Bird Classification System: Implementation Analysis

Jahnabi Roy (2022CS11094)
Abhinav Rajesh Shripad (2022CS11596)

1 Overview

This report details the implementation and training results of a PyTorch-based bird classification system. We tested our model with Train-Val-Test split of 80-10-10 and achieved an accuracy of 89.06% on the test dataset. The final model bird.pth was trained on Train-Val split of 80-20 and achieved a final Validation accuracy of 89.61%.

2 System Configuration

Learning Rate:	0.001	**Number of Epochs:**	80
Train-Val-Test Split:	80-10-10	**Batch Size:**	128
Early Stopping Patience:	20 epochs	**Image Size:**	64×64 pixels
Number of Classes:	10	**Data Augmentation :**	Ver/Hor/15° Rotate

3 Model Architecture

3.1 Neural Network Components

3.1.1 Background Removal Network

- **Encoder**:
  - Input → Conv2D(3→64) → BatchNorm2D → ReLU → MaxPool2D
  - Conv2D(64→128) → BatchNorm2D → ReLU → MaxPool2D
  - Conv2D(128→256) → BatchNorm2D → ReLU → MaxPool2D
  - Conv2D(256→512) → BatchNorm2D → ReLU → MaxPool2D
- **Decoder**:
  - ConvTranspose2D(512→256) → BatchNorm2D → ReLU
  - ConvTranspose2D(256→128) → BatchNorm2D → ReLU
  - ConvTranspose2D(128→64) → BatchNorm2D → ReLU
  - ConvTranspose2D(64→1) → Sigmoid

3.1.2 Classification Network

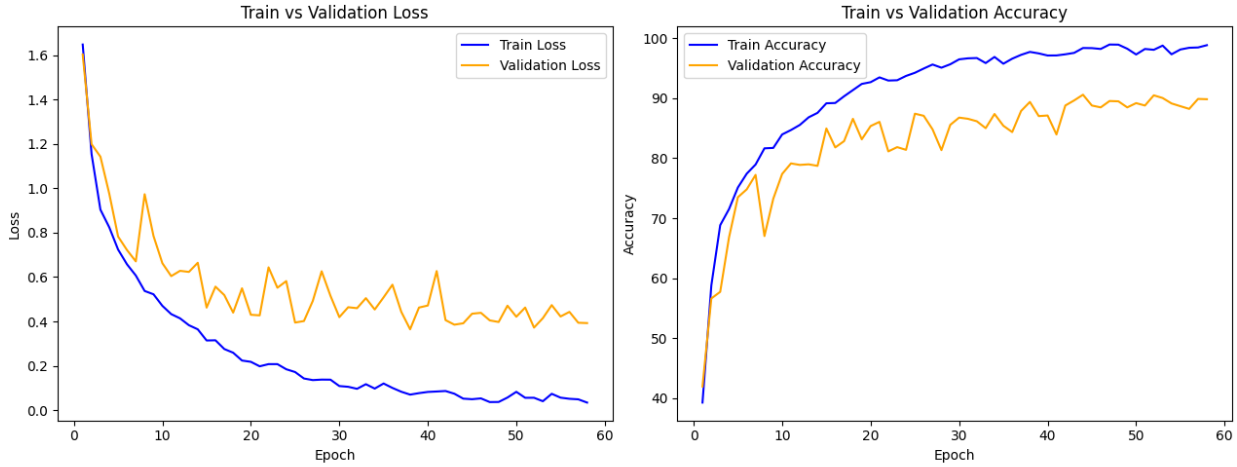
- Four convolutional blocks with structure:
  - Conv2D → BatchNorm2D → ReLU → MaxPool2D
  - Channel progression: 3→64→128→256→512
- Fully connected layers:
  - Flatten
  - Linear(8192→1024) → BatchNorm1D → ReLU
  - Linear(1024→512) → BatchNorm1D → ReLU
  - Linear(512→10)

3.2 Regularisation

- Early Stopping : 20 Epochs
- Batch Normalisation

4 Training Analysis

4.1 Training Progression



4.2 Training Observations

- Both Training and Validation Loss show a decreasing trend, though validation loss is not as smooth as the training loss. Similarly Training and Validation Accuracy increase as the training progresses. The progression is much smoother for training than for validation, however LR=0.001 output the best validation loss and accuracy than others.

5 Results

5.1 Regularisation

- With **Dropout** : **76%** accuracy on test set

- With **Batch Normalisation** : **80.5%** accuracy on test set
- With **Batch Normalisation, early stopping and adding data augmentation** : **89.06%** accuracy on test set

5.2 Performance Metrics

- Total Dataset Size: 9,960 images
- Train-Val-Test Split : 80-10-10
- Final Accuracy on Test Set: 89.06%

6 GradCAM Results

The GradCAM program was run over the validation dataset, which was also subject to transformations applied to train dataset (horizontal flip, vertical flip and at max 15 degrees rotation). Hence the images are pixellated. However the GradCAM results on the Classifier CNN do reveal what the model has learnt.

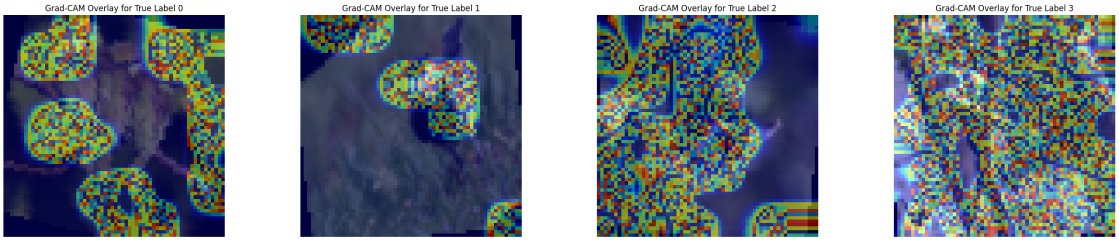

Grad-CAM Overlay for True Label 0
Grad-CAM Overlay for True Label 1
Grad-CAM Overlay for True Label 2
Grad-CAM Overlay for True Label 3

Figure-1 : Classes 0-3

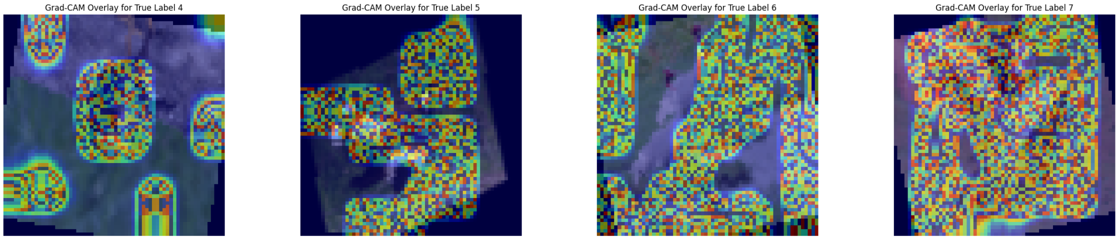

Grad-CAM Overlay for True Label 4
Grad-CAM Overlay for True Label 5
Grad-CAM Overlay for True Label 6
Grad-CAM Overlay for True Label 7

Figure-2 : Classes 4-7

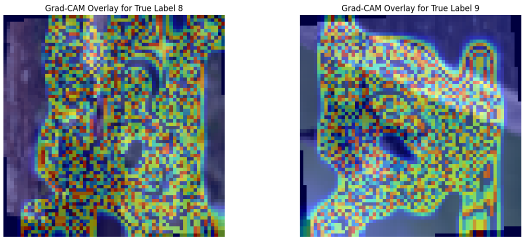

Grad-CAM Overlay for True Label 8
Grad-CAM Overlay for True Label 9

Figure-3 : Classes 8-9