**A Project Report On**

**BIRD SPECIES CLASSIFICATION**

**Prepared By**

**Shashi Prakash Shah (16900116046)**

**Sayan Sarkar (16900116047)**

**Riteshnu Ghosh (16900116062)**

**Prithwish Sarkar (16900116072)**

**Srijon Rej (16900116029)**

**Under the guidance of**

**Prof. Arindrajit Pal**

**A Project Report**

**to be submitted in the partial fulfillment of the requirements**

**for the degree of**

**Bachelor of Technology in Computer Science and Engineering**



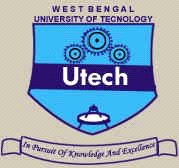
**Department of Computer Science and Engineering**

**Academy Of Technology**

**Aedconagar, Adisaptagram,**

**Hooghly- 712121, West Bengal**

Affiliated to



**Maulana Abul Kalam Azad University of Technology,**

**West Bengal.**

May, 2020

**Academy Of Technology**



**CERTIFICATE**

This is to certify that the project entitled “Bird Species Classification” submitted to MAULANA ABUL KALAM AZAD UNIVERSITY OF TECHNOLOGY in the partial fulfillment of the requirement for the award of the B.TECH degree in COMPUTER SCIENCE AND ENGINEERING is original work carried out by

Shashi Prakash Shah (16900116046),

Sayan Sarkar (16900116047),

Riteshnu Ghosh (16900116062),

Prithwish Sarkar (16900116072)

Srijon Rej (16900116029)

under my guidance.

The matter embodied in this project is genuine work done by the student and has not been submitted whether to this University or to any other University/Institute for the fulfillment of the requirement of any course of study.

Prof. Arindrajit Pal

Department of Computer Science and Engineering

Academy of Technology

Aedconagar

Hooghly-712121,

Dated: West Bengal, India

Countersigned By

**Prof. Prasenjit Das**

Head, Department of Computer Science and Engineering.

Academy of Technology,

Aedconagar,Hooghly-712121, West Bengal, India

**STATEMENT BY THE CANDIDATES**

Shashi Prakash Shah Roll 16900116046

Sayan Sarkar Roll 16900116047

Riteshnu Ghosh Roll 16900116062

Prithwish Sarkar Roll 16900116072

Srijon Rej Roll 16900116029

B. Tech 8th Semester

Dept. of Computer Science and Engineering

Academy of Technology

We hereby state that the Project Report entitled “Bird Species Classification” has been prepared by us to fulfill the requirements of **CS892** during the period January 2020 to May 2020.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Shashi Prakash Shah(16900116046)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Sayan Sarkar (16900116047)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Riteshnu Ghosh (16900116062)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prithwish Sarkar (16900116072)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Srijon Rej (16900116029)

**ACKNOWLEDGEMENT**

At the very outset, we would like to convey our sincere gratitude to our beloved founder-chairman **Prof. J Banerjee** and respected director **Prof. D Bhattacharya** for all the encouragement and support extended to us during the tenure of this project and also our years of studies in this institute.

We are indebted to our guide **Prof. Arindrajit Pal** for his/her epitome guidance, assistance and cooperation that facilitated the successful conclusion of our project.

We express our heartfelt thanks to our Head of the Department, **Prof Prasenjit Das**, who has been actively involved and very influential from the start till the completion of our project.

We would also like to thank all teaching and non-teaching staffs of the Computer Science and Engineering Department for their constant support and encouragement given to us. Last but not the least it is our great pleasure to acknowledge the wishes of friends and well-wishers, both in academic and non-academic spheres.

Shashi Prakash Shah Roll 16900116046

Sayan Sarkar Roll 16900116047

Riteshnu Ghosh Roll 16900116062

Prithwish Sarkar Roll 16900116072

Srijon Rej Roll 16900116029

B. Tech 8th Semester

Dept. of Computer Science and Engineering

Academy of Technology

LIST OF FIGURES

**Figures Page No**

1.1.a Peacock vs. Crow 01

1.1.b White Crowned Sparrow vs. Tree Sparrow 01

2.1 Structure of Deep Neural Network 06

2.2 Convolutional Neural Network Layers 09

2.3 VGG16 10

2.4 VGG16 Architecture 10

2.5 Residual Learning : a building block 12

2.6 Different type of Fit Curve 13

4.1 Proposed Model 16

4.2 CUB-200 2011 dataset 17

4.3 Features of thedataset 18

5.1 Structure of Classifier 24

5.2 Accuracy of VGG16 Without pre-trained weights 24

5.3 Accuracy of VGG16 With pre-trained weights 25

5.4 Accuracy of ResNet With pre-trained weights 25

5.5 Structure of Classifier of Convnet-1 26

5.6 Accuracy of Convnet-1 27

6.1 Structure of Classifier of Convnet-2 29

6.2 Accuracy of Convnet-2 30

List of Tables

**Tables Page No**

Table 5.1 Accuracy of Algorithms 27

Table 6.1 Distribution of images in Training, 28

Testing and Validation sets

**Abstract**

The analysis of birds present in an area is of great importance for ornithologists and bird watchers as well as to the study of ecology. Learning about birds in an environment provides useful information regarding environmental changes and to grasp significant information regarding nature. Current methods for recognition of birds require manual classification by ornithologists based on collected visual and audio data. This is a cumbersome process due to some much variations between different species of birds as well as variations within the birds of same species. In this regard, there is a need for an automated system for detecting birds with accuracy, replacing current methods which are time consuming and erroneous. Our project aims to apply state-of-the-art Machine learning algorithms and techniques suited to the classification of birds into its corresponding family with the help of various input features that have been proven to succeed in a similar task in the past. For the project, we will be using the Caltech-UCSD-Birds-200-2011 dataset for training as well as testing purpose. The selected species from the dataset will be trained on a CNN as well as pre-trained CNN based on state-of-the-art Machine learning algorithms with fixed feature extraction using transfer learning.

Table of contents

**Chapter No: Title Page No.**

Certificate ii

List of Figures v

List of Tables vi

Abstract vii

**Chapter 1 Introduction** 01

* 1. Purpose of This Study 02
  2. Brief Overview of the Project Report 03

**Chapter 2 Literature Overview** 04

* 1. Previous Related Work 04
     1. MLSP 04
     2. NIPS4B 05
     3. ICCVIP 05
  2. Deep Neural Network 06
     1. The History of Deep Neural . Network(DNN) 06
     2. Convolution Neural Network 08
  3. Algorithms 09
     1. VGG16 09
     2. Residual Neural Network 11
  4. Cross-validation 12
     1. Underfitting 12
     2. Overfitting 13

**Chapter 3 Problem Definition & Objectives** 14

* 1. Motivation 14
  2. Objectives & Goals 15

**Chapter 4 Design** 16

* 1. Proposed Model 16
  2. Implementation 17
     1. Data collection 17
     2. Data description 18
     3. Pre-processing the dataset 19
     4. Split Dataset 19
     5. Algorithm 19
     6. Software Requirement 20

**Chapter 5 Experimental result and analysis** 23

* 1. VGG16 model without ImageNet weight 23
     1. Structure 23
     2. Training accuracy 24
  2. VGG16 model with ImageNet Weight 24
     1. Structure 24
     2. Training accuracy 25
  3. ResNet-50 model 25
     1. Structure 25
     2. Training accuracy 25
  4. Convnet1 26
     1. Structure 26
     2. Training accuracy 26
  5. Comparison between Algorithms 27

**Chapter 6 Further Studies** 28

* 1. Bird Species Classification For Indian . Subcontinent regions 28
     1. Dataset 28
     2. Algorithm 29
     3. Evaluation 29
  2. Future work 30

**Chapter 7 Conclusion**  31

* 1. Concluding Remarks 31

**Bibliography** 32

**CHAPTER 1**

**INTRODUCTION**

Bird species classification from images is an important and challenging problem with real-world applications like environment protection and endangered animal rescue. Study of birds helps us to evaluate the quality of the living environment and also learn about the different organisms present around us like insects they feed on. Accurate bird recognition is also essential for avian biodiversity conservation. But, gathering and collecting information about birds is tedious, erroneous as well as time-consuming process. Bird Species classification is also an interesting problem in computer vision as it pushes the limits of the visual abilities for both humans and computers. It is difficult for non-professionals to identify the sub-category of a bird only by its appearance as different bird species share the same basic set of part, different bird species can vary dramatically in shape and appearance ( e.g. consider peacock vs. crow). At the same time, other pair of bird species is nearly visually indistinguishable even for expert birdwatchers (e.g., many sparrow species are visually similar).

1.1.a Peacock vs. Crow 3.1.b White Crowned Sparrow vs. Tree Sparrow

Professional bird watchers, park rangers, ecology consultants, and ornithologists all over the world sometimes disagree on the species given an image of a bird. In such a scenario, it is also difficult and exhausting to annotate all the images by human beings with expert knowledge. Thus, an automatic classiﬁcation system for bird species are needed, which will be a great convenience for many practical applications. For researchers working outdoors, pictures taken by bird watchers can be classiﬁed and analyzed immediately by the system, reducing the need for illustration books for bird identification.

This project aims to develop such an automated system to recognize birds from the images by studying different Artificial Neural Networks, which will simplify the identification process of birds. Our project we have taken 20 species under consideration from the “Caltech-UCSD Birds-200-2011 (CUB-200-2011)” dataset containing 11,788 images of 200 bird species. After creating the dataset for the considered bird species viz.

* ‘Black footed Albatross',
* 'Rhinoceros Auklet',
* ‘Rusty Blackbird',
* ‘American Crow',
* ‘Purple Finch',
* 'Yellow bellied Flycatcher',
* 'American Goldfinch',
* 'Blue Grosbeak',
* 'Ruby throated Hummingbird',
* 'Green Jay',
* ‘Gray Kingbird',
* 'Orchard Oriole',
* 'White Pelican',
* 'White necked Raven',
* 'American Redstart',
* ‘White throated Sparrow',
* 'White crowned Sparrow',  '
* ‘Bank Swallow',
* 'Tree Swallow',
* ‘Red eyed Vireo’

for their distinct features, the machine learning model based on convolutional neural network and state-of-the-art Machine Learning Algorithms are trained on above mention dataset by varying different parameters, to find suitable algorithm and parameter for better detection of birds in an image.

* 1. **Purpose of this study**

The primary purpose of this project is to develop a bird recognition system somewhat similar to Merlin Bird ID in the USA, for Indian birds for quick and error-less identification of birds found around the Indian subcontinent, which will help bird watchers and ornithologists to learn about the birds easily and make a better effort towards conservation and preservation of birds’ species.

* 1. **Brief Overview of the Project Report**

In this chapter, we have explained to the reader why the problem we have chosen to have our project, what it can be used for, and have given the reader an introduction and basic idea of the problem at hand. The outline of the rest of the Project Report on “Bird Species Classification” is:

In Chapter 2 we review the results produced during several of the most recent bird species identification/classification challenges and present the theory of the implementation tools used in this project.

In Chapter 3, we stated the motivation behind the selection of this domain for the project along with our Objectives and Goals.

In Chapter 4 and 5, we discussed the proposed Model, and its implementation, with the result obtained from them. In Chapter 5 we analyzed the result.

In Chapter 6, we discussed the future works and the further studies that had been done on this project.

Lastly, we present the concluding remark.

**CHAPTER 2**

**LITERATURE OVERVIEW**

This chapter literature overview deals with the literary works of Convolutional Neural Network, classification algorithms like VGG16, ResNet. This chapter also refers to the previous work on the bird species classification.

* 1. **PREVIOUS RELATED WORK:**

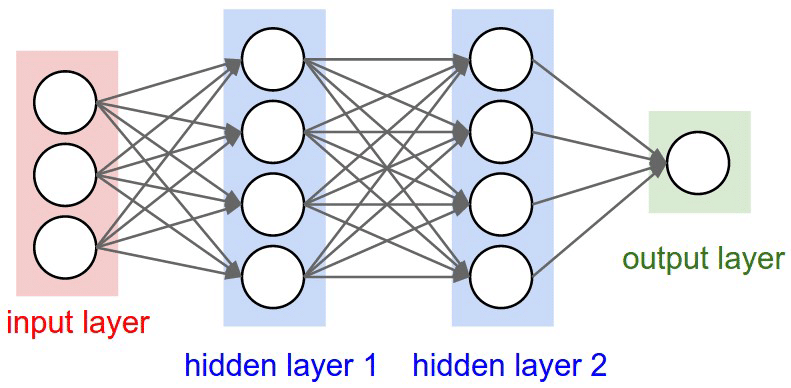
In this section previous work on automated species classification is reviewed where acoustical as well as image classification methods have been used to identify birds from their audio recordings and images. Several bird species classification challenges with closely related, but different, task descriptions have been held during the last few years. The interest and participation in these challenges have been high which indicates that these are relevant problems and that there is a need to solve them. The challenges are usually to predict which species are present in a set of data with hidden labels, called the test set, and to submit the predicted species for each test data point for evaluation against the ground truth labels. The task description can vary from predicting only the presence or absence of birds in an image (or in a recording) to predicting all active bird species. That is, the challenges have a varying degree of difficulty. In the rest of this section, we present the results of some of the most recent such challenges in chronological order

* + 1. **MLSP:** The IEEE International Workshop on Machine Learning for Signal Processing (MLSP) announced a bird species

classification challenge in the year of 2013. The challenge was to determine all of the acoustically active bird species in each audio recording of a test set with a total of 19 different bird species. The data set consisted of 645 ten-second audio recordings which were split into a training set (50%) and a test set (50%). The bird species labels for each recording in the training set was made public, but the labels for each recording in the test set was kept secret. 79 teams participated in the challenge. The winning team used a random forest (RF) classifier, where features were extracted from the input using template matching. Many of the teams designed task-specific features; however, one team which came in fourth place used raw spectrogram data to train a convolutional neural network which signifies the use of convolutional neural networks in this problem domain.

* + 1. **NIPS4B:** In the Neural Information Processing Scaled for Bioacoustics (NIPS4B) Bird Species Classification Challenge, the task description was similar to that of MLSP 2013. Participants were asked to identify all actively singing birds in each of the test files. However, the number of possible species was 87 instead of 19, and the recordings could vary in length (from around 0.5s to 5.5s). This is also formulated as a single-instance multi-label classification problem. The winning solution of this challenge used an approach where additional features are extracted for each audio file, in addition to the features extracted by evaluating the template matching.
    2. **ICCVIP:** In the International Conference on Computer Vision and Image Processing (ICCVIP) Bird Species Classification Challenge held in 2018 the task description was based on image classification. Participants were asked to identify all birds in each of the test files in the provided smaller dataset, which has a larger variation in terms of scale, illumination etc. The data set consisted of 308 images which were split into a training set (150 images) and a test set (158 images). The resolution of the images lies in between 800x600 to 6000x4000. The winning solution of this challenge used Mask Recurrent Convolutional Neural Network with Ensemble Model.
  1. **DEEP NEURAL NETWORK**
     1. The History of Deep Neural Network(DNN)

Computational neurobiology has conducted significant research on constructing computational models of artificial neurons. Artificial neurons, which try to mimic the behaviour of the human brain, are the fundamental component for building Artificial Neural Networks (ANNs). The basic computational element (neuron) is called a node (or unit) which receives inputs from external sources and has some internal parameters (including weights and biases that are learned during training) which produce outputs. This unit is called a perceptron. ANNs or general NNs consist of Multilayer Perceptron’s (MLP) which contain one or more hidden layers with multiple hidden units (neurons) in them.



2.1 Structure of Deep Neural Network

DNN is trained with the popular Back-Propagation (BP) algorithm with Stochastic Gradient Descent or another optimization algorithm. The learning rate is an important component of training DNN. The learning rate is the step size considered during training which makes the training process faster. However, selecting the value of the learning rate is sensitive. For example: if we choose a larger value for learning rate then the network may start diverging instead of converging. On the other hand, if we choose a smaller value for learning rate it will take more time for the network to converge. Also, it may easily get stuck in local minima. The typical solution to this problem is to reduce the learning rate during training.

The history of DNN is as follows:

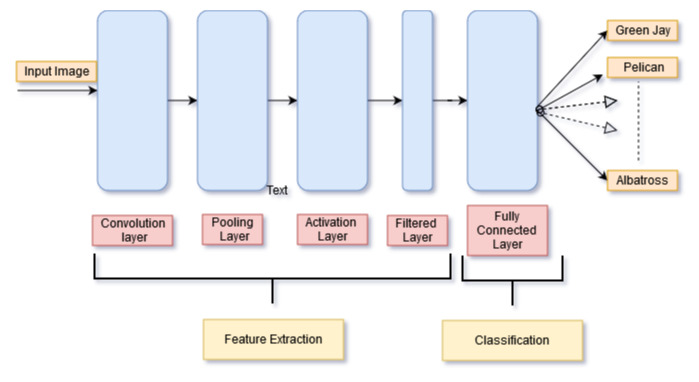
* 1943: Warren McCulloch and Walter Pitts create a computational model for neural networks based on mathematics and algorithms called threshold logic.
* 1958: Frank Rosenblatt creates the perceptron, an algorithm for pattern recognition based on a two-layer computer neural network using simple addition and subtraction
* 1980: Kunihiko Fukushima proposes the Neoconitron, a hierarchical, multilayered artificial neural network that has been used for handwriting recognition and other pattern recognition problems.
* The mid-2000s: The term “deep learning” begins to gain popularity after a paper by Geoffrey Hinton and Ruslan Salakhutdinov showed how a many-layered neural network could be pre-trained one layer at a time.
* 2009: NIPS Workshop on Deep Learning for Speech Recognition discovers that with a large enough data set, the neural networks don’t need pre-training, and the error rates drop significantly.
* 2012: Artificial pattern-recognition algorithms achieve human-level performance on certain tasks.  And Google’s deep learning algorithm discovers cats.
* 2015: [Facebook](https://www.forbes.com/companies/facebook/) puts deep learning technology - called DeepFace - into operations to automatically tag and identify Facebook users in photographs. Algorithms perform superior face recognition tasks using deep networks that take into account 120 million parameters.
* 2016: Google DeepMind’s algorithm AlphaGo masters the art of the complex board game Go and beats the professional Go player Lee Sedol at a highly publicized tournament in Seoul
  + 1. Convolutional Neural Network**:**

In deep learning, a convolutional neural network (CNN) is a class of deep neural network mostly used for analyzing visual images. It consists of an input layer and output layer as well as multiple hidden layers. Every layer is made up of a group of neurons and each layer is fully connected to all neurons of its previous layer. The output layer is responsible for the prediction of output. The convolutional layer takes an image as input and produces a set of feature maps as output. The input image can contain multiple channels such as colour, wings, eyes, the beak of birds which means that the convolutional layer performs a mapping from 3D volume to another 3D volume. 3D volumes considered are width, height, depth.

The CNN has two components:

1. Feature extraction part: features are detected when the network performs a series of convolutional and pooling operation.
2. Classification part: Extracted features are given to fully connected layer which acts as a classifier.

CNN consists of four layers: convolutional layer, activation layer, pooling layer and fully connected. Convolutional layer allows extracting visual features from an image in small amounts. Pooling is used to reduce the number of neurons from previous convolutional layer but maintaining the important information. Activation layer passes a value through a function which compresses values into range. Fully connected layer connects a neuron from one layer to every neuron in another layer.



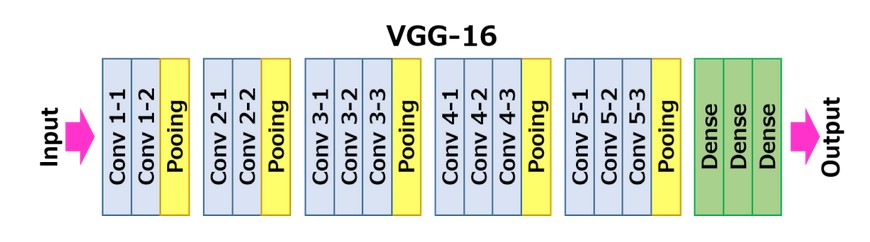
2.2 Convolutional Neural Network Layers

As CNN classifies along each neuron in depth, so it provides more accuracy. Image classification: image classification in machine learning is commonly done in two ways: 1) Grayscale 2) Using RGB values. Normally, all the data is mostly converted into grayscale. In the grayscale algorithm, the computer will assign values to each pixel based on how the value of the pixel is it. All the pixel values are put into an array and the computer will perform operation on that array to classify the data.

* 1. **ALGORITHMS**
     1. VGG16

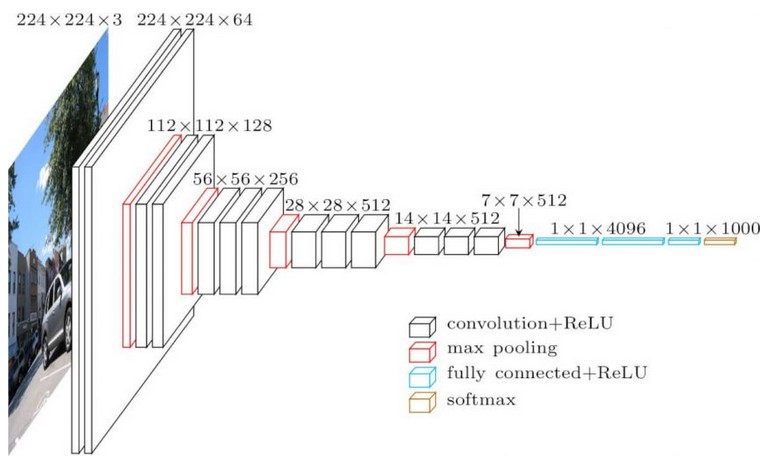
VGG16 is a convolutional neural network model proposed by K. Simonyan and A. Zisserman from the University of Oxford in the paper “Very Deep Convolutional Networks for Large-Scale Image Recognition”. The model achieves 92.7% top-5 test accuracy in ImageNet, which is a dataset of over 14 million images belonging to 1000 classes. It was one of the famous model submitted to ILSVRC-2014. It makes the improvement over AlexNet by replacing large kernel-sized filters (11 and 5 in the first and second convolutional layer, respectively) with multiple 3×3 kernel-sized filters one after another. VGG16 was trained for weeks and was using NVIDIA Titan Black GPU’s.

During training, the input to VGG ConvNets is a fixed-size 224×224RGB image. The only pre-processing done on the input is subtracting the mean RGB value, computed on the training set, from each pixel. The image is passed through a stack of convolutional (conv.)layers, where filters with a very small receptive field:3×3 was used. The size of the receptive field is the smallest size to capture the notion of left/right, up/down, centre).

****

2.3 VGG16

In one of the configurations, 1×1convolution filters were used as a linear transformation of the input channels (followed by non-linearity). The convolution stride was fixed to1 pixel; the spatial padding of Conv. layer input is such that the spatial resolution was preserved after convolution, i.e. the padding is 1pixel for 3×3conv. layers. Spatial pooling was carried out by five max-pooling layers, which follow some of the Conv. layers (not all the conv. layers are followed by max-pooling). Max-pooling was performed over a 2×2pixel window, with stride 2.



2.4 VGG16 Architecture

A stack of convolutional layers is followed by three Fully-Connected (FC) layers: the first two have 4096 channels each, the third performs 1000-way ILSVRC classification and thus contains 1000 channels (one for each class). The final layer was the soft-max layer. The configuration of the fully connected layers was the same in all networks. All hidden layers are equipped with the rectification (ReLU) non-linearity.

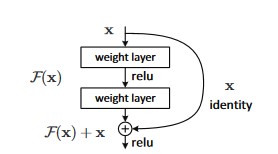
* + 1. Residual Neural Network

A residual neural network (ResNet) is an artificial neural network (ANN) of a kind that builds on constructs known from pyramidal cells in the cerebral cortex. Residual neural networks do this by utilizing *skip connections*, or *shortcuts* to jump over some layers.

ResNet is the network that won the 2015 ILSVRC classification competition with a top-5 error rate of 3.57%. It also won COCO 2015 competition in ImageNet Detection, ImageNet localization, Coco detection and Coco segmentation.

It solves the problem of accuracy saturation and accuracy degradation with the convergence of deeper networks known as vanishing gradient problem.

With any network, we compute the error gradient at the end of the network and use backpropagation to propagate our error gradient backwards through the network. Using the chain rule, we have to keep multiplying terms with the error gradient as we go backwards. However, in the long chain of multiplication, if we multiply many things together that are less than 1, then the final result will be very small. This applies to the gradient as well: the gradient becomes very small as we approach the earlier layers in a deep architecture. This small gradient is an issue because then we can’t update the network parameters by a large enough amount and training is very slow. In some cases, the gradient actually becomes zero, meaning that the earlier parameters are not updated at all! The idea behind ResNet is to make network deep but it should remain shallow. It does this by stacking *residual blocks* together where an identity function is used between them to preserve the gradient.



2.5 Residual Learning: a building block

If we take our input, apply some function to it and add it to our original input. Then, when we take the gradient, it is simply 1.

Mathematically, we can represent the residual block like this.

\[ H(x) = F(x) + x \]

Where, F(x) = W2 \*relu(W1\*x+b1)+b2

It turns out that these residual blocks due to gradient preserving properties are so powerful that we can stack many of these to produce networks that are over 5 times deeper than before! The deepest variant of ResNet was ResNet-151. That’s *151 layers deep*.

* 1. **CROSS -VALIDATION**

Cross-validation is the step where the best parameters for the algorithm are selected. The problem of overfitting and underfitting is discovered using cross-validation. Normally a machine learning problem has many input feature, so it is not possible to visualize the data or the problems that might be occurring. Using cross-validation, such problems can be identified via the learning curves. The two main problems encountered are underfitting and overfitting.

* + 1. Underfitting:

Underfitting occurs when the algorithm cannot properly fit the training set. The curve produced is probably not complex enough for the classification purpose. A synonym to underfitting is high bias. To identify the presence of underfitting, learning curves need to be plotted. A learning curve with the training error and cross-validation error needs to be plotted. If both the training error and 16cross validation are high and there is a small gap between the curves, it can be positively inferred that the algorithm has underfitted the training set.



2.6 Different type of Fit Curve

* + 1. Overfitting**:**

Overfitting occurs when the algorithm fits the training set a bit too well and does poorly in the test set. The algorithm fit the training set a bit too much, thus it was not able to generalize for unseen examples in the test set. A synonym to overfitting is high variance.

**CHAPTER 3**

**PROBLEM DEFINITION & OBJECTIVES**

* 1. **MOTIVATION**

Classification of bird Species is an interesting area in ecology for monitoring birds’ populations, to better understand about a large number of birds’ species and other organisms they interact in the environment. Accurate bird recognition is also essential for avian biodiversity conservation. But, gathering and collecting information about birds is tedious, erroneous as well as time-consuming process. Many efforts have been made for information to be shared and discussed, so that recorded information can be easily analyzed and classified. Even after all this "large scale bird identification" remains almost an impossible task to be done manually. To solve this problem Cornell Lab of Ornithology, USA developed an application in 2014 called Merlin Bird ID, which presents a shortlist of possible species based on descriptions provided by the user. It provides result after learning through the description provided by the contributors. Similarly, The E-Guide to Birds of the Indian Subcontinent is an interactive companion to Birds of the Indian Subcontinent – the definitive guide for birdwatchers visiting the Indian subcontinent. It covers India, Pakistan, Nepal, Bhutan, Bangladesh, Sri Lanka and the Maldives. It provides the Bird information through Index by Common or Scientific bird names either alphabetically or taxonomically. This motivates us to aim for an application which will help in recognition of birds around the Indian subcontinent, which will help bird watchers and ornithologists to learn about the birds easily and make better effort towards conservation and

preservation of birds’ species.

* 1. **OBJECTIVES & GOALS**

This project mainly focuses on the development of a system which will predict the species of birds, from an image and classify them as per their species. The primary objective of our project is the development of a Deep Convolutional Neural Network (DCNN) model, and training that model on Caltech-UCSD-Birds-200-2011 dataset, such that it will generate a score sheet during testing, which will be analyzed to predict the species of birds present in the image. Our other objective is to analyze the performance of the model by comparing it to some state-of-the-art classification algorithms.

Our goal is to create a DCNN model which will give decent result, and develop an application which will replace the current hectic manual methods of classification of birds.

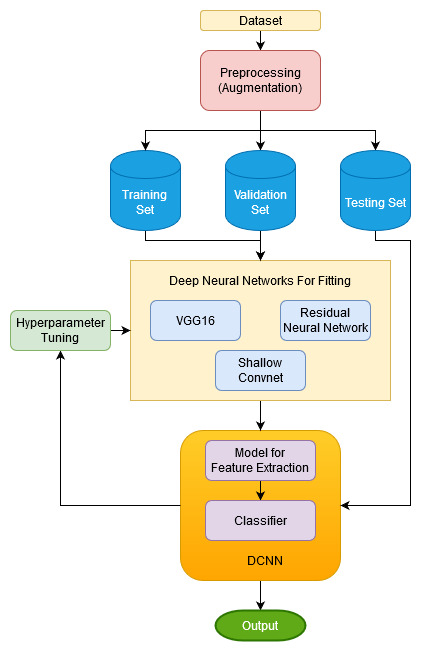
**CHAPTER 4**

**DESIGN**

This chapter contains the proposed model, information regarding dataset, and classifying algorithms that are used for analysis performance.

* 1. **PROPOSED MODEL**

The first phase of our project is data collection. We have collected our dataset by selecting 20 birds species from the 200 bird’s species present in the Caltech-UCSD Birds 200 (CUB-200-2011) dataset.



4.1 Proposed Model

The dataset contains images of birds organized by scientific classification (order, family, genus, and species). From the

directory structure of dataset, features and labels of the dataset are identified, After that, the dataset is divided into two sets, one for training where most of the data is used and the other one is testing.

On the training set, three different classification algorithms have been fitted for the analysis performance of the model. The algorithms we used are Convolutional Neural Network (Convnet-1), VGG-16 based Neural Network and Resnet-50 based Neural Network. After the system has done learning from training datasets, newer data is provided without outputs. The final model generates the output using the knowledge it gained from the data on which it was trained. In the final phase, we get the accuracy of each algorithm and get to know which particular algorithm will give us more accurate results for the prediction of bird’s species.

* 1. **IMPLEMENTATION**

To implement the algorithms for the classification of birds concerning their species we have performed the following steps:

* + 1. Data Collection:

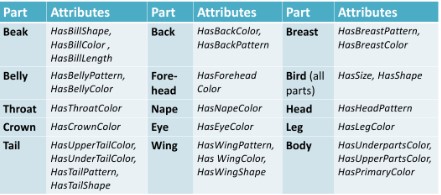
In our project, we will be using the Caltech-UCSD-Birds-200-2011 dataset. This dataset includes 200 categories of bird species, 11,788 total number of images, and other information such as labelled visible bird parts, binary attributes, and bounding boxes surrounding the birds. The authors also include a recommended test and training set split of the data.



4.1 CUB-200 2011 dataset

The list of species names in the dataset was obtained using an online field guide. Images were harvested using Flickr image search and then filtered by showing each image to multiple users of Mechanical Turk.

* + 1. Data Description:
* **Attributes:** A vocabulary of 28 attribute groupings (see Fig 4.2) and 312 binary attributes (e.g., the attribute group belly colour contains 15 different colour choices) was selected based on an online tool for bird species identification. All attributes are visual in nature, with most pertaining to colour, pattern, or shape of a particular part.



4.2 Features of the dataset

* **Species**: In the Caltech-UCSD-Birds-200-2011, there are images of 200 birds species of which, following 20 birds species were chosen for this project.
* ‘Black footed Albatross',
* 'Rhinoceros Auklet',
* ‘Rusty Blackbird',
* ‘American Crow',
* ‘Purple Finch',
* 'Yellow bellied Flycatcher',
* 'American Goldfinch',
* 'Blue Grosbeak',
* ‘Green Jay',
* 'Ruby throated Hummingbird',
* 'Green Jay',
* ‘Gray Kingbird',
* 'Orchard Oriole',
* 'White Pelican',
* 'White necked Raven',
* 'American Redstart',
* ‘White throated Sparrow',
* 'White crowned Sparrow',
* ‘Bank Swallow',
* 'Tree Swallow',
* ‘Red eyed Vireo’
* **Dataset Size**:

The dataset contains a total of 1171 images for selected 20 species.

* + 1. Pre-processing the Dataset

After selecting 20 species, and splitting their images based on recommended train-test-split file provided by the author of the dataset, we had 599 images for training and 572 images for testing, which is nearly divided in the ratio of 50:50. Since the number of images was small, we apply augmentation like varying brightness of the image, rotating image both horizontally and vertically as well as shearing images to a small extent to increase the number of images for training. After augmentation out training data increases to 4112 images which is further divided into training and validation set in the ratio of 9:1.

* + 1. Split Dataset:

Separating data into training and testing sets is an important part of evaluating data mining models. Typically, when separating a data set into two parts, most of the data is used for training, and a smaller portion of the data is used for testing. We have also split our dataset into two sets. One is for training and another for testing. The training set contains a known output and the model learns on this data in order to be generalized to other data later on. After the model has been processed by using the training set, we have tested the model by making predictions against the test set. Because the data in the testing set already contains known values for the attribute that we want to predict, it is easy to determine whether the model's guesses are correct or not. In addition, we have used 79% of our data for training, 9% for validation and 12% for testing.

* + 1. Algorithm

The proposed approach for bird species classification by considering features and parameters such as size, shape, etc. of the bird on the Caltech-UCSD Birds 200 (CUB-200-2011) dataset is evaluated by different Neural Network Models to learn which algorithm gives good results for which use case. In this, the training of dataset is done by using Google-Collab, which is a platform to train dataset by uploading the images from the local machine or the Google drive. After training labelled dataset is ready for classifiers for image processing. Before providing an image as input to the model, the pixel value is changed in the range of 0-1, to make loss calculation more effective and the image size is also changed to 128 X 128 X 3. Initially, the image was read in grayscale and was trained on the CNN model (designed by us by tuning parameters like an optimizer, depth of hidden layers) and was again trained on the RGB images which give the better result as compared to the grayscale model. To compare the performance of our CNN model, we decided to choose VGG-16 based CNN (for 20 output classes) with large numbers of hidden layers as compared to our CNN. We decide to train our model using ImageNet weight as well as training the model by initializing the nodes randomly. We get a better result with the model with a higher number of the hidden layer as compared to our shallow CNN model (Convnet-1). To evaluate the performance of the model with the increase in hidden layer we trained our model on Residual Neural Network 0f 50 layers which have accuracy less than our model, Convnet1 for the same epoch (i.e. 50). The accuracy of the ResNet model varies proportionally with the epochs. VGG-16 model with pre-trained ImageNet weight gives a better result than other models.

* + 1. Software Requirement

Software used in this project played a big role in terms of result.

**Google colab**

Google Colab is a free cloud service and it supports free GPU. It supports Python 2.7 and 3.6. Colab is ideal for everything from improving Python coding skills to working with deep learning libraries, like PyTorch, Keras, TensorFlow, and OpenCV. One can create notebooks in Colab, upload notebooks, store notebooks, share notebooks, mount your Google Drive and use whatever got stored in there, import most of your favourite directories, upload your personal Jupyter Notebooks, upload notebooks directly from GitHub, upload Kaggle files, download the notebooks, and do just about everything else that one might want to be able to do.

**Python**

A great choice of libraries is one of the main reasons Python is the most popular programming language used for machine learning. A library is a module or a group of modules which include a pre-written piece of code that allows users to reach some functionality or perform different actions. Python libraries provide base-level items so developers don’t have to code them from the very beginning every time. Python programming language resembles the everyday English language, and that makes the process of learning easier. Python is not only comfortable to use and easy to learn but also very versatile. What we mean is that Python for machine learning development can run on any platform including Windows, macOS, Linux, Unix, and others.

**Keras**

Keras is an open-source neural-network library written in Python. It is capable of running on top of TensorFlow, Microsoft Cognitive Toolkit, R, Theano or PlaidML. Designed to enable fast experimentation with deep neural networks, it focuses on being user-friendly, modular, and extensible. It was developed as part of the research effort of project ONEIROS (Open-ended Neuro-Electronic Intelligent Robot Operating System). It offers a higher-level, more intuitive set of abstractions that make it easy to develop deep learning models regardless of the computational backend used. Keras contains numerous implementations of commonly used neural-network building blocks such as layers, objectives, activation functions, optimizers, and a host of tools to make working with image and text data easier to simplify the coding necessary for writing deep neural network code. In addition to standard neural networks, Keras has support for convolutional and recurrent neural networks. It supports other common utility layers like dropout, batch normalization, and pooling.

**Numpy:**

NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.

**Matplotlib:**

Matplotlib is a plotting library for Python. It is used along with NumPy to provide an environment that is an effective open-source alternative for MatLab. It can also be used with graphics toolkits like PyQt and wxPython.

**Scikit learn:**

Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. The library is built upon the SciPy (Scientific Python).

**CHAPTER 5**

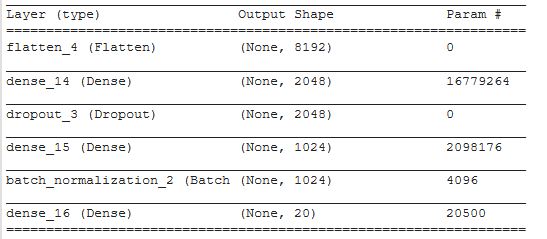
**EXPERIMENTAL RESULTS AND ANALYSIS**

In the previous chapter, we have discussed about our proposed system and implementation of our project. We have demonstrated how we collected our dataset, dataset description, and algorithms we used. Now, we will discuss the results that we obtained from our experiments upon the implementation of this system. We have divided our dataset into two parts- training and testing dataset. We will show the outcome of the training(and validation) and testing dataset. As mentioned before, we have used two machine learning algorithms. First, we trained our dataset with these two algorithms and then we built a model. Then, we tested our testing dataset in this model. If the test set accuracy is near to train set accuracy then we can conclude that we built a good model. We have total 4684 data of different individual images along with 20 labels in our dataset. After splitting the data into two parts now we have 4152 images for training and for testing we have 572 images. When we trained our training data on the models, the result we got for analysis of the performance of different algorithms is as follows–

* 1. **VGG16 model without ImageNet weights**
     1. Structure**:**

The top feature extraction layers were taken from VGG16 with total trainable parameters equal to 14,714,688 and the classifier

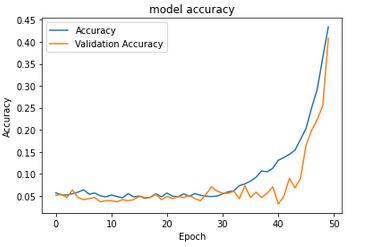
has trainable parameters equal to 18,899,988, which makes the total trainable parameters equal to 33,614,676. The structure of the classifier is given in figure 5.1.



5.1 Structure of Classifier

* + 1. Training accuracy:

For VGG-16 model trained without any pre-trained weights, we got both training and testing accuracy of 52%. After epoch 30, the validation accuracy and testing accuracy starts increases drastically.



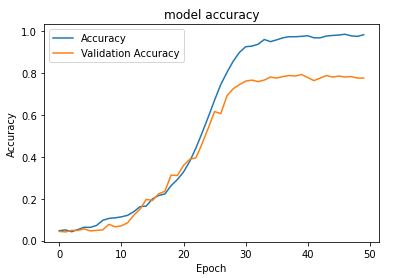
5.2 Accuracy of VGG16 Without pre-trained weights

* 1. **VGG16 model with ImageNet weight**
     1. Structure:

The top feature extraction layers were taken from VGG16 with total trainable parameters equal to 14,714,688 and the classifier has trainable parameters equal to 18,899,988, which makes the total trainable parameters equal to 33,614,676. The structure of the classifier is given in figure 5.1.

* + 1. Training accuracy:

For VGG-16 model initialize with ImageNet weights, we got both training and testing accuracy of 97%. After Epoch 15 accuracy starts increasing rapidly till it starts saturating around Epoch 30.



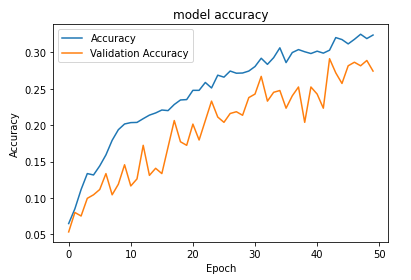
5.3 Accuracy of VGG16 With pre-trained weights

* 1. **ResNet-50 model**
     1. Structure:

The top feature extraction layers were taken from ResNet-50 layers model with total trainable parameters equal to 14,714,688 and the classifier has trainable parameters equal to 18,899,988, which makes the total trainable parameters equal to 33,614,676. The structure of the classifier is given in figure 5.1.

* + 1. Training accuracy:

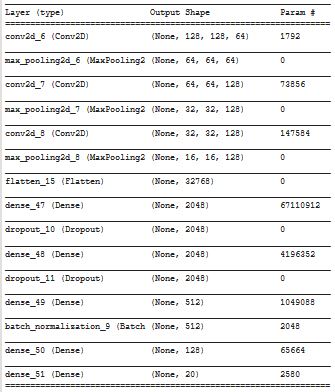
For ResNet-50 model initialize with ImageNet weights, we got both training and testing accuracy of 32% and 52% respectively.



5.4 Accuracy of ResNet With pre-trained weights

* 1. **Convnet-1**
     1. Structure:

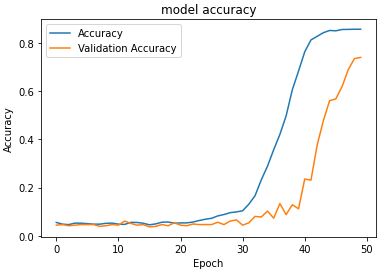
This Convolutional Neural Network is designed by us by varying different parameters and depth of the hidden layer. The total trainable parameters in this model are equal to 72,648,852. The structure of the model is given in figure 5.5



5.5 Structure of Classifier of Convnet-1

* + 1. Training accuracy:

For VGG-16 model initialize with ImageNet weights, we got both training and testing accuracy of 85% and 52% respectively.

****

5.6 Accuracy of Convnet-1

* 1. **Comparison Between Algorithms**

|  |  |  |
| --- | --- | --- |
| Algorithms | Testing Accuracy  (in %) | Training Accuracy  (in %) |
| Convnet-1 | **52** | **85** |
| VGG16 With ImageNet weight | **97** | **97** |
| VGG16 Without ImageNet weight | **52** | **52** |
| ResNet-50 With ImageNet weight | **52** | **32** |

Table 5.1 Accuracy of Algorithms

**CHAPTER 6**

**FURTHER STUDIES**

In this chapter, we will discuss, the further studies that we have done during this project and about the future work of this project.

* 1. **BIRD SPECIES CLASSIFICATION FOR INDIAN SUBCONTINENT REGION**

In our dataset “Caltech-UCSD Birds-200-2011 (CUB-200-2011)” most of the bird species belong to the American Continent, and it does not have birds species found in Indian Subcontinent. We looked for the dataset for Indian Birds but couldn’t find one so, we decided to create our own dataset for Indian birds, and evaluate our model on our own dataset.

* + 1. Dataset:

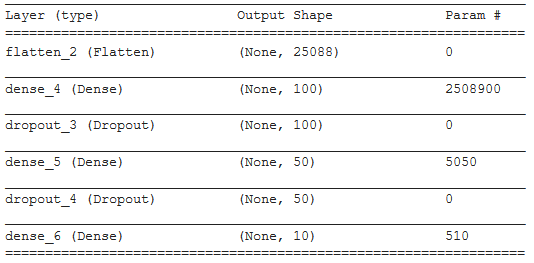
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Species | | No of Training images | No. of Testing images | No. of Validation Images |
| Himalayan bluetail | 466 | | 12 | 98 |
| Sarus Crane | 439 | | 10 | 96 |
| The Himalayan Griffon Vulture | 443 | | 14 | 96 |
| Himalayan Monal | 412 | | 10 | 89 |
| Kalij Pheasant | 449 | | 11 | 99 |
| The Oriental Dwarf Kingfisher | 405 | | 11 | 88 |

|  |  |  |  |
| --- | --- | --- | --- |
| The Indian Pitta | 417 | 12 | 87 |
| Red headed Trogon | 534 | 10 | 122 |
| The Indian Peafowl | 403 | 10 | 89 |
| Mrs Goulds sunbird | 459 | 10 | 100 |

Table 6.1 Distribution of images in Training, Testing and Validation sets

* + 1. Algorithm:

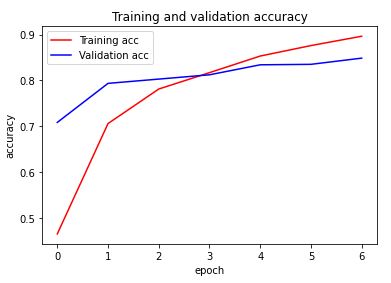
In our model, the top feature extraction layers were taken from VGG16 with ImageNet weights, and the dataset is run once on the model to create bottleneck file which will save the static value of parameters, since re-training those layers again on the new dataset is time-consuming. The structure of fully connected classifier layers is given in figure 6.1.



6.1 Structure of Classifier of Convnet-2

* + 1. Evaluation:

Evaluation of our model on the dataset of Indian Birds, we got both training and testing accuracy of 84% and 91% respectively.



6.2 Accuracy of Convnet-2

* 1. **FUTURE WORK**
* We have implemented the project only for 20 Species in case of CUB 200-2011 dataset and for 10 birds for Indian Birds dataset as when we ran it on our machine for just 10 epoch for a large dataset, it went out of memory and could not complete. So, we can try to run Neural Networks on high-performance computing machines.
* We will use Computer vision algorithms for automatic feature extraction.
* We will develop an application that identifies a bird in real-time on clicking its photo.
* We will create a model based on residual blocks to increase the depth of hidden layers.
* The birds Classification system can be implemented using cloud which can store a large amount of data for comparison and provide high computing power for processing.
* We could use a detection algorithm to localize birds in the image and run our classification algorithm on the localize part.

**CHAPTER 7**

**CONCLUSION**

* 1. **CONCLUDING REMARKS**

We have tried to construct a neural network model to predict if a bird belongs to a species using features from images. After training and testing the model the accuracy we get is quite similar. For both sets VGG-16 NN with pre-trained weights providing higher accuracy rate. Despite the shortcomings in reaching good performance results, this work provided a means to make use and test multiple machine learning algorithms. It also allowed exploring a little feature selection, parameter selection and dataset generation problems and experiences the constraints in computation time when looking for possible candidate models in high combinatorial spaces, even for a small dataset as the one used. The structure of our project has been built in such a way that with a proper dataset and minor alternation it can work to classify the birds in any number of categories.

**BIBLIOGRAPHY**

1. Wah C., Branson S., Welinder P., Perona P., Belongie S. “The Caltech-UCSD Birds-200-2011 Dataset.” Computation & Neural Systems Technical Report, CNS-TR-2011-001 ,<http://www.vision.caltech.edu/visipedia/CUB-200-2011.html>
2. Classification of Birds,  [https://www.birds.com/species/classifications/](%20https:/www.birds.com/species/classifications/)
3. Karen Simonyan, Andrew Zisserman *,” Very Deep Convolutional Networks For Large-Scale Image Recognition”*, Published as a conference paper at ICLR 2015, arXiv:1409.1556.
4. A. Marini, A. J. Turatti, A. S. Britto and A. L. Koerich*,* "Visual and acoustic identification of bird species", IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*,* 2015
5. Madhuri A. Tayal, Atharva Mangrulkar, Purvashree Waldey, Chitra Dangra, *“Bird Identification by Image Recognition”*, Helix Vol. 8(6): 4349-4352, 31st October 2018
6. Alter, A. L., & Wang, K. M. *“An Exploration of Computer Vision Techniques for Bird Species Classification”*, 15th December 2017, Retrieved from <http://cs229.stanford.edu/proj2017/final-reports/5161697.pdf>
7. Prof. Pralhad Gavali, Ms. Prachi Abhijeet Mhetre, Ms. Neha Chandrakhant Patil, Ms. Nikita Suresh Bamane, Ms. Harshal Dipak Buva, *“Bird Species Identification using Deep Learning* “, IJERT, vol. 8 Issue 4 April-2019”
8. Aditya Bhandari, Ameya Joshi, Rohit Patki, *“Bird Species Identification from an Image”*, Stanford University Report, 2016.
9. Juha Niemi and Juha T. Tantt, *“Deep Learning Case Study for Automatic Bird Identification“*, Applied Sciences Article, 29th October 2018
10. Tasfia Shermin, Shyh Wei Teng, Manzur Murshed, Guojun Lu, FerdousSohel, and Manoranjan Paul, *“Enhanced Transfer Learning with ImageNet Trained Classification Layer”*,2019, arXiv:1903.10150v2
11. John Martinsson, *“Bird Species Identification using Convolutional Neural Network”*, Univ. of Gothenburg, Master’s thesis in Computer Science, 2017
12. Alom, M.Z.; Taha, T.M.; Yakopcic, C.; Westberg, S.; Sidike, P.; Nasrin, M.S.; Hasan, M.; Van Essen, B.C.; Awwal, A.A.S.; Asari, V.K. *“A State-of-the-Art Survey on Deep Learning Theory and Architectures”*. Electronics *2019,* 8*, 292.*