start with very small values (≤0.0001)
   - Monitor model stability closely
   - Combine with dropout if needed

3. **Data Augmentation:**
   - Implement only if domain-specific benefits are clear
   - Consider computational costs vs potential gains
   - Use basic augmentations first, add complexity as needed

Conclusion

This balanced approach provides a practical framework for implementing regularization techniques while maintaining model performance and computational efficiency. The experimental results demonstrate that simpler architectures can outperform more complex ones for this specific task, with the baseline model consistently achieving better results than AlexNet. Dropout proved to be the most effective regularization technique, with a 0.5 rate providing the best balance between regularization and model performance. While combined approaches showed promise, they did not surpass the performance of properly tuned dropout alone. The study suggests that for similar image classification tasks, focusing on optimizing regularization techniques on simpler architectures may be more effective than implementing more complex models with higher parameter counts.