

# Animal Image Classifier

By: Isaiah Doan, Owen Downs, Chhi Ly



# Building a Cats vs. Dogs vs. Birds Image Classifier

- Project Goal: Classify images into three classes (Cats, Dogs, Birds)
- Motivation: Expand ML-driven capabilities in animal identification
- High-level Approach: Transfer learning with VGG16

# Data Preparation & Cleaning

- Source: Organized images into cats, dogs, birds directories
- Preprocessing: Removed zero-byte & corrupted files
- Ensured balanced classes & representative validation set

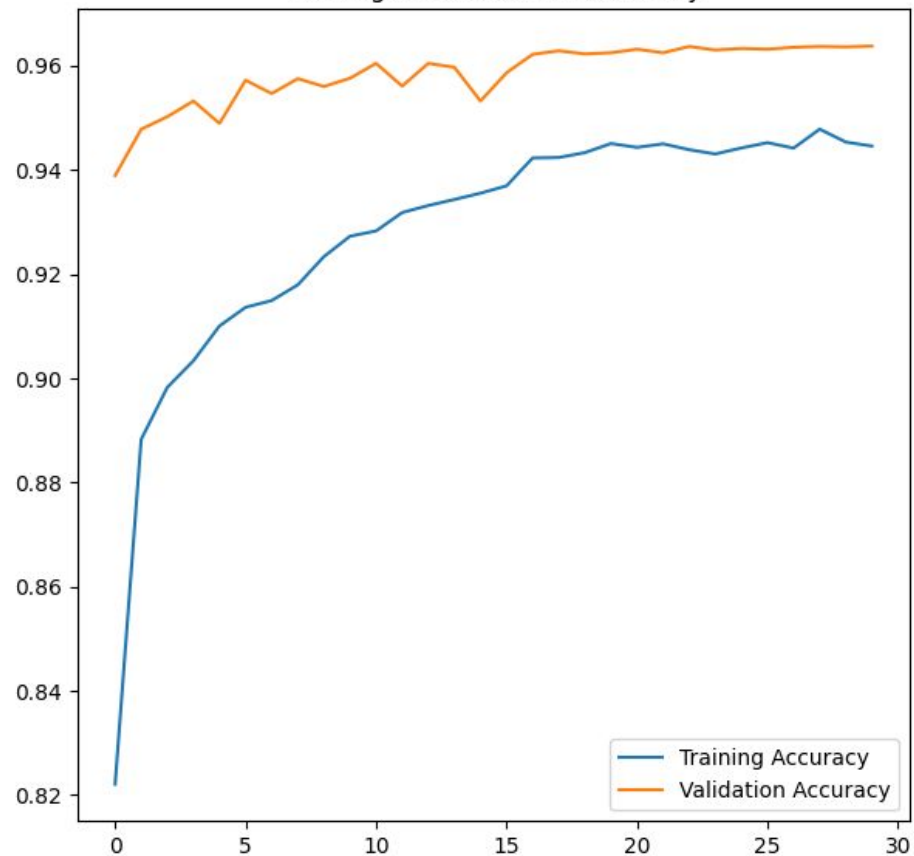
# Leveraging Transfer Learning

- Base Model: VGG16 (pre-trained on ImageNet)
- Conv Base Frozen: Retain learned features
- Added Dense + Dropout layers for classification
- Final output: 3-class softmax layer

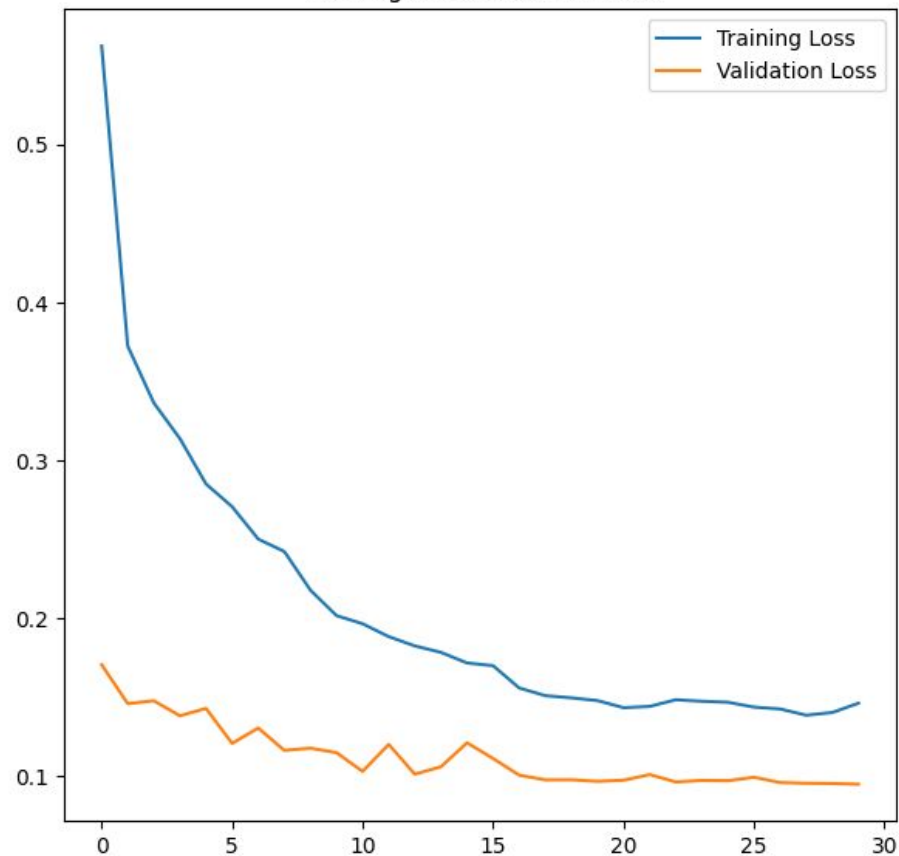
# Training Configuration

- Loss: `sparse_categorical_crossentropy` (simplified label encoding)
- Data Augmentation: Mild rotations, shifts, zooms, flips
- Callbacks: `EarlyStopping`, `ReduceLROnPlateau` for stable convergence
- Achieved ~94–95% validation accuracy

Training and Validation Accuracy



Training and Validation Loss



# Key Challenges & Solutions

- Data Integrity: Cleaned dataset to avoid runtime errors
- Class Mode & Output Structure: Used sparse labels for seamless training
- Ensuring Generalization: Augmentation & callbacks to prevent overfitting

# Results & Takeaways

- High Validation Accuracy (~95%) indicates strong generalization
- Minor accuracy fluctuations are normal near performance plateau
- No significant need for class weights due to balanced dataset
- Smooth pipeline = stable and reliable model for production



# Demo: Code & Model in Action

- Quick run-through of the app.py Flask interface
- Show how to upload an image & get a prediction
- Verify prediction with a known sample

Repo can be found here: <https://github.com/lsc-compsci/CPTS437Project/tree/main>