Identification of Bird Species by Image Classification

Alan Gaugler
Faculty of Science and Technology
University of Canberra
U885853@uni.canberra.edu.au

Abstract**-- The accurate identification of species is important for all forms of biological, ecological and evolutionary research. Many research projects require accurate identification of species including: population monitoring, studying biodiversity of environmental habitats and the impact of climate change on species distribution. There is an abundance of images online today of birds that can be used in datasets to train machine learning algorithms to classify the species. This project developed various models utilising three machine learning algorithms, Random Forest, SVM and CNN to classify the images of 10 species of birds. Various methods were applied to the models and the dataset including standardization, cross-validation with hyperparameter tuning in a grid search and feature reduction using PCA. Overfitting was present in the training of all models. The best results achieved were with a CNN model with an accuracy of 83% on the test set. The CNN model proved to be more accurate than the best random forest (64%) and the SVM (69%). The dataset will be increased in future work and the models will be developed further.**

Keywords: Image classification, Random Forest, Support Vector Machine, Convolutional Neural Network.

I. INTRODUCTION

This project is the identification of bird species from images of birds captured in their natural habitat using pattern recognition and machine learning (PRML) algorithms. The models were trained to classify 10 species of birds. Images of birds captured by cameras or smartphones can be uploaded and identified by the algorithms.

The accurate identification of species is important for all forms of biological, ecological and evolutionary research [1]. Many research projects require accurate identification of species including: population monitoring of endangered species, studying biodiversity of environmental habitats and the impact of climate change, habitat loss and deforestation on species distribution and migration patterns. Additionally, bird watching is a popular pastime for many amateurs and accurately identifying new species from field guides can be an arduous task, especially when travelling to new regions with different species.

With the vast number of digital images available online today taken from cameras and smartphones, there is an abundance of images of all bird species that can be used in datasets to train PRML algorithms to classify bird species. With the advancement of PRML algorithms, it can be faster and more accurate to identify bird species through the use of applications utilizing properly trained PRML models than by manual methods [2]. Two of the most important criteria for image classification is the selection of a good dataset and the machine learning algorithm.

II DATASET

The most comprehensive dataset on bird images that was found is “BIRDS 400 - SPECIES IMAGE CLASSIFICATION” on Kaggle [3]. It consists of 400 bird species and contains a total of 58388 training images, 2000 test images and 2000 validation images (5 per species). Due to processing power and time constraints, the dataset for this project was reduced down to 10 species. The total number of images used in the training set was 1475, giving a mean of 147.5 per species, ranging from a minimum of 131 to a maximum of 170.

The images are full colour consisting of a width of 224 pixels and a height of 224 pixels giving a total resolution of 150528 pixels. It was ensured that all images consisted of one bird only that took up at least 50% of the image area.

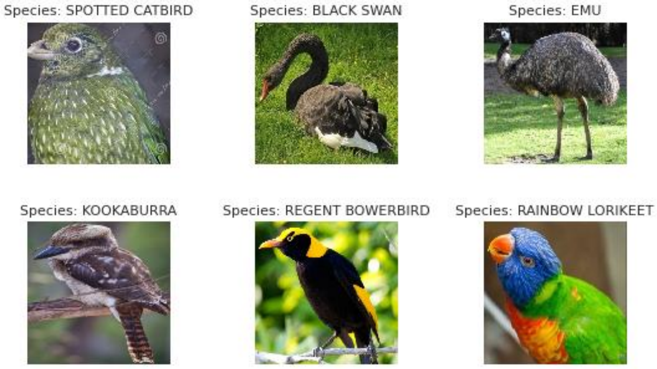


Figure 1 - Sample of Images in the Dataset

The reduced dataset was closely examined to ensure all images were correct. 85% of images were male and 15% female. In many species, their plumage is very different and so it was decided not to use these species in the final dataset as this would lead to higher errors in predicting these classifications. Additionally, there were several examples of misclassified species in the dataset and so those images were removed. In total

3.7% of images were removed due to the above reasons. In many species, this resulted in the removal of several images. It was desired to have at least 130 images per species and so if there were fewer, additional images were obtained online. They were cropped and resized to ensure that the image was in the right format of 224 x 224 pixels with at least 50% of the area taken by the bird.

III METHODOLOGY

A – Evaluation of Classifiers.

For the bird species dataset, a test run on 5 common machine learning models was carried out using sklearn on a subset of 5 bird species. The data was split into 80% training and 20% test. Figure 2 below shows the accuracy score of the models on the dataset. Out of the 5 models, the SVM (Support Vector Machine), Random Forest Classifier, and Logistic Regression algorithms performed more accurately at classifying the bird species with accuracy scores of 81.25%, 80.20%, and 77.60%, respectively. On the other end, the Multilayer Perceptron Classifier (MPL) and the Decision Tree Classifier performed poorly on this dataset. Their accuracy scores were below 60%. It was decided to use the two best of these algorithms for this project.

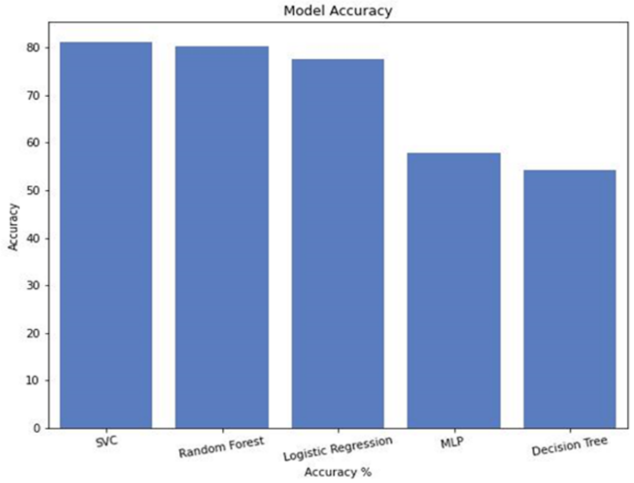


Figure 2 - Model Prediction Accuracy on the Sample Dataset

The plot above confirms the SVM, Random Forest and Logistic Regression models have much higher accuracy than the MLP and Logistic Regression models.

A model that was not evaluated in the sklearn evaluation but has been widely used in image classification of wildlife with high success is a Convolutional Neural Network (CNN). Many research papers [4], [5] have lauded this model for its high accuracy in image classification.

To summarise, it was decided to use the Random Forest Classifier and SVM classifier from Scikit-learn and also Tensorflow's CNN algorithm in the classification of the bird species.

B – Building and training the models.

To carry out the machine learning project the following steps were applied:

- The image data was collected and prepared as described previously.
- The dataset was loaded. It was ensured that the data was valid, the labels were set correctly and the data was shaped appropriately for the various classifiers.
- The data was split into training and test sets. In most cases a train-test split of 80/20% was used. Model performance on a 70/30% split was also evaluated.
- The algorithms were fit with the training data and predictions were run on the test set. The performance of the models was evaluated on their accuracy at predicting images on the test set.
- Several different scenarios were applied to the three different models which will be described below.
- The final, best performing model was applied to an unseen validation set of 50 images.

Three very different classifier algorithms were modelled and tested in this project. Each algorithm had a different set of procedures applied to its testing method.

1. Random Forest.

Decision trees can be susceptible to overfitting the data as they can go down several nodes and can start modelling the random noise. They are also susceptible to instability. Even a small change in the dataset could produce a remarkably different tree. This could lead to a big variation in accuracy of predicting the test or actual data. A random forest constructs multiple decision trees utilizing different subsets of the training dataset and its features for each tree in the 'random forest' [6]. The decisions of each tree are counted and the result with the highest count will be the decision that the forest makes. SKLearn's random forest model is very similar to a decision tree model and it will be evaluated here.

RF1. A random forest classifier (RF) was created with its default settings being used including the number of estimators (trees in the forest) of 100 and a criterion of 'gini'.

RF2. Another RF model was created but with cross validation and a grid search to tune important hyperparameters.

There is no definite way to predetermine which combination of hyperparameters will produce the most accurate model for a particular dataset. For that reason, a grid search with a set of hyperparameter settings was created. The grid-search models the data with every combination of the hyperparameter settings. It will determine which combination of hyperparameters models the data most accurately. These parameters setting will be applied to the model.

Cross-validation of 5 folds and 2 repeats were applied. Cross-validation is an important step that checks the capability of the model to predict new data. It can also detect problems with high variance or overfitting. Repeats were kept at 2 to reduce processing time.

The best estimator from the grid search was applied to the model for testing.

RF3. Another random forest model with a large number of trees was tested to see if there would be improvement in modelling accuracy.

2. Support Vector Machine.

SVMs are a good choice for classification of complex models such as images and for small to medium sized datasets [7] as this project is. They are powerful and are a very popular choice because of their ability to deal with high dimensionality of features [8].

SVM1. A default support vector machine classifier (SVM) was created with its default settings including a kernel of ‘rbf’ (radial basis function) and ‘C’ of 1.

SVM2. A default support vector machine classifier (SVM) was created with its default settings as SVM1. The test and training sets had standardization applied to them before being used in SVM2.

Standardization is a feature scaling process of subtracting the mean value from each feature and then dividing the difference by the feature's standard deviation [9]. Distance algorithms including SVM are adversely affected by the difference in the range of features [10]. With standardization, all features will have a mean value of 0 and a standard deviation of 1, so all will be treated with equal weight by the SVM. Random Forest Classifiers are not distance based, so standardization will have no impact on their performance.

SVM3. Builds on SVM2 with a standardized dataset. It will have its hyperparameters tuned in a grid search with cross validation applied as was the case in model RF2.

The grid search had an extensive list of hyperparameters to be tuned, including the important parameters of ‘C’, ‘gamma’ and ‘kernel’. The best estimator from the grid search will be applied to the model for testing.

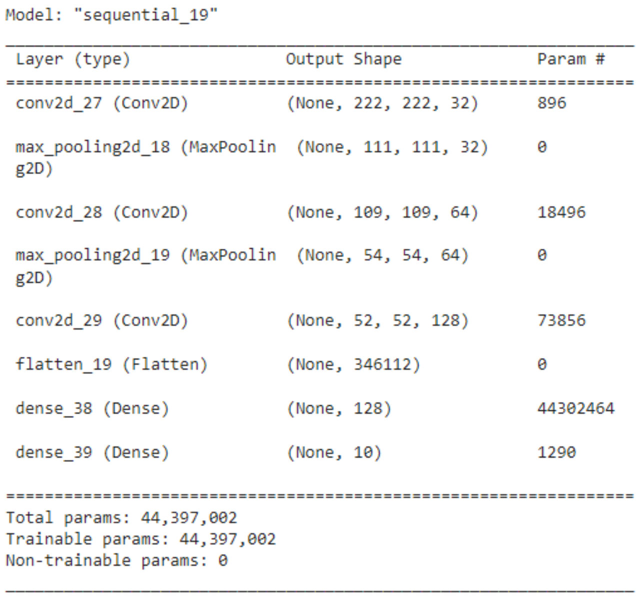
SVM4. Builds on SVM3 with Principal Component Analysis (PCA) applied.

PCA is a common method used to speed up the machine learning algorithms by reducing the number of dimensions (or features) in the dataset. It will retain the features that account for the majority of the variance in the model. This will save a lot of processing time with a very minimal impact on the model’s performance. This dataset has 150528 features and so it will be evaluated to determine if PCA can reduce the processing time significantly without affecting the accuracy.

SVM5 or RF4. The best performing model of the previous seven will be tested on a dataset with a 70/30 training-test split to observe if there is any improvement in its performance compared to an 80/20 split.

3. Convolutional Neural Network.

A layered CNN model was built with the architecture shown in Figure 3.



```
Model: "sequential_19"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_27 (Conv2D)          (None, 222, 222, 32)      896

 max_pooling2d_18 (MaxPoolin  (None, 111, 111, 32)     0
 g2D)

 conv2d_28 (Conv2D)          (None, 109, 109, 64)      18496

 max_pooling2d_19 (MaxPoolin  (None, 54, 54, 64)       0
 g2D)

 conv2d_29 (Conv2D)          (None, 52, 52, 128)       73856

 flatten_19 (Flatten)        (None, 346112)            0

 dense_38 (Dense)            (None, 128)               44302464

 dense_39 (Dense)            (None, 10)                1290

=================================================================
Total params: 44,397,002
Trainable params: 44,397,002
Non-trainable params: 0
_________________________________________________________________
```

Figure 3 - Architecture of the CNN Built for Image Classification

The input shape was set up for the image size of 224 x 224 x 3. The final ‘softmax’ dense layer was set up to have 10 outputs for the 10 classifications. This is a comprehensive model with a total of 44,397,002 parameters.

The Convolutional Layer uses a set of filters which will detect specific features on the image by taking the dot product of the image with the filter. These filters are “slid” along the image and will create filter maps. The activation function is a non-linear transformation performed on the input data. The transformed output is used as input by the next layer. An activation of ‘relu’ (Rectified Linear Unit) was mainly used in this project It is the most common for image classification. The main advantage of ReLU is that it does not activate all neurons simultaneously. All negative inputs are set to zero and so are not activated. This drastically reduces processing time, by up to a factor of 6 compared to other activation functions [11]. Different activation functions were also tried.

The Pooling Layers are located between the convolutional layers. They reduce the number of parameters in the model which will reduce processing time and control overfitting. Max Pooling takes only the maximum values by sliding filters across the input. In every step, only the maximum parameter is kept.

Three convolutional layers were used in this model. In each case the number of filters were doubled from 32 to 64 to 128.

The next layer is Flatten which converts the resultant two-dimensional output from the pooling layer into a linear vector. This is done because the dense layer requires a one-dimensional input.

Finally, the Dense Layer receives the output of all the neurons from the previous flatten layer. The Dense layer will classify the image after being processed in the previous layers. In this project, there are 10 classifications. The final 10 neurons will calculate the probability that the image is of their class. The class with the highest probability value will be chosen as the classification.

The standard dataset was applied to this CNN model and its performance was evaluated. In future work, an augmented dataset will be applied to the CNN model. Network architecture will also be changed.

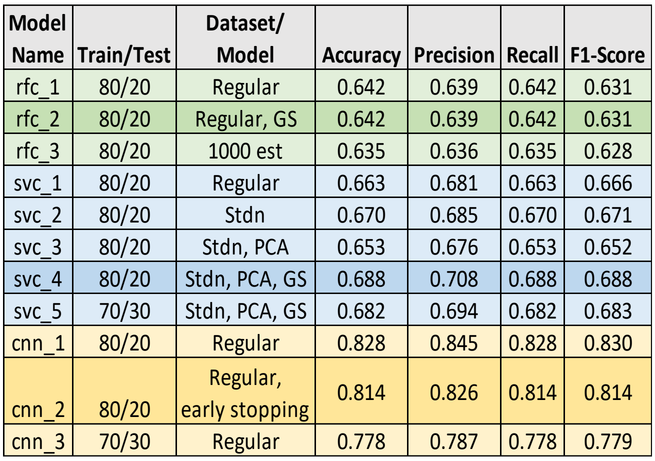
C - Evaluate the final model on unseen data.

The final model that proved to be the most accurate was evaluated on a validation set that was not used in the models' training. This will give an independent and true test of the results.

IV RESULTS AND EVALUATION

Over 11 models were evaluated for the three classifiers including various CNN configurations. In each successive model, one additional procedure was applied to observe what impact it alone would have on the results. The results are summarised in Table 1. The following abbreviations were used.

Stdn – Standardization
PCA – Principal Component Analysis
GS – Grid search with hyperparameter tuning.
1000 est – 1000 estimators in the random forest.



Model Name	Train/Test	Dataset/ Model	Accuracy	Precision	Recall	F1-Score
rfc_1	80/20	Regular	0.642	0.639	0.642	0.631
rfc_2	80/20	Regular, GS	0.642	0.639	0.642	0.631
rfc_3	80/20	1000 est	0.635	0.636	0.635	0.628
svc_1	80/20	Regular	0.663	0.681	0.663	0.666
svc_2	80/20	Stdn	0.670	0.685	0.670	0.671
svc_3	80/20	Stdn, PCA	0.653	0.676	0.653	0.652
svc_4	80/20	Stdn, PCA, GS	0.688	0.708	0.688	0.688
svc_5	70/30	Stdn, PCA, GS	0.682	0.694	0.682	0.683
cnn_1	80/20	Regular	0.828	0.845	0.828	0.830
cnn_2	80/20	Regular, early stopping	0.814	0.826	0.814	0.814
cnn_3	70/30	Regular	0.778	0.787	0.778	0.779

Table 1 - Summary of Accuracy Results of Evaluated Models

Table 1 shows that the random forest and support vector classifiers were predicting with an accuracy ranging from 63.5% to 68.8%. The best CNN model eclipsed that with a score of 81.4%. This is a considerable improvement of 12.6%.

It was observed that applying a grid search with hyperparameters did not improve random forest performance. The grid search was limited because of processing time constraints. With a more powerful computer available, the grid search would have been expanded. Standardization was applied to the SVM model with only a slight improvement. With PCA and hyperparameter tuning through a grid search, an improvement of 2.5% over the original SVC was observed. The PCA was set to maintain 99.5% of this variance. This resulted in a reduction of 150528 image components down to 1026.

In both the SVC and the CNN models, applying a train/test ratio of 70/30 slightly decreased the classification accuracy.

In all cases of the RFC and SVM models, high accuracies were achieved on the training set (always above 0.9) and much lower accuracies were produced in the cross validation and on the test set. (Test set results are in Table 1). This is a strong indication that the model is overfitting the image data. This is a common machine learning problem. Each successive model in this study was built on the previous model and although the overfitting was reduced, it was still present.

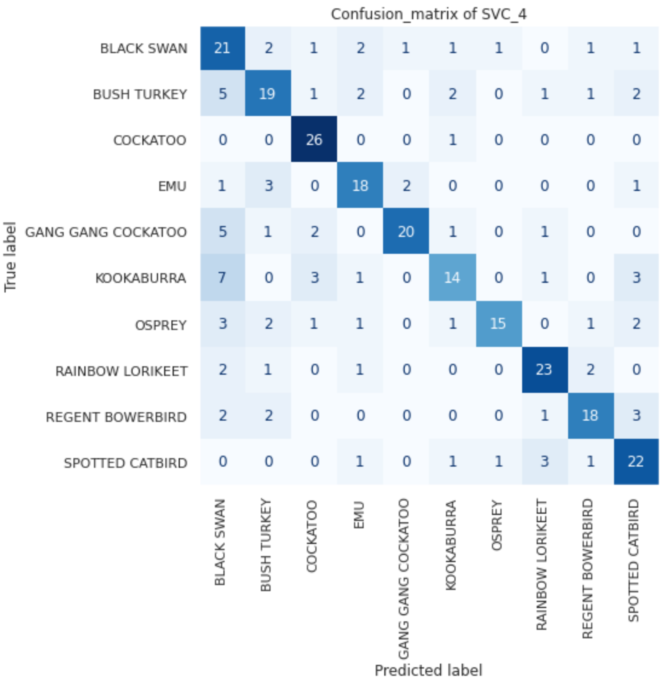


Figure 4 - Confusion Matrix Example of Classification

Figure 4 shows an example of the confusion matrix which was used to evaluate the accuracy, precision, recall and F1-score of the models. In this example of SVC_4, the highest errors produced were for an image of a Bush Turkey, Kookaburra or a Gang Gang Cockatoo to be misclassified as a Black Swan (1st column). This contributed to a poor Precision Score of 0.46.

The 8 layered CNN architecture shown in Figure 3 provided an accuracy of 80.3% on the test set. Various other configurations were tried. The architecture was slightly modified by going deeper with additional layers. The results were very similar but with a considerably longer processing time.

Other parameters were also tuned with various options for optimizer (SGD, Momentum, AdaMax) and for Activation (Sigmoid, tanh, Leaky ReLU). The tuning of these parameters did not result in higher accuracy.

Overfitting was also observed in the CNN models with a training accuracy mostly close to 1.0 and the resulting validation accuracy of 0.828.

To avoid the model from overfitting through several epochs, early stopping was implemented. Early stopping is a regularization technique that will cause the model training to stop once there is no longer any improvement in the validation set. Looking at the plots below in Figures 5 and 6, it can be seen that without early stopping (Figure 5) the model's accuracy levels off at 0.83, however the model loss increases, indicating that overfitting is increasing.

Figure 6 shows the results of the same model being fit with early stopping implemented. After only 6 epochs, a similar accuracy on the validation set was reached when early stopping halted the model fitting. The model loss here is at a minimum and so overfitting will not increase.

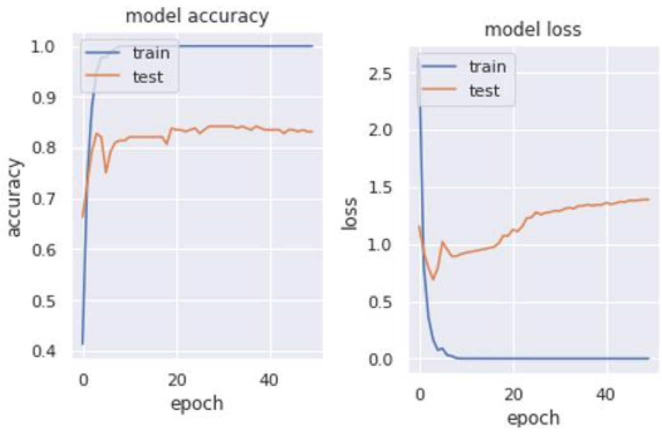


Figure 5 – Model Accuracy and Model Loss without Early Stopping

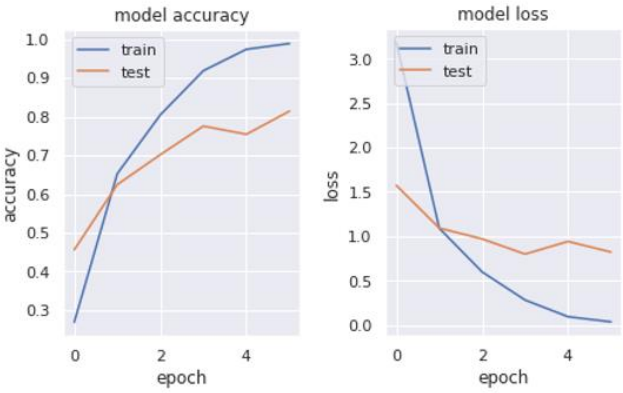


Figure 6 – Model Accuracy and Model Loss with Early Stopping

The CNN model was clearly the most accurate of all the models that were developed. This was the final model chosen.

The final model was used to predict on an unseen set of 50 images, 5 of each classification. The result was 80% accuracy which is in line with the test set accuracy of 81.4%. This confirms that the model holds up well on independent data.

V ETHICAL CONCERNS

There are not any major ethical concerns with image classification of birds. Taking images of humans and identifying them through algorithms has several ethical concerns, in particular invasion of privacy. However, these do not apply to birds as it will not affect them at all. If anything, this application will aid in their conservation as it would mainly be used by researchers in the introduction. What the photographer must ensure however, is that approaching birds (or other animals in their habitat is done ethically without causing any harm to the birds or the environment. In particular endangered species should not be approached, especially where they are nesting as this could result in them abandoning their nesting grounds for fear of their own safety.

VI. CONCLUSION & FUTURE WORK

Three classifier algorithms were evaluated and tested with various setups applied to their configurations and to the dataset of 10 bird species. It can be concluded that the CNN algorithm with an 8 layered architecture was clearly the most accurate of all the models. Its accuracy was 81.4% on the test set and 80.0% on the validation set. The accuracy of the best Random Forest Classifier was 64.2% and for the best Support Vector Machine it was 68.8%.

It was found that with this dataset, there were issues with overfitting. Every model performed very well on the training set and in each case the score on the test set was considerably lower. Various methods were applied to the dataset and the models to reduce overfitting including standardization, cross validation with hyperparameter tuning in a grid search and feature reduction using principal component analysis. It was improved but not eliminated.

Further work that can be carried out on this project include:

- Data augmentation can be used in the short term to increase the size of the dataset by slightly modifying existing images and adding them to the training set.
- Increase the size of the dataset through collecting more accurate images. The acquisition of good quality data is the best approach to training a more accurate model. This will also reduce the level of overfitting.
- Try various reconfigurations of the architecture of the CNN model.
- Try pre-trained neural network architectures such as ResNet-50 on this dataset.
- With the acquisition of more quality images, the scope of this project will be expanded to classify many more species of birds in the local region as well as differentiate between male and female of the species.

REFERENCES

[1] J. Wäldchen and P. Mäder, “Machine learning for image based species identification,” Methods in Ecology and Evolution, vol. 9, no. 11, pp. 2216–2225, Sep. 2018, doi: 10.1111/2041-210x.13075.

[2] [2] - Huang, Y.-P. and Basanta, H. (2019). Bird Image Retrieval and Recognition Using a Deep Learning Platform. IEEE Access, 7, pp.66980–66989. doi:10.1109/access.2019.2918274.

[3] Piosenka, G. (n.d.). BIRDS 400 - SPECIES IMAGE CLASSIFICATION. [online] Available at: https://www.kaggle.com/datasets/gpiosenka/100-bird-species.

[4] Norouzzadeh, M. S. et al. Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning. Proc. Natl. Acad. Sci., https://doi.org/10.1073/pnas.1719367115 http://www.pnas.org/content/early/2018/06/04/1719367115.full.pdf (2018).

[5] Miao, Z., Gaynor, K.M., Wang, J. et al. Insights and approaches using deep learning to classify wildlife. Sci Rep 9, 8137 (2019). https://doi.org/10.1038/s41598-019-44565-w

[6] Simic, M. (2022). Decision Trees vs. Random Forests | Baeldung on Computer Science. [online] www.baeldung.com. Available at: https://www.baeldung.com/cs/decision-trees-vs-random-forests.

[7] A. Lamidi, “Breast Cancer Classification Using Support Vector Machine (SVM),” Medium, Nov. 27, 2018. https://towardsdatascience.com/breast-cancer-classification-using-support-vector-machine-svm-a510907d4878

[8] K. Sanghvi, “Image Classification Techniques,” Medium, Sep. 25, 2020. https://medium.com/analytics-vidhya/image-classification-techniques-83fd87011cac

[9] EliteDataScience. (2017). Python Machine Learning Tutorial, Scikit-Learn: Wine Snob Edition. [online] Available at: https://elitedatascience.com/python-machine-learning-tutorial-scikit-learn.

[10] Bhandari, A. (2020a). Feature Scaling | Standardization Vs Normalization. [online] Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2020/04/feature-scaling-machine-learning-normalization-standardization/.

[11] U. Udofia, “Basic Overview of Convolutional Neural Network (CNN),” Medium, Sep. 20, 2019. https://medium.com/dataseries/basic-overview-of-convolutional-neural-network-cnn-4fcc7dbb4f17#:~:text=The%20activation%20function%20is%20a