**Experiment Run**

**Experiment Run Report**

**Experiment Title:** Numerosity-Based Categorization - Experiment Run 5

**Date:** 7/03/2025

**Researcher:** Karoki Evans Njogu

**1. Experiment Details**

| **Parameter** | **Value** |
| --- | --- |
| Seed | 42 |
| Dataset Size | 5000 samples |
| Image Size | 128x128 pixels |
| Categories | Few (1-5), Medium (6-15), Many (>16) |
| Batch Size | 256 |
| Learning Rate | 0.0001 |
| Epochs | 20 |
| Optimizer | AdamW |
| Dropout Rate | 0.4 |
| Weight Decay | 5e-4 |
| Loss Function | CrossEntropyLoss |
| Early Stopping | Yes (Patience = 5) |
| Device Used | GPU – NVIDIA L4 |
| eps | 1e-6 |
| betas | 0.9, 0.98 |
| Accumulation steps | 2 |

**2. Experiment Setup**

* **Dataset:** Synthetic Dot Patterns
* **Model Architecture:** CNN-Transformer architecture
* **Training Strategy:**
  + Train on 70% of data.
  + Validate on 15%.
  + Test on 15%.
* **Evaluation Metrics:**
  + Accuracy
  + Loss Curves
  + Confusion Matrix
  + Precision, Recall, and F1-Score

**3. Training & Validation Performance**

**3.1 Loss and Accuracy Trends**

| **Epoch** | **Train Loss** | **Validation Loss** | **Validation Accuracy (%)** |
| --- | --- | --- | --- |
| 1 | 0.6258 | 3.9177 | 17.20% |
| 2 | 0.2809 | 4.3694 | 17.20% |
| 3 | 0.2418 | 2.7796 | 26.93% |
| 4 | 0.2068 | 0.5529 | 79.07% |
| 5 | 0.1733 | 0.2157 | 91.73% |
| 6 | 0.1746 | 0.2260 | 90.80% |
| 7 | 0.1380 | 0.2675 | 90.00% |
| 8 | 0.1507 | 0.1666 | 93.07% |
| 9 | 0.0973 | 0.2100 | 90.93% |
| 10 | 0.1296 | 0.1616 | 93.47% |
| 11 | 0.0927 | 0.3915 | 86.93% |
| 12 | 0.1672 | 0.1621 | 93.60% |
| 13 | 0.1563 | 0.1776 | 91.47% |
| 14 | 0.1040 | 0.3166 | 88.40% |
| 15 | 0.0768 | 0.1538 | 93.07% |
| 16 | 0.0817 | 0.1654 | 93.33% |
| 17 | 0.0450 | 0.1765 | 93.33% |
| 18 | 0.0314 | 0.2202 | 93.07% |
| 19 | 0.0246 | 0.3278 | 90.93% |
| 20 | 0.0223 | 0.2017 | 92.67% |

**3.2 Loss Curve & Accuracy Plot**

A graph of loss curves


validation accuracy graph


**4. Test Set Evaluation**

**Final Test Accuracy:** 93.33%

**4.1 Confusion Matrix**

**A graph of a test data

AI-generated content may be incorrect.**

**4.2 Classification Report**

| **Class** | **Precision** | **Recall** | **F1-Score** | **Support** |
| --- | --- | --- | --- | --- |
| Few | 0.94 | 0.97 | 0.95 | 122 |
| Medium | 0.93 | 0.89 | 0.91 | 277 |
| Many | 0.94 | 0.95 | 0.95 | 351 |

**5. Observations & Insights**

* **Key Findings:**
  + Architecture Change: The model was modified from a pure CNN to a hybrid CNN-Transformer architecture.
  + Better generalization, increasing the batch size to 256 improved validation accuracy and helped the model generalize better.
  + There is performance improvement since the final model reached a test accuracy of 93.33%, showing a consistent improvement in accuracy and stability across epochs.
  + The loss trends in both training and validation loss curves are smoother, indicating better convergence and reduced overfitting.
  + The generalization is better, the model now effectively classifies across all categories (Few, Medium, Many), with a high precision and recall for all three classes.
* **Error Analysis:**

A collage of white circles

AI-generated content may be incorrect.

* + The "Few" and "Many" classes performed best, achieving over 95% recall.
  + The "Medium" class saw a few misclassifications, particularly with "Many" samples. However, the model still maintained an F1-score of 0.91 for this category.
  + While the validation loss is slightly higher towards the end, the accuracy remains stable, suggesting minor overfitting but still strong generalization.
  + Some Overfitting Still Present: Even with increased dropout and weight decay, the validation loss fluctuates in later epochs, meaning further regularization might still be needed.
* **Next Steps:**
  + Shape Generalization Study: Now that the baseline performance is strong, the next phase will test generalization by introducing different shapes beyond circles.
  + Further Regularization: Experiment with dropout tuning and weight decay adjustments to ensure the model does not overfit to specific patterns.
  + Data Augmentation: Consider applying transformations (rotation, scaling, contrast changes) to make the model more robust to visual variations.
  + Transformer Attention Analysis: Investigate how the self-attention layers influence classification decisions.

**6. Conclusion**

This final run successfully demonstrated that combining CNNs with Transformer-based representations significantly improves classification performance. The model generalizes well, but there is still potential for further refinement through architectural optimizations and additional generalization tests.

**7. Additional Notes**

* Reproducibility was ensured by setting a fixed random seed and using pre-saved datasets.
* This run also followed the structured experiment template, making future runs easy to compare.
* Some variability in validation loss was observed, which may indicate the need for better regularization techniques.
* Early stopping was applied, preventing overfitting, but further adjustments may be needed.