**Understanding High Loss Against Strong Accuracy in Fine-Grained Image Classification:**

**---------------------------------------------------------------------------------------------------**

**Abstract**

In fine-grained image classification, models often exhibit a mismatch between validation accuracy and categorical crossentropy loss. This paper explores this phenomenon through experimental evaluation of multiple CNN architectures trained on the Stanford Dogs dataset (120 classes). Despite achieving over 60% validation accuracy, models report high validation loss values (~2.5+). We analyze the root causes and architectural behaviors contributing to this pattern and recommend practical solutions for future fine-grained tasks.

**1. Introduction**

Fine-grained classification involves distinguishing between visually similar classes (e.g., dog breeds), requiring models to extract subtle, localized patterns. This introduces challenges such as class imbalance, overfitting, and weak generalization — especially with limited data per class. Traditional evaluation using only accuracy may not reflect the full picture, as high loss can still occur even when accuracy appears reasonable.

**2. Dataset Overview**

* **Stanford Dogs Dataset**
* 120 classes with class imbalance (some classes <100 samples)
* Image sizes normalized and augmented
* Different training/validation splits tested:
  + 80-10-10 → early plateau in validation
  + 70-20-10 → slight improvement
  + 60-30-10 → most stable generalization

**3. Models and Techniques**

**📌 Tested Architectures**

| **Model** | **Train Accuracy** | **Validation Accuracy** | **Notes** |
| --- | --- | --- | --- |
| MobileNetV2 | ~10% | ~9% | Underfits heavily |
| EfficientNetB0 | ~40% | ~36% | Balanced, limited feature depth |
| ResNet50 | ~60% | ~41% | Stable but saturates |
| **DenseNet121** | **99%** | **60–69%** | Best performer, skip connections improve reuse and generalization |

**🔧 Training Techniques**

* Data Augmentation (flip, zoom, rotation)
* Fine-Tuning (last 20 layers unfrozen)
* Regularization (L2 penalty at λ = 0.001)
* Dropout (less effective)
* Early Stopping (based on validation loss)

**4. Observations**

**🔍 Why Does Loss Remain High Despite Good Accuracy?**

1. **Categorical Crossentropy Nature**:  
   Penalizes even slightly low-confidence correct predictions, which are common in 120-class problems.
2. **Regularization Impact (L2)**:  
   Increases the total loss by adding a penalty term to avoid overfitting — accuracy improves, but loss does not necessarily drop.
3. **Model Confidence vs. Correctness**:  
   A correct prediction with 60% confidence still incurs a noticeable penalty.
4. **Data Limitations**:  
   Many classes have too few samples. The model ends up memorizing rather than generalizing, especially in early epochs.

**5. Case Study – DenseNet121**

**🧠 Epoch Breakdown**

| **Epoch Range** | **Behavior** |
| --- | --- |
| 1–4 | Accuracy jumps from 29% to 63% (augmentation helped) |
| 5–10 | Steady accuracy climb; loss decreased gradually |
| 11–13 | Peak validation accuracy at 69.84%, ~75% train accuracy |
| 14–16 | Slight val dip; early stopping avoided overfitting |

Even at this performance level, validation loss stayed around **2.5+**, due to L2 penalties and prediction confidence issues.

**6. Conclusion**

Despite the common perception that high loss = poor performance, this project shows that **context matters**. When dealing with:

* Fine-grained data
* Many classes
* Regularization
* High penalty loss functions

…a **high loss value can still accompany strong generalization**. DenseNet121, with feature reuse and skip connections, proved to be the most reliable model under these constraints.