Image Classification: 225 Aves Species

Arsh Irfan Modak,[1] Omkar Waghmare,[2] Siddarth Sathyanarayanan[3]
Northeastern University Khoury College of Computer Sciences[1,2,3]
modak.a@northeastern.edu,[1] waghmare.o@northeastern.edu[2], sathyanarayanan.s@northeastern.edu[3]

Abstract

Bird watching is the practice of observing birds in their natural environment. The objective of this project is to assist birdwatchers as well as ornithologists correctly identify different species of birds with ease. To achieve this, we implement Supervised Machine Learning techniques such as deep learning and classification algorithms to develop a model to accurately identify 225 different bird species given an image of a bird.

1 Introduction

More than 45 million people watch birds around their homes and away from home, according to the findings of the U.S. Fish & Wildlife service.[1] Nowadays, bird species identification is seen as a mystifying problem which often leads to discombobulation and uncertainty. Many people visit bird sanctuaries to look at the birds, while they barely recognize the differences between various species of birds and their characteristics. Understanding such discrepancies between species can increase our knowledge of birds, their ecosystems and their biodiversity.

The identification of birds with bare eyes is based solely on the basic characteristics due to observer constraints such as location, distance and equipment. Appropriate classification based on specific characteristics is often found to be tedious. Even ornithologists have faced difficulties in distinguishing bird species. To properly identify a particular bird, they need to have all the specificities of birds, such as their distribution, genetics, breeding climate and environmental impact.[2] In this project, we develop a model which can classify birds solely using images of the birds, thereby eliminating the need of knowing these specificities. In the future this model could also be applied to create a platform that can identify more species of birds which will give additional information such as habitat, conservation status, as well as links to other resources pertaining to that species.

2 Technical Approach

In the following sections, we will give details on the methods used to develop the classification models. The first method was for the development of traditional supervised learning models such as Support Vector Machines (SVMs) and Logistic Regression. The second method was for the development of deep learning models such as Convolutional Neural Networks (CNNs).

2.1 Traditional Supervised Learning Models

2.1.1 Logistic Regression

The extracted features were used to train the Logistic Regression model. The l2 norm was used as penalty, Limited-memory Broyden-Fletcher-Goldfarb-Shanno (lbfgs) was selected as the solver and λ value was set to 1. The maximum number of iterations for the solver to converge was set to 10,000. Since there were a total of 225 classes, one-vs-rest logistic regression was used. In this method, a separate model is trained for each class, thereby turning it into 225 binary classification problems. The mathematical background of Logistic Regression is as follows.[3]

Hypothesis: $h_\theta(X) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X)}}$

Cost Function: $J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}[\, y(i) \log (h\theta(x(i))) + (1 - y(i)) \log (1 - (h\theta(x(i))))] + \frac{\lambda}{2m}\sum_{j=1}^{n} \theta_j^2$

2.1.2 Support Vector Machines

Like logistic regression, the extracted features were also used to train the SVM model. Again, the l2 norm was used as penalty. C value was set to 1 and lbfgs solver was used. Linear kernel and one-vs-rest classification was selected due to the number of classes being 225. The mathematical background for SVM is as follows.

Hypothesis: $y = \mathrm{sgn}\,(w^T x + b)$

Cost function: $J = \frac{1}{2}||w||_2^2 + C\sum_{i=1}^{N} \xi_i$

2.2 Deep Learning: Convolutional Neural Networks

A convolutional neural network (convnet) has hidden layers consisting of convolutional layers, max-pooling layers, fully connected layers and normalization layers. They are most commonly used for tasks such as Image Classification. We implement seven pretrained CNNs offered by Pytorch, namely, densenet121, densenet161, resnet101, resnet101_2, resnet152, squeezenet1.1 and vgg16.

We modified the classifier to our needs by adding extra linear layers and activation functions as well as changed the output to 225 since our data has 225 classes. We experimented with various hyper parameters such as epochs, learning rate, dropout, batch size, optimizers and loss functions. The best output in our experiment was given using an epoch of 75, learning rate of 0.003, dropout of 0.2, batch size of 300, the Adam optimizer. The loss function we decided to use were Cross Entropy Loss and Non-Linear Log Loss (with a Log SoftMax Layer as the activation on the output layer).

The Densenet models out performed the other models, specifically the Densenet 161 which accurately predicted the species 98.2% of the time.

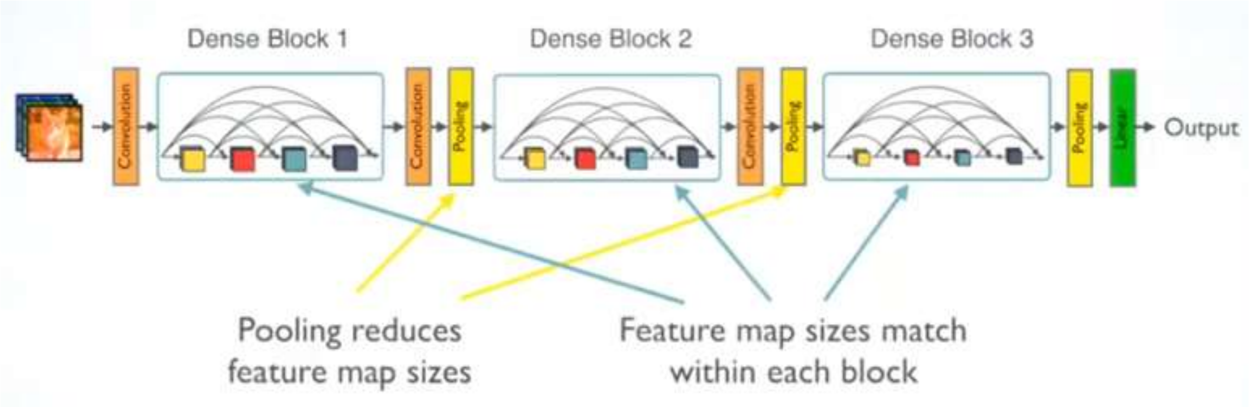


Figure 1: Densenet Architecture

The Densenets tend to perform better due to their unique architecture. All the layers in each dense block are fully connected to each other. This means, the information passes from each layer to each subsequent layer and vice versa (for back propagation). Due to this, each layer is able to learn from all the previous layers resulting in richer features. The size of all convolutional layers in each dense block is the same and goes through another convolutional layer at the end to reduce the input size, thus reducing the number of parameters to learn at each subsequent dense block. At the end of each dense block also lies a pooling layer.

3 Experimental Results

3.1 Dataset

The dataset used for this project was that 225 Bird Species dataset from Kaggle.[4] This dataset contained a total of 33,566 color images of 225 different species of birds. Each image was 224 x 224 x3 in size and was of the JPEG format. The training set had 31,316 images whereas the testing and validation sets each had 1125 images with 5 images per species of bird. Each set contained 225 sub directories, corresponding to each bird species. The training dataset had a slight imbalance, but on average had about 140 images per bird species. The folder names were the class labels for the images. There were no images which were common between the training, test and validation sets. The images were cropped so that the bird occupies at least 50% of the pixels in the image. The cropping ensures that when the images are processed by a CNN, there is adequate information in the images to create a highly accurate classifier.

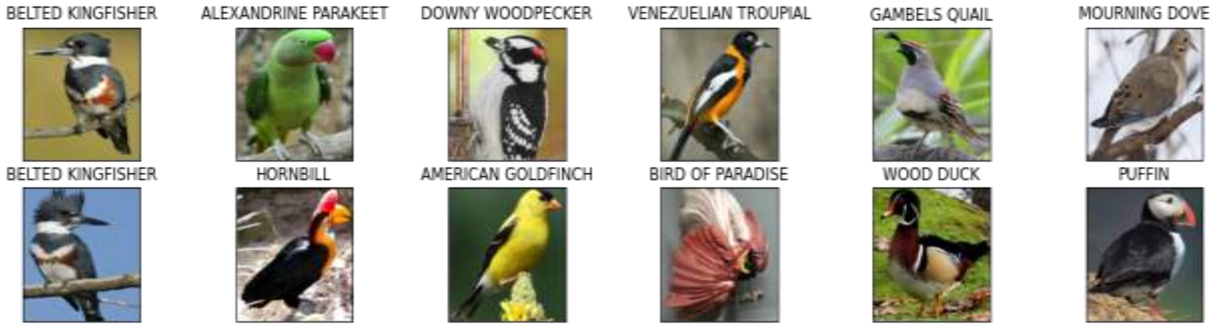


Figure 2: Sample of the dataset

3.2 Data Preparation for Traditional Supervised Learning Models

For the traditional supervised learning models such as SVM and logistic regression, the first step was to extract features in order to obtain feature vectors which could be used to train the SVM and logistic regression classification models. For feature extraction, we implemented three different techniques.

3.2.1 Technique 1: Flattening the RGB Matrix

In this technique the red, green, and blue values for each given pixel of each image was flattened into a single long feature vector. Since each image was of size 224 x 224 x 3, the length of the feature vector for each image was 150,528. Since we had 33,566 images in total, this technique led to 5,052,622,848 data points. The number of data points were too large and hence the SVM and logistic regression model could not be trained due to the huge size of the resulting training set.

3.2.2 Technique 2: Global Feature Extraction

In this technique, global features were manually extracted from each image. These global features included Haralick Texture, Color Histogram, and Hu Moments.

3.2.2.1 Haralick Texture

Image texture is a quantification of grey tone values. Haralick et al. suggested the use of grey level co-occurrence matrices or GLCM.[5] This method is based on the joint probability distributions of pairs of pixels. GLCM show how often each grey level occurs at a particular pixel which is located at a fixed geometric position relative to every other pixel, as a function of the gray level.[6] Extracting the Haralick texture gave us a feature vector of 13 features.

3.2.2.2 Hu Moments

Image moments are weighted averages of image pixel intensities. All 7 Hu Moments are invariant under translations (move in x or y direction), scale and rotation. If one shape is the mirror image of the other, the seventh Hu Moment flips in sign [7][8] Hu Moments quantifies the shape of the image.

3.2.2.3 Color Histogram

A color histogram is a representation of the distribution of colors in an image. It helps us quantify the color of each image. Each image gave us an 8 x 8 x 8 matrix, which was flattened to give a feature vector of 512 features.

3.2.2.4 Combining the Global Features

After extracting the Haralick textures, Hu moments, and color histograms, they were combined into one feature vector with a combined length of 532 features for each image. Using global features, we now had 17,857,112 data points. Using these feature vectors, the SVM and logistic regression models were trained in reasonable amounts of time.

3.2.3 Technique 3: Feature extraction using VGG16

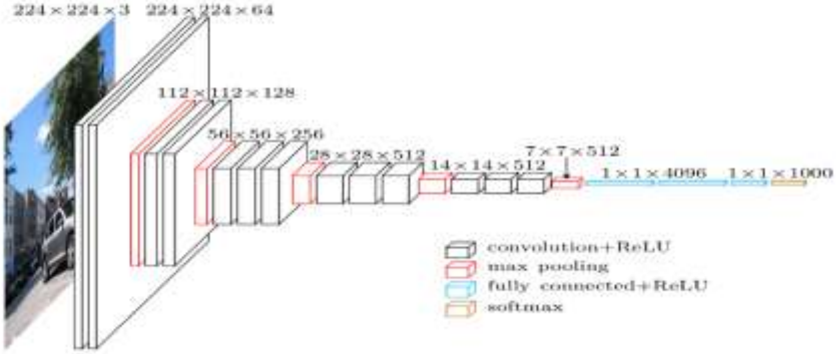
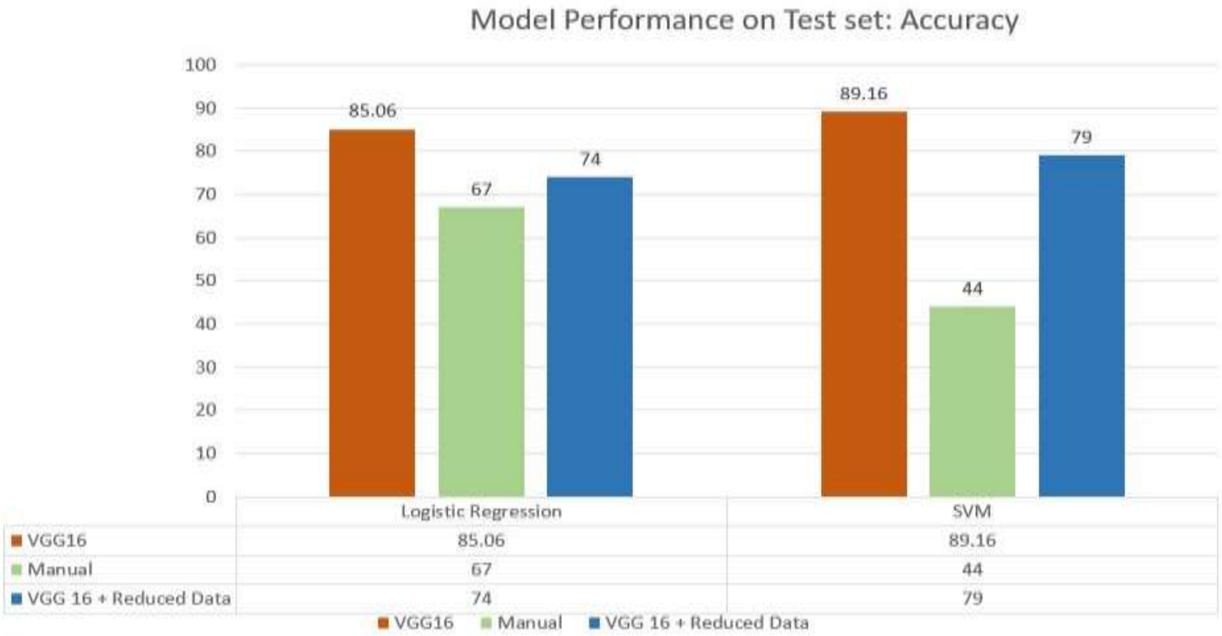


Figure 3: VGG16 Architecture

Convolutional Neural Networks, such as the VGG16 can be used for both prediction as well as feature extraction. VGG16 was used to win the ImageNet Large Scale Visual Recognition Challenge in 2014. Figure 3 shows the architecture of the VGG16 network. The network has convolutional layers of 3x3 filter with a stride of 1 and uses same padding. It has max pool layers of 2x2 filter with a stride of 2. The 16 in VGG16 refers to the 16 layers thar have weights in the network. The VGG16 model was loaded with pretrained weights from the ImageNet dataset. The layer which is the input to the final max pooling layer of the network is the feature extraction part of the model. It gives us a matrix of size 7 x 7 x 512, which is flattened to give a feature vector of size 25,088 features per image. Using these feature vectors, we had a total of 842,103,808 data points which was a large number, but it allowed us to train the SVM and logistic regression models.

3.3 Results



	Logistic Regression	SVM
🟧 VGG16	85.06	89.16
🟩 Manual	67	44
🟦 VGG 16 + Reduced Data	74	79

Figure 4: Performance of Traditional Models

We trained our traditional models on the data with features extracted using VGG16, a subset of that dataset which had 50 images in each class and on the data with manually extracted features explained in section 3.2.2. SVM performed better than Logistic Regression with an accuracy of 89.16% and 85.06% respectively as shown in Figure 4.

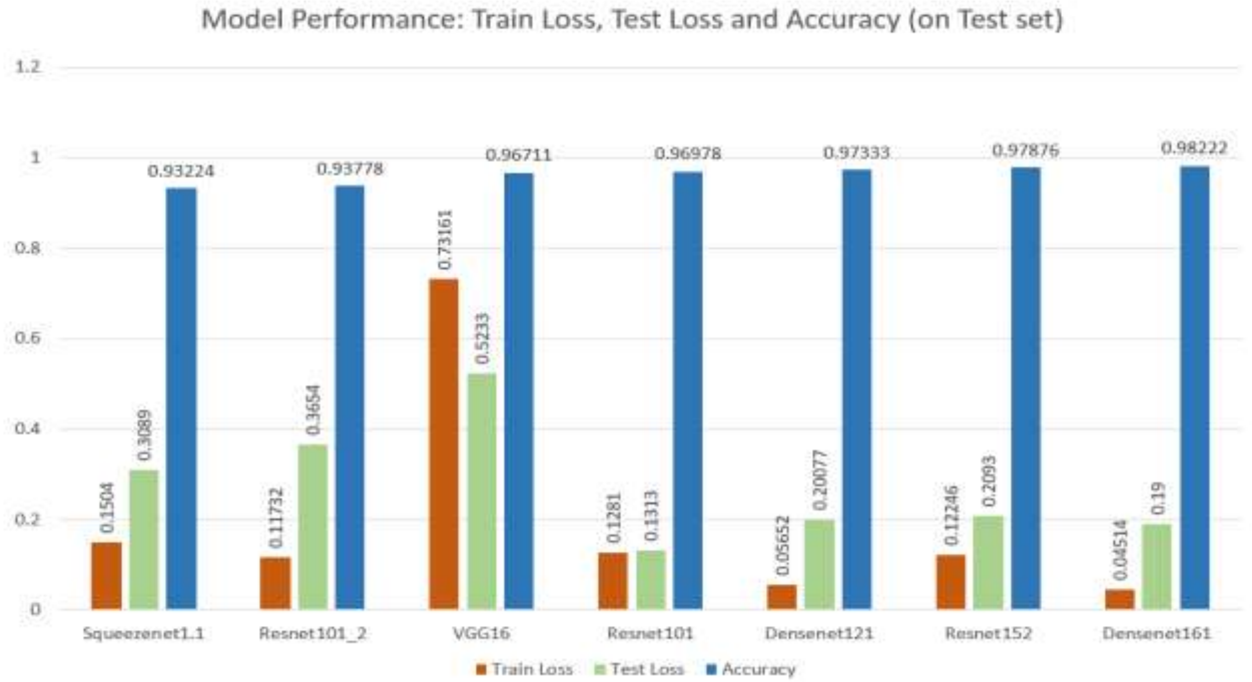


Figure 5: Performance of CNNs

As we can see from figure 5, densenet161 performed the best our of all the models having the lowest train loss, test loss and the highest accuracy. Although VGG16 had a decent accuracy, it had the highest train and test loss making it one of the lower performing models for the task.

Figure 6 depicts the information shown above for each epoch.

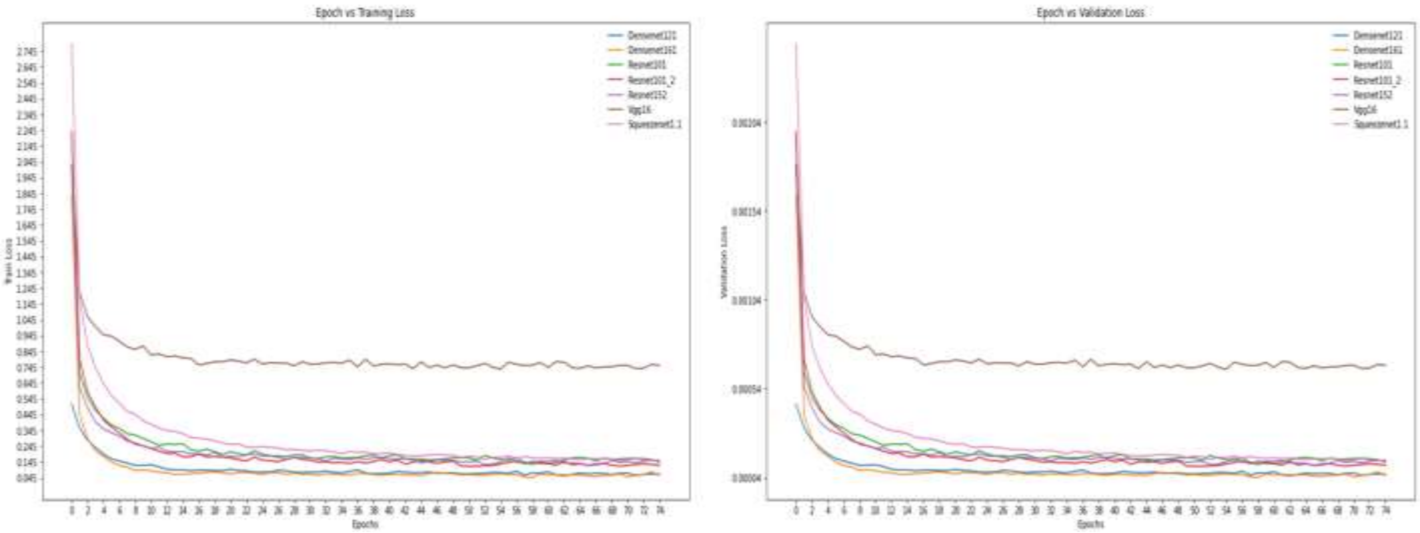


Figure 6: Train and Validation Loss

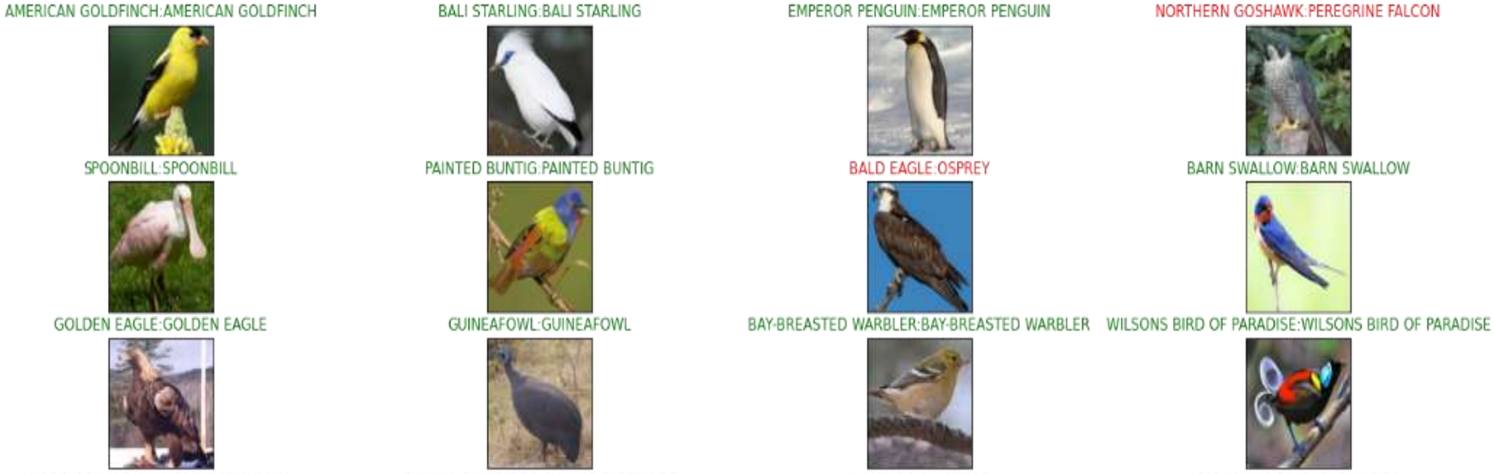

AMERICAN GOLDFINCH:AMERICAN GOLDFINCH
BALI STARLING:BALI STARLING
EMPEROR PENGUIN:EMPEROR PENGUIN
NORTHERN GOSHAWK:PEREGRINE FALCON
SPOONBILL:SPOONBILL
PAINTED BUNTIG:PAINTED BUNTIG
BALD EAGLE:OSPREY
BARN SWALLOW:BARN SWALLOW
GOLDEN EAGLE:GOLDEN EAGLE
GUINEAFOWL:GUINEAFOWL
BAY-BREASTED WARBLER:BAY-BREASTED WARBLER
WILSONS BIRD OF PARADISE:WILSONS BIRD OF PARADISE

Figure 7: Predicted Outputs (Predicted : Ground Truth). Correct predictions are green otherwise red.

Conclusion and Future Work

I. CNNs performed better than any other approach we took for the fine-grained classification of bird species since they are known to learn richer features making them better image classifier.
II. Denser networks do not always necessarily perform better as we see the difference between Densenet and Resnet.
III. Global feature extraction wasn't able to extract good features, whereas VGG16 gave much better extracted features.
IV. The worst performing densenet performed much better than traditional models with VGG16 extracted features.
V. We attempted hyperparameter tuning with GridSearchCV for traditional models with VGG16 features, but due to memory and compute power limitations weren't able to finish with the training.
VI. We also implemented SIFT using OpenCV, but weren't able to get any results with it as it was computationally very heavy. We plan to use spark for distributed computing.
VII. We also plan to isolate the best performing model and use it as the engine for a bird watching application.

Participants Contribution

Omkar implemented manual global feature extraction and trained various traditional models with hyperparameter tuning. Also implemented SIFT and bag of visual words to use a combination of local and global features for training traditional models with their hyperparameter tuning. Arsh implemented the CNN models such as densenets, resenets, VGG16, squeezenet etc. Experimented with different hyperparameters, classifiers and visualized outputs pertaining to the CNNs. Siddarth implemented feature extraction using the VGG 16 models as well as manual feature extraction by flattening the RGB matrices of the images. Also responsible for training, testing and validation of the Logistic Regression and Support Vector Machine models. Attempted GridSearchCV on both models, but training could not be completed due to computation limitations.

REFERENCES

[1] - U.S Fish & Wildlife Service, “2016 National Survey of Fishing, Hunting, and Wildlife-Associated Recreation”, 2016
[2] - Satyam Raj , Saiaditya Garyali , Sanu Kumar , Sushila Shidnal, “Image based Bird Species Identification using Convolutional Neural Network”, June 2020
[3] - https://machinelearningmedium.com/2017/09/15/regularized-logistic-regression/
[4] - https://www.kaggle.com/gpiosenka/100-bird-species
[5] - Harlick et al. 1973
[6] - Srinivasan and Shobha 2008
[7] - https://www.learnopencv.com/shape-matching-using-hu-moments-c-python/
[8] - https://www.researchgate.net/publication/
/224146066_Analysis_of_Hu's_moment_invariants_on_image_scaling_and_rotation

GitHub Repo: https://github.com/arshmodak/Image-Classification-of-225-Aves-Species