International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 5, pp 190-204, May-2024

https://doi.org/10.47001/IRJIET/2024.805029

Classification of Endangered Bird Species of Nepal Using Deep Learning

[1]Sumit Pant, [2]Sandip Shrestha, [3]Abhishek Aryal, [4]Ozan Wagle, [5]Nabin Lamichhane

[1,2,3,4]Affiliated to Tribhuvan University, Department of Electronics and Computer Engineering, Institute of Engineering (IOE) Pashchimanchal Campus, Pokhara, Gandaki, Nepal

[5]Assistant Professor at Tribhuvan University, Department of Electronics and Computer Engineering, Institute of Engineering (IOE) Pashchimanchal Campus, Pokhara, Gandaki, Nepal

***Abstract* - This paper introduces an innovative approach to the pressing conservation challenge of accurately identifying endangered bird species, with a focus on Nepal's diverse avian population. Using Convolutional Neural Networks (CNNs), our deep learning system achieves impressive accuracy in classifying bird species from images. We compile a comprehensive dataset of 8,457 high-quality images representing 38 endangered bird species native to Nepal, sourced from various online platforms. Through meticulous data augmentation, we enhance dataset diversity and model robustness. Four CNN models are developed and rigorously evaluated, with test accuracies ranging from 83.29% to an impressive 90.8%. The highest-performing model is seamlessly integrated into a user-friendly web application built on Django, allowing users to upload bird images for real-time classification. Our findings highlight the potential of deep learning in advancing conservation efforts, offering scalable solutions for monitoring and protecting endangered avian populations. This work contributes to the intersection of artificial intelligence and conservation biology, demonstrating the crucial role of technology in preserving Earth's biodiversity.**

Keywords: CNN, Image Classification, Bird Species Classification System, Web Application.

I. INTRODUCTION

Birds, spanning from the verdant rainforests of Indonesia to the rugged landscapes of Nepal, hold a significant place in global ecosystems. Despite their adaptability, numerous bird species are threatened by human activities and environmental changes. The International Union for Conservation of Nature (IUCN) Red List highlights the perilous state of avian biodiversity, with approximately 1,445 species classified as threatened or endangered worldwide. Nepal, renowned for its rich avifauna, faces a pressing conservation challenge as 38 of its bird species are classified as endangered, primarily due to habitat loss, climate change, and hunting pressures.

Artificial intelligence (AI) intersects with conservation biology to offer innovative solutions to mitigate the threats facing endangered bird species. Deep learning algorithms, a subset of AI, provide opportunities for precise species classification and monitoring. By harnessing AI-driven image processing and data analysis, researchers can develop advanced tools to aid in the identification and protection of endangered bird species, thereby enhancing conservation efforts.

This research endeavors to address Nepal's conservation challenge by developing a deep learning system for the accurate classification of endangered bird species. Leveraging advanced AI techniques, the research project aims to improve monitoring efforts and inform targeted conservation interventions. Additionally, the project seeks to curate a comprehensive dataset of Nepal's avian fauna, laying the groundwork for future conservation initiatives. By embracing the synergy between AI and conservation biology, this research aims to contribute to the preservation of Nepal's avian biodiversity and beyond, emphasizing the urgency of innovative conservation strategies in the face of escalating threats to endangered bird species worldwide.

II. LITERATURE SURVEY

In the paper “Bird Species Identification using Convolutional Neural Networks”, John Martinsson et al. discuss the need of system for monitoring animal populations to better understand their behavior, biodiversity, and population dynamics. The CNN method and deep residual neural networks were presented to detect a picture in two ways, based on feature extraction and signal classification. They conducted an experimental investigation for datasets made up of various image types. They neglected to consider the background species, though. Larger volumes of training data are necessary to identify the background species; however, these may not be available [2].

Juha Niemi, Juha T Tanttu et al (2018), proposed a Convolutional neural network trained with deep learning algorithms for image classification. It also proposed a data

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 5, pp 190-204, May-2024

https://doi.org/10.47001/IRJIET/2024.805029

augmentation method in which images are converted and rotated in accordance with the desired color. The final identification is based on a fusion of parameters provided by the radar and predictions of the image classifier. They demonstrated that data augmentation method is suitable for image classification problem and it significantly increases the performance of the classifier [3].

Madhuri A. Tayal, Atharva Magrulkar et al (2018), developed a software application that is used to simplify the bird identification process. This bird identification software takes an image as an input and gives the identity of the bird as an output. They used the concept of transfer learning and pre-trained algorithm made feature extraction possible from an image. The result obtained were of high efficiency as the software could easily identify a bird species from an image whose dataset was present in the database [4].

Andreia Marini, Jacques Facon and Alessandro L. Koerich et al (2013), presents a novel approach for bird species classification based on color features extracted from unconstrained images. The approach addresses challenges such as variations in scenarios, bird poses, sizes, and viewing angles, as well as occlusions and lighting variations in the images. A color segmentation algorithm is applied to eliminate background elements and identify candidate regions where the bird may be present. Experimental results on the CUB-200 dataset show a 75% correct segmentation rate and varying bird species classification rates (ranging from 90% to 8%) depending on the number of classes considered [5].

In this paper, “Genetic Algorithm based hyper-parameters optimization for transfer Convolutional Neural Network”, X. Xiao et al. proposed a Genetic Algorithm to optimize the hyper-parameters in CNNs. In their work, they did not restrain the depth of the model. Experimental results show that they can find satisfactory hyper- parameter combinations efficiently with accuracy about 88.92% and within 24.55 hours which is relatively better than random search algorithm [6]. Similarly, In the study titled "Nepali Image Captioning: Generating Coherent Paragraph-Length Descriptions Using Transformer," Subedi et al. (2024) employed the Inception V3 model for feature extraction from cultural heritage images which can classify a wide range of objects across various classes [11].

In this paper, “Samrakshyan: An Endangered Birds Recognition Portal”, B. Khatiwada, B. Subedi, N. Duwal and R. et al. proposed an Endangered Birds Recognition portal for the preservation of bird species and monitoring the status of birds in the ecosystem which can assist researchers of Nepal’s biodiversity in planning different strategies for their preservation. They developed a system to identify bird calls from the audio data set collected from Xeno-canto.org. They extract spectral characteristics of the audio signal through Mel-Spectrogram and MFCC (Mel-Frequency Cepstral Coefficients) which generated the spectrogram. It was fed into the deep learning model architecture like efficient Net which is based on a convolutional neural network. Experimental results show that they can find satisfactory hyper- parameter combinations efficiently with F1-Score of 79% for 41 species of birds [7].

III. PROBLEM DEFINITION

The current scenario in Nepal underscores a pressing conservation challenge: the existence of 38 endangered bird species, each representing a critical component of the country's biodiversity. These species face imminent threats, accentuating the urgency for effective conservation measures. However, the efficacy of current strategies is hindered by the absence of efficient methods for accurately identifying these endangered birds. Manual identification processes are often time-consuming and resource-intensive, leading to gaps in monitoring efforts. As a result, there is a compelling need for a more precise classification approach that harnesses advanced technologies to enhance our ability to identify and protect these vulnerable species.

IV. METHODOLOGY

4.1 System Block Diagram

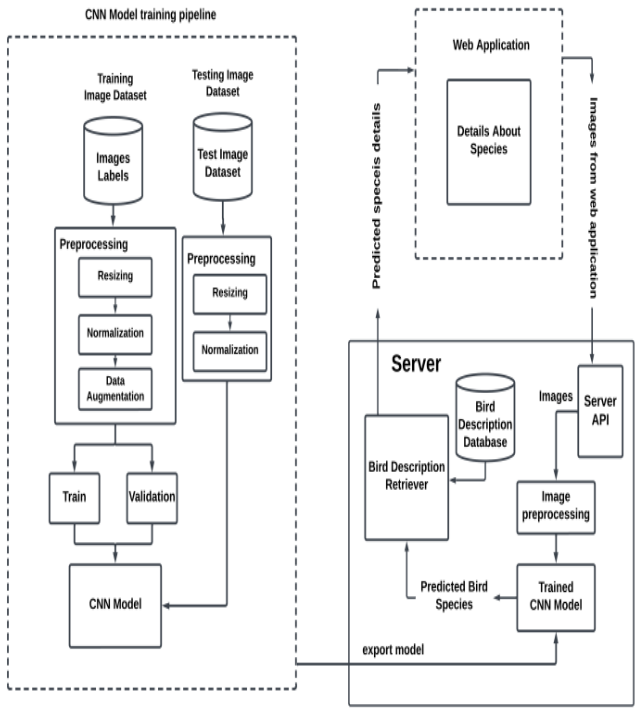

Figure 4.1: Block diagram of the system

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 5, pp 190-204, May-2024

https://doi.org/10.47001/IRJIET/2024.805029

- **System Overview**

The block diagram depicted in Figure provides a high-level understanding of the system's operation. It delineates the flow of data within the system, illustrating how the user interacts with the server through a web application. At the core of the server lies a CNN model, meticulously trained to accurately predict bird species.

- **CNN Model Workflow**

The CNN model's workflow commences with the segmentation of the image dataset into training and testing subsets. This segmentation is indispensable for evaluating the model's performance on unseen data, ensuring its generalization capability. Preprocessing plays a pivotal role in preparing the image data for training, encompassing tasks such as resizing images to a consistent size, normalizing pixel values, and employing techniques like data augmentation to enhance training set diversity. Furthermore, the training data undergoes further partitioning for validation, enabling continuous monitoring of the model's performance and averting overfitting.

- **Training Process**

Subsequent to data preparation, the training process commences by feeding the training data into the CNN model. Through iterative optimization of internal parameters via loss function minimization, the model learns to discern intricate patterns within the input data, refining its ability to classify bird species accurately.

- **Evaluation Metrics**

Post-training, the model's efficacy is assessed using evaluation metrics such as accuracy, weighted precision, and weighted F1-score. These metrics furnish insights into the model's performance on unseen data, gauging its effectiveness in making precise predictions.

- **Prediction and Output**

Upon successful training and evaluation, the server stands ready to process incoming images using the trained CNN model. Leveraging its learned knowledge, the model predicts the bird species depicted in the input images. The predicted output, comprising the identified bird species and their corresponding details, is then relayed to the web application for user interaction and further analysis. This seamless integration of prediction and output delivery ensures a streamlined user experience and facilitates informed decision-making based on the model's insights.

4.2 Image collection and dataset preparation

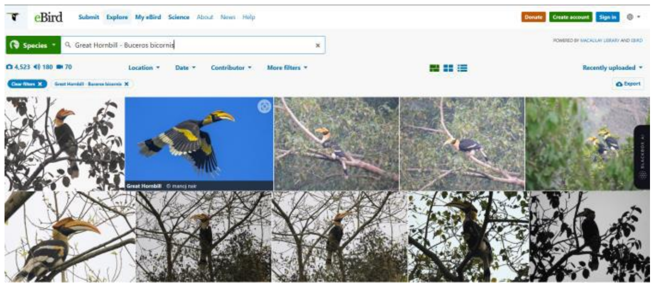

Figure 4.2: ebird.org user interface

Globally shared bird photos can be found on the ebird.org websites [8]. The contributors who travel around post the photographs on the website. We collected 8475 images of 38 bird species from a website. These images were divided into two sets: one with a split ratio of 0.9:0.5:0.5 and another with a split ratio of 80:10:10. In the first split, the training set consisted of 7671 images for training, the validation set contained 402 images for validation during training, and another 402 images for testing the model. For the second split, the training set included 6780 images, while the validation set and testing set each contained 847 images.

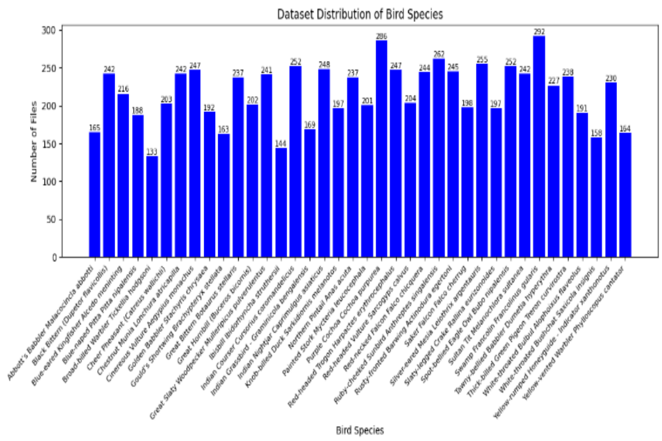

Figure 4.3: Dataset distribution of bird species

4.3 Data preprocessing

To ensure uniformity in the dataset, we addressed significant variations in image size and quality. Originally high-quality images occupied considerable storage space, prompting us to standardize them. We resized all images to a consistent dimension of 224x224 pixels while maintaining the original aspect ratio. This step ensures uniformity across the dataset and facilitates effective model training.

All images in the dataset are in RGB color format. To prevent potential numerical issues during model training, we normalized the pixel values between 0 and 255. This

ISSN (online): 2581-3048
Volume 8, Issue 5, pp 190-204, May-2024
https://doi.org/10.47001/IRJIET/2024.805029

normalization process ensures that pixel values are represented in a standardized range suitable for effective model training.

These preprocessing steps are crucial for achieving consistent image sizes, reducing storage requirements through compression, and normalizing pixel values to improve model training outcomes. They lay the foundation for effective CNN model training by ensuring uniformity and standardization across the dataset.

4.4 Data augmentation

Data augmentation involves creating modified copies of a dataset using pre-existing data, effectively expanding the training set. In deep learning model training, having a large and diverse dataset is crucial for achieving good performance. However, if the dataset is relatively small, it can limit the model's ability to learn and generalize patterns effectively.

Leveraging the Keras deep learning library, we implemented an image generator for data augmentation. Instead of directly using the original images, this generator produces randomly altered images. These alterations include minor changes to the dataset, generating new data points. By introducing variations in the training data, data augmentation mitigates overfitting and enhances model accuracy.

Data augmentation plays a vital role in deep learning model training by expanding the training set and introducing diversity. It helps to reduce overfitting by exposing the model to a broader range of data variations, ultimately improving its ability to generalize and make accurate predictions.

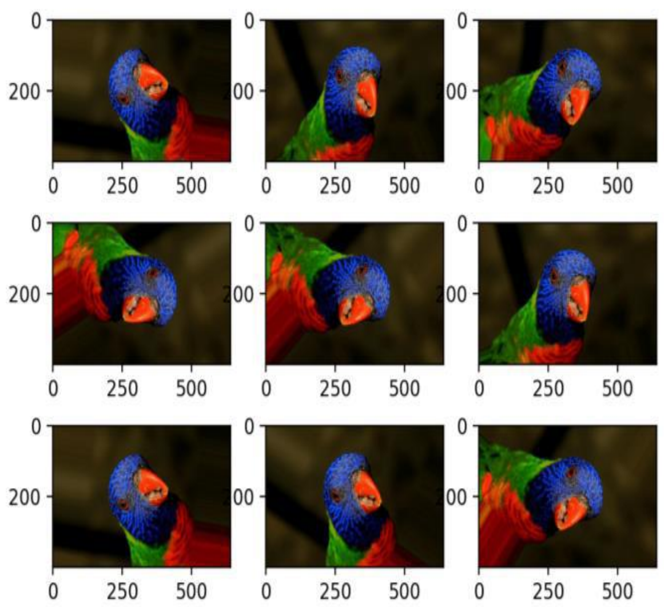

Figure 4.4: Data Augmentation

4.5 CNN (Convolution Neural Network) Architecture

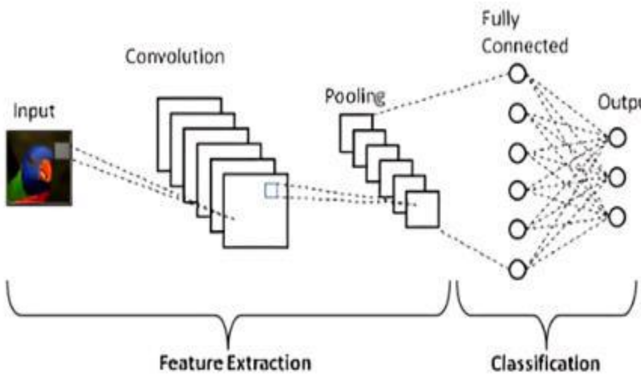

Figure 4.5: A basic CNN architecture

A crucial component of our project's neural architecture is the Convolutional Neural Network (CNN). Specifically designed to process image data efficiently, CNNs employ layers of interconnected neurons that autonomously refine their performance through optimization. In our context, where bird image datasets serve as inputs, CNNs excel due to their inherent ability to extract intricate patterns from visual data.

4.5.1 Key Layers

- Convolutional Layer: Serving as the initial layer, the convolutional layer processes raw image inputs by applying convolution operations. This foundational step enables the network to extract essential features from the images, such as colors, textures, and edges, which are crucial for identifying bird species.
- Pooling Layer: Subsequent layers in the CNN architecture often include pooling layers, which reduce the dimensionality of the feature maps generated by the convolutional layers. By summarizing the information obtained from preceding layers, pooling layers help streamline the learning process and enhance the network's ability to generalize.
- Fully Connected Layer: Positioned at the network's end, the fully connected layer integrates the extracted features to make predictions about the input data. In our bird species classification task, this layer plays a pivotal role in mapping the learned features to specific bird species, facilitating accurate classification.

4.5.2 Hierarchical Feature Extraction

As data progresses through the network's layers, the CNN progressively learns to detect increasingly complex features within the images. Beginning with basic elements like colors and edges, early layers pave the way for higher-level abstractions, enabling the network to discern intricate patterns characteristic of different bird species.

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 5, pp 190-204, May-2024

https://doi.org/10.47001/IRJIET/2024.805029

4.6 Model Analysis

This research project involved the development of four distinct convolutional neural network (CNN) models for the classification of endangered bird species in Nepal. The entire algorithm development process was carried out using Python programming language. Various Python libraries, including NumPy, Pandas, Flask, Scikit-learn, Keras, and TensorFlow, were employed throughout the development phase. The dataset utilized for training and evaluation purposes was obtained from eBird.org, an extensive repository of bird observations.

4.6.1 Model 1:

This Convolutional Neural Network (CNN) model is tailored for processing RGB images of size 224 x 224 pixels. It begins with an initial convolutional layer featuring 64 filters and a kernel size of (3,3), activated by the Rectified Linear Unit (ReLU) function. Batch normalization is applied following each convolutional layer to enhance training stability. Subsequently, max-pooling layers with a pool size of (2,2) and strides of 2 are employed to down sample the feature maps. This pattern of convolution, batch normalization, and max-pooling is repeated three more times with decreasing filter sizes (64, 64, and 32). After the final max-pooling layer, the feature maps are flattened into a one-dimensional array. The flattened output is then passed into dense layers, starting with 512 neurons activated by ReLU, followed by dropout regularization with a rate of 0.3 to prevent overfitting. Another dense layer with 256 neurons and ReLU activation is added, followed by dropout regularization. The output layers comprise a dense layer with the SoftMax activation function, generating probabilities for each class label in the specified classification task. Overall, the model contains a total of 11,275,014 parameters, with 11,274,694 being trainable, indicating its moderate complexity and capacity to learn intricate patterns within the input data.

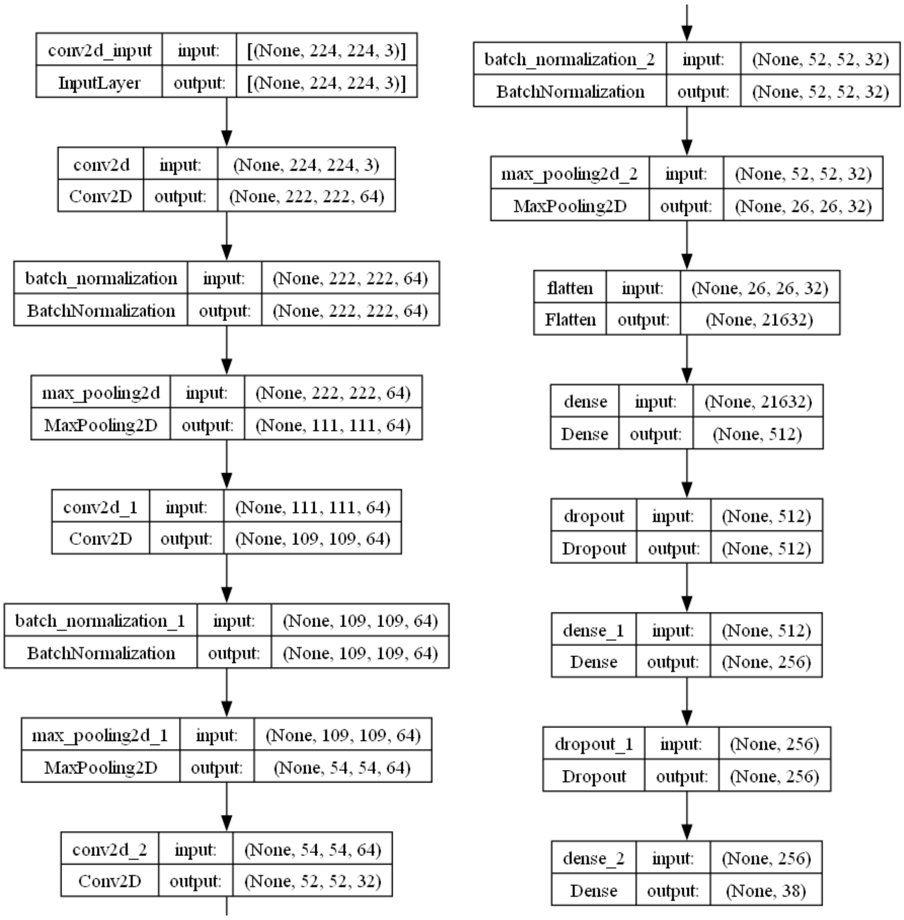

Figure 4.6: Model 1

4.6.2 Model 2:

This CNN classifier comprises 4 convolutional layers, each followed by batch normalization and max pooling. It also integrates 3 fully connected layers, each followed by dropout (30%), culminating in an output layer with 38 nodes. The

International Research Journal of Innovations in Engineering and Technology (IRJIET)
ISSN (online): 2581-3048
Volume 8, Issue 5, pp 190-204, May-2024
https://doi.org/10.47001/IRJIET/2024.805029

convolutional layers employ 64 filters of size 33 with same padding, activated by ReLU. Input images are (224, 224, 3), indicating 224*224 pixel size with RGB channels. Batch normalization normalizes activations, enhancing training stability. Max pooling performs max pooling (2*2) with stride 2 and valid padding, reducing input dimensions. Three additional convolution layers follow with filters 64, 32, and 32, respectively, replicating previous configurations. Flattened layers prepare the output for fully connected layers. The first dense layer has 512 units, ReLU activation, and dropout (30%). The subsequent layers (256, 128, and 64 units) follow a similar pattern with dropout. The final dense layer has 38 units, employing SoftMax activation. This architecture systematically extracts features from input images for bird species prediction.

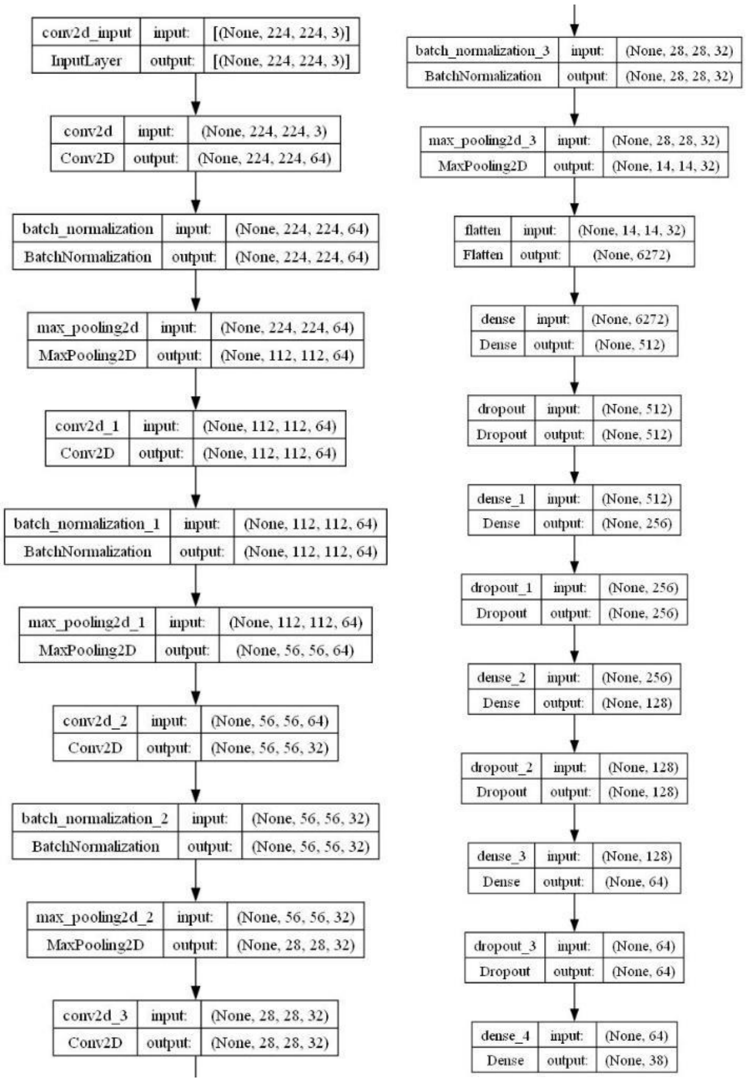

Figure 4.7: Model 2

4.6.3 Model 3:

This CNN model is tailored for processing RGB images with dimensions of 224 x 224 pixels. It commences with an initial convolutional layer comprising 64 filters and a kernel size of (3,3), employing the Rectified Linear Unit (ReLU) activation function. Batch normalization is subsequently applied to stabilize the training process, followed by max-pooling with a pool size of (2,2) and strides of 2. This pattern is replicated with subsequent convolutional layers, each consisting of 64, 32, and 16 filters, respectively, with identical configurations of convolution, batch normalization, and max-pooling. After the final max-pooling layer, the feature maps are flattened into a one-dimensional array. Dense layers follow, starting with 512 ReLU-activated neurons, dropout (0.3), and another dense layer with 256 neurons, dropout. The output layer comprises a dense layer with the SoftMax activation function, producing probabilities for each of the 38 classes in the specified classification task. The model contains a total of 1,383,766 parameters, with 1,383,414 of them being trainable, highlighting its moderate complexity and ability to learn intricate patterns within the input data.

International Research Journal of Innovations in Engineering and Technology (IRJIET)
ISSN (online): 2581-3048
Volume 8, Issue 5, pp 190-204, May-2024
https://doi.org/10.47001/IRJIET/2024.805029

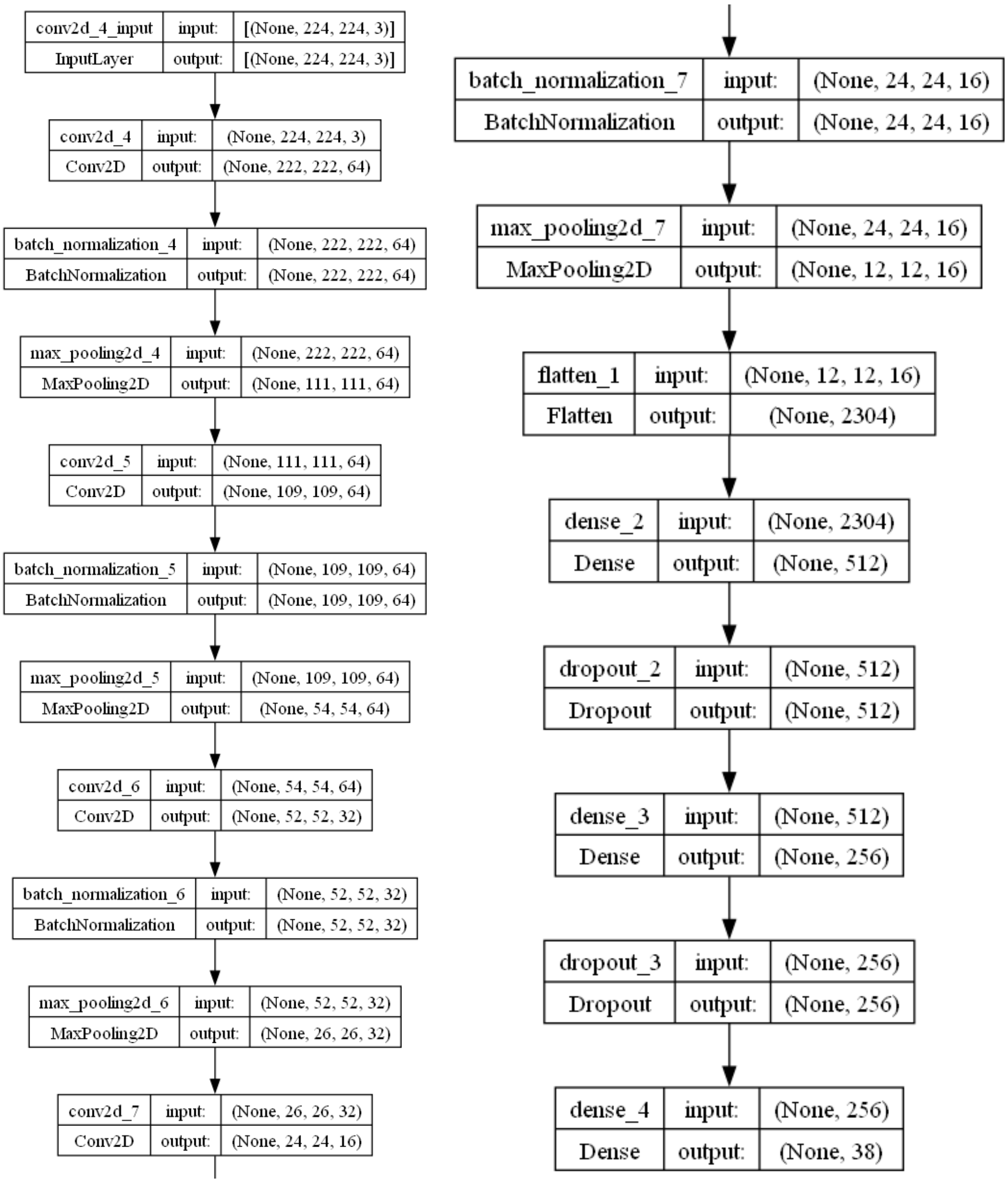

Figure 4.8: Model 3

4.6.4 Model 4:

This CNN model is optimized for processing RGB images with dimensions of 224 x 224 pixels. It initiates with a convolutional layer featuring 64 filters and a kernel size of (3,3), activated by the Rectified Linear Unit (ReLU) function. Batch normalization is applied after each convolutional layer to enhance training stability. Subsequently, max-pooling layers with a pool size of (2,2) and strides of 2 are employed to down sample the feature maps. This pattern of convolution, batch normalization, and max-pooling is repeated in subsequent layers, with decreasing filter sizes of 64, 32, and 16, respectively. After the final max-pooling layer, the feature maps are flattened into a one-dimensional array. The flattened output is then passed into dense layers, starting with 256 ReLU-activated neurons, followed by dropout regularization (30%). Two more dense layers follow, each with 128 and 64 neurons, respectively, also employing ReLU activation and dropout regularization. The output layer comprises a dense layer with the SoftMax activation function, generating class probabilities for each of the 38 classes in the specified classification task. The model contains a total of 696,214 parameters, with 695,862 of them being trainable, indicating its moderate complexity and capacity to learn intricate patterns within the input data.

ISSN (online): 2581-3048
Volume 8, Issue 5, pp 190-204, May-2024
https://doi.org/10.47001/IRJIET/2024.805029

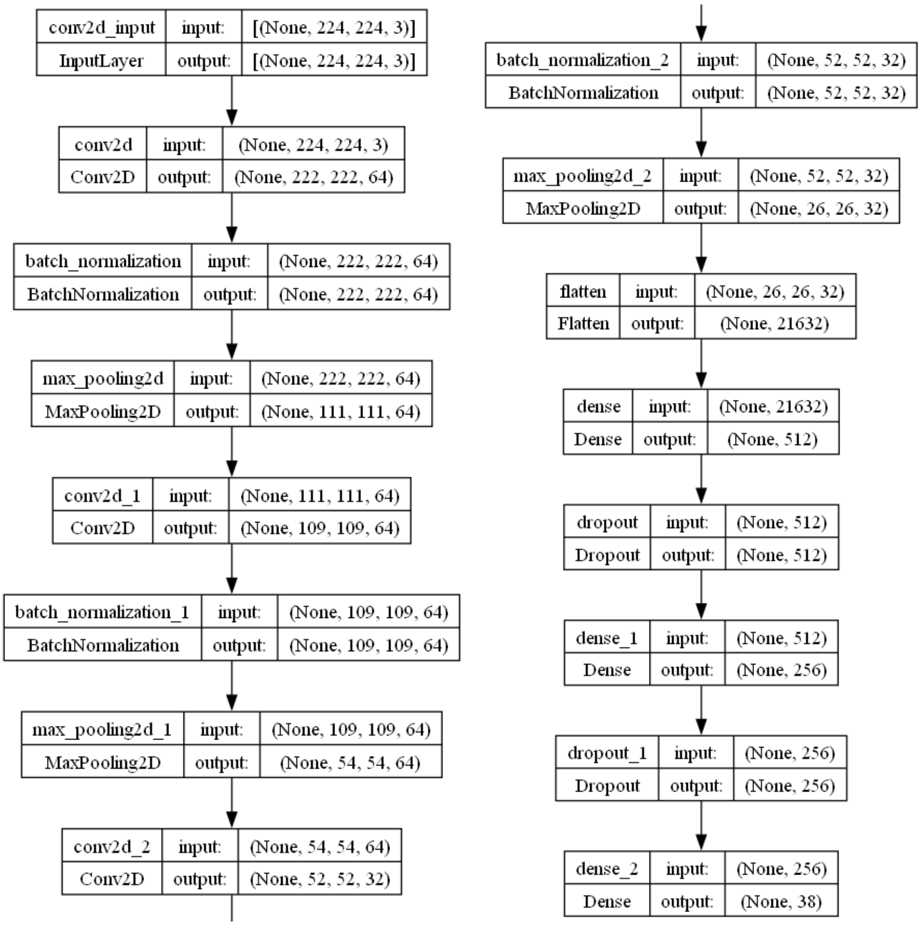

Figure 4.9: Model 4

V. RESULTS AND DISCUSSION

5.1 Model 1:

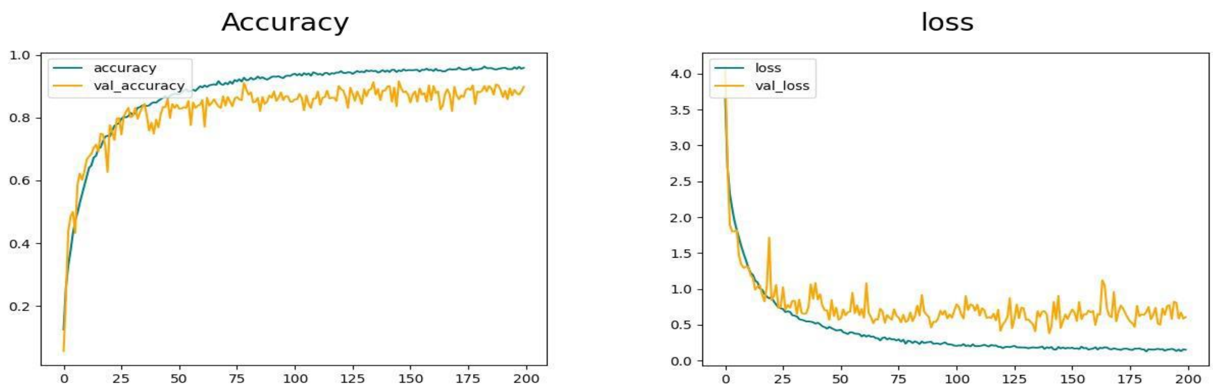

Figure 5.1: Model 1 Accuracy and loss graph

The above graphs represent the accuracy and loss of model respectively obtained after training the model 1 on 90:05:05 split for 200 epochs. In accuracy graph, the X-axis represents the number of epochs and the Y-axis represents the accuracy of model. From the graph, there is sharp increase in accuracy between epochs 0 and 25. At the end of the 200 epoch the accuracy is 0.9336 and the validation accuracy is 0.8856. The almost 4.8% difference between accuracy and validation accuracy shows very minor level of overfitting. So, the model is generalizing well to the unseen data. In loss graph, the X-axis represents the number of epochs and the Y-axis represents the loss of model. From the graph, at the end of the 200 epoch the loss is 0.252 and the validation accuracy is 0.7131.

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 5, pp 190-204, May-2024

https://doi.org/10.47001/IRJIET/2024.805029

5.2 Model 2:

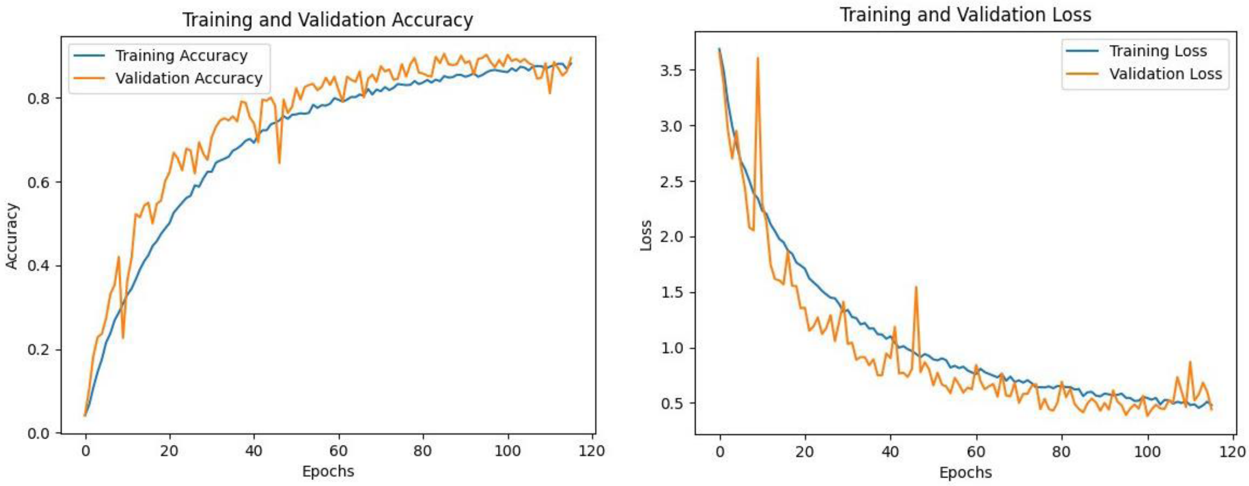

Figure 5.2: Model 2 Accuracy and loss graph

The above graphs represent the accuracy and loss of model respectively obtained after training the model 2 on 90:05:05 split for 116 epochs. From the accuracy graph, at the end of the 116 epoch the accuracy is 0.8823 and the validation accuracy is 0.8955. The almost 1.32% difference between accuracy and validation accuracy shows very minor level of overfitting. So, the model is generalizing well to the unseen data. Both training and validation loss can be seen decreasing as the model trains for enough number of epochs. The training loss at the 116 epochs is 0.4789 whereas the validation loss is 0.4406.

5.3 Model 3:

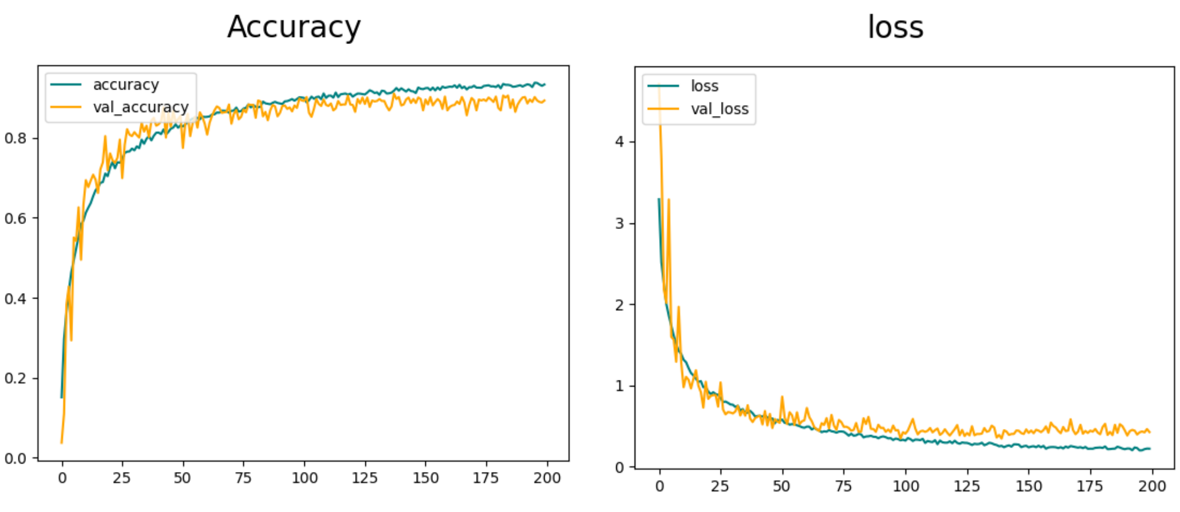

Figure 5.3: Model 3 Accuracy and loss graph

The above graphs represent the accuracy and loss of model respectively obtained after training the model 3 on 80:10:10 split for 200 epochs. From the graph, at the end of the 200 epoch the accuracy is 0.9336 and the validation accuracy is 0.8936. The almost 4% difference between accuracy and validation accuracy shows very minor level of overfitting. So, the model is generalizing well to the unseen data.

In the loss graph, X-axis represents the number of epochs and the Y-axis represents the loss of model. From the graph, at the end of the 200 epoch the loss is 0.2215 and the validation loss is 0.4256.

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 5, pp 190-204, May-2024

https://doi.org/10.47001/IRJIET/2024.805029

5.4 Model 4:

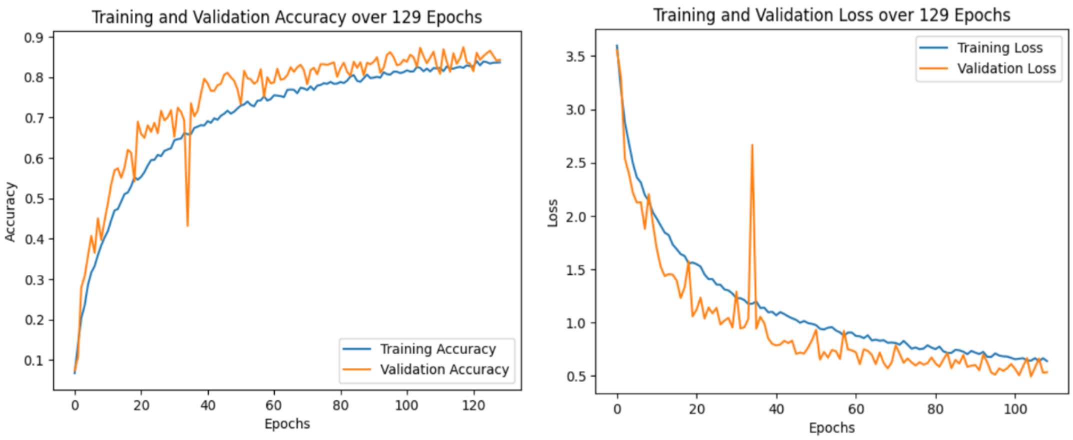

Figure 5.4: Model 4 Accuracy and loss graph

The above graphs represent the accuracy and loss of model respectively obtained after training the model 3 on 80:10:10 split for 129 epochs. From the graph, at the end of the 129 epoch the accuracy is 0.8364 and the validation accuracy is 0.8428. The almost 0.64% difference between accuracy and validation accuracy shows very minor level of overfitting. In the loss graph, the X-axis represents the number of epochs and the Y-axis represents the loss of model. From the graph, at the end of the 200 epoch the loss is 0.5893 and the validation loss is 0.5505.

5.5 Evaluation on test datasets

After training, all four models were evaluated using the test dataset. Testing accuracy refers to the proportion of correct predictions made on a test dataset, expressed as a percentage.

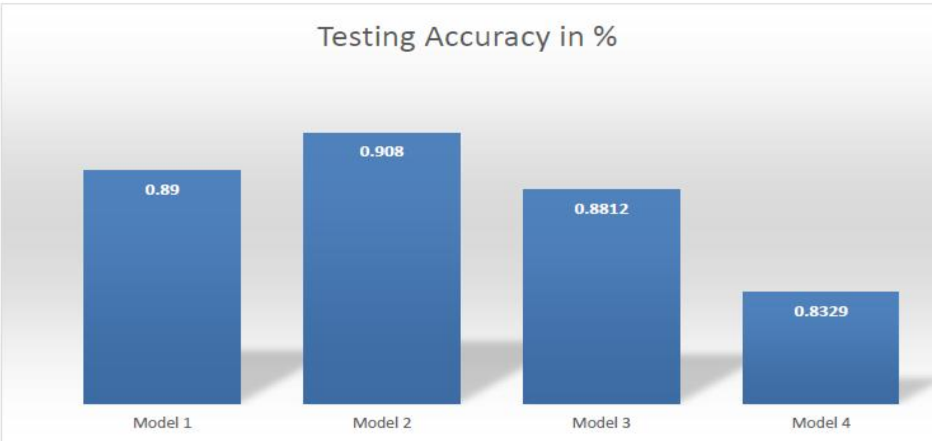

Figure 5.5: Testing Accuracy of models

The model 1 uses the dataset which was distributed in a 90:05:05 ratio for training, testing, and validation and run for 200 epochs to produce testing accuracy of 0.89(89.3%).

The model 2 uses the dataset which was distributed in a 90:05:05 ratio for training, testing, and validation and run for 116 epochs to produce testing accuracy of 0.908(90.8%).

The model 3 uses the dataset which was distributed in an 80:10:10 ratio for training, testing, and validation and run for 200 epochs to produce testing accuracy of 0.8812(88.12%).

The model 4 uses the dataset which was distributed in an 80:10:10 ratio for training, testing, and validation and run for 200 epochs to produce testing accuracy of 0.8329(83.29%).

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 5, pp 190-204, May-2024

https://doi.org/10.47001/IRJIET/2024.805029

The experiment with various configuration yield different result as:

Table 1: Model experiment with different configuration

Model	Dataset	Convolution layer				Fully Connected Layer				Output layer	Dropout	No of epochs	Training accuracy	Validation accuracy	Testing accuracy
		1st	2nd	3rd	4th	1st	2nd	3rd	4th						
1	90:05:05	64	64	32	-	512	256	-	-	38	0.1	200	0.9336	0.8856	0.893
2	90:05:05	64	64	32	32	512	256	128	64	38	0.3	116	0.8823	0.8955	0.908
3	80:10:10	64	64	32	16	512	256	-	-	38	0.3	200	0.9336	0.8936	0.8812
4	80:10:10	64	64	32	16	256	128	64	-	38	0.3	129	0.8364	0.8428	0.8329

5.6 Model Evaluation on Unseen Images

From the four models considered, the top-performing model, characterized by its highest test accuracy, was selected (model 2) for further assessment on unseen test dataset.

The model's performance is evaluated on the testing dataset which is spitted from the total dataset. The testing dataset contains the 5% of each species of the total dataset that the model is not trained with before. Evaluation of test metrics is more important than training metrics, as the test metrics indicate the model's ability to generalize on unseen data from the real world. The model performed very well on the test dataset, achieving an accuracy of 0.9080. The weighted F1 score, which considers precision and recall for all classes, was 0.9084. This indicates that the model has a good balance between precision and recall for all classes and is effective at classifying bird species.

	precision	recall	f1-score	support
Abbott's Babbler Malacocincla abbotti	1.00	0.88	0.93	8
Black Bittern (Dupetor flavicollis)	0.69	0.92	0.79	12
Blue-eared Kingfisher Alcedo meninting	1.00	1.00	1.00	11
Blue-naped Pitta Pitta nipalensis	1.00	0.78	0.88	9
Broad-billed Warbler Tickellia hodgsoni	1.00	0.86	0.92	7
Cheer Pheasant (Catreus wallichii)	0.90	0.90	0.90	10
Chestnut Munia Lonchura atricapilla	1.00	0.92	0.96	12
Cinereous Vulture Aegypius monachus	0.79	0.92	0.85	12
Golden Babbler Stachyris chrysaea	1.00	0.90	0.95	10
Gould's Shortwing Brachypteryx stellata	1.00	0.88	0.93	8
Great Bittern Botaurus stellaris	0.79	0.92	0.85	12
Great Hornbill (Buceros bicornis)	0.91	1.00	0.95	10
Great Slaty Woodpecker Mulleripicus pulverulentus	1.00	0.83	0.91	12
Ibisbill Ibidorhyncha struthersii	1.00	1.00	1.00	7
Indian Courser Cursorius coromandelicus	1.00	1.00	1.00	13
Indian Grassbird - Graminicola bengalensis	0.83	1.00	0.91	5
Indian Nightjar Caprimulgus asiaticus	1.00	0.77	0.87	13
Knob-billed Duck Sarkidiornis melanotos	0.89	0.80	0.84	10
Northern Pintail Anas acuta	0.92	0.92	0.92	12
Painted Stork Mycteria leucocephala	0.82	0.90	0.86	10
Purple Cochoa Cochoa purpurea	1.00	0.80	0.89	5
Red-headed Trogon Harpactes erythrocephalus	1.00	1.00	1.00	13
Red-headed Vulture Sarcogyps calvus	0.90	0.90	0.90	10
Red-necked Falcon Falco chicquera	1.00	1.00	1.00	12
Ruby-cheeked Sunbird Anthreptes singalensis	1.00	0.92	0.96	13
Rusty-fronted Barwing Actinodura egertoni	0.90	0.75	0.82	12
Saker Falcon Falco cherrug	0.67	0.60	0.63	10
Silver-eared Mesia Leiothrix argentauris	1.00	1.00	1.00	13
Slaty-legged Crake Rallina eurizonoides	0.91	1.00	0.95	10
Spot-bellied Eagle Owl Bubo nipalensis	0.81	1.00	0.90	13
Sultan Tit Melanochlora sultanea	0.92	0.92	0.92	12
Swamp Francolin Francolinus gularis	0.79	1.00	0.88	15
Tawny-bellied Babbler Dumetia hyperythra	0.83	0.91	0.87	11
Thick-billed Green Pigeon Treron curvirostra	0.92	0.92	0.92	12
White-throated Bulbul Alophoixus flaveolus	0.91	1.00	0.95	10
White-throated Bushchat Saxicola insignis	1.00	0.88	0.93	8
Yellow-rumped Honeyguide - Indicator xanthonotus	1.00	0.83	0.91	12
Yellow-vented Warbler Phylloscopus cantator	0.88	0.88	0.88	8
accuracy			0.91	402
macro avg	0.92	0.90	0.91	402
weighted avg	0.92	0.91	0.91	402

Figure 5.6: Accuracy, Precision, Recall and F1-Score of best performing model

ISSN (online): 2581-3048
Volume 8, Issue 5, pp 190-204, May-2024
https://doi.org/10.47001/IRJIET/2024.805029

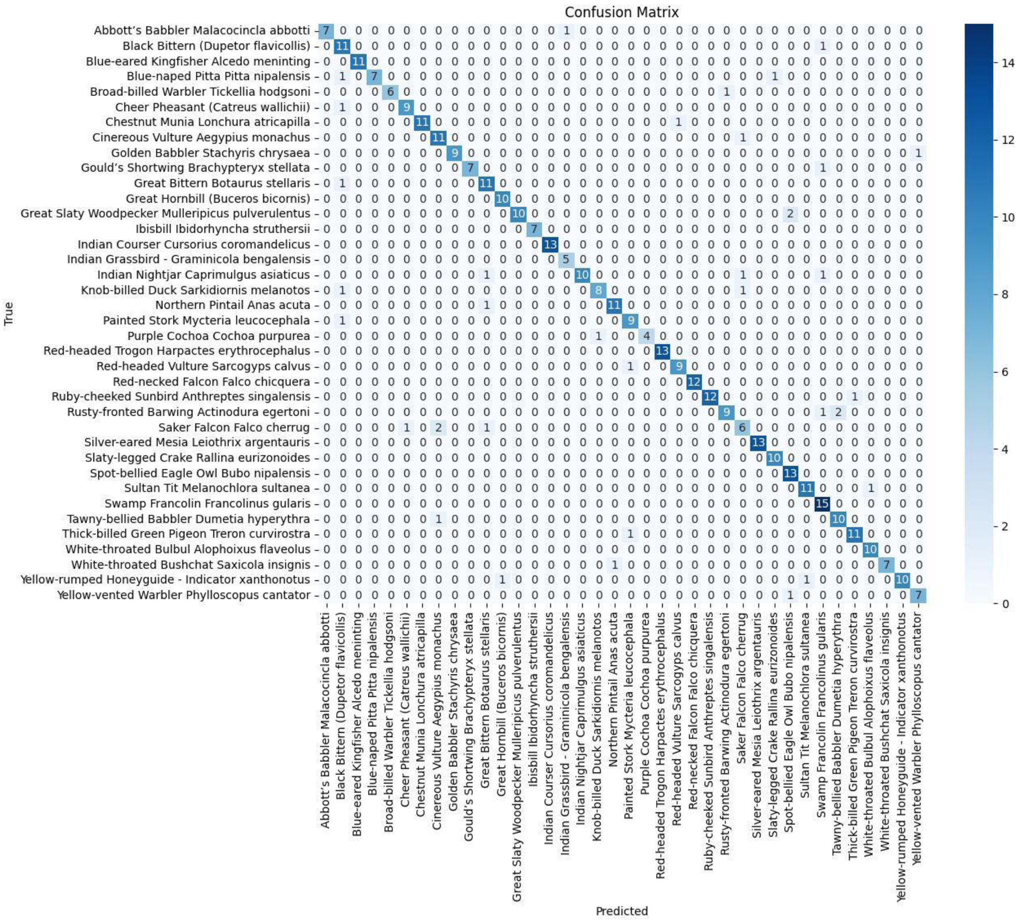

Figure 5.7: Confusion matrix of the test dataset

5.7 Web Application

The most effective model was deployed in the web application, where it leveraged the probabilities of 38 bird species to make predictions. The Web application is build using Django and Tailwind CSS. It mainly consists of 3 pages. The Home page serves as the welcoming gateway, introducing the application's purpose.

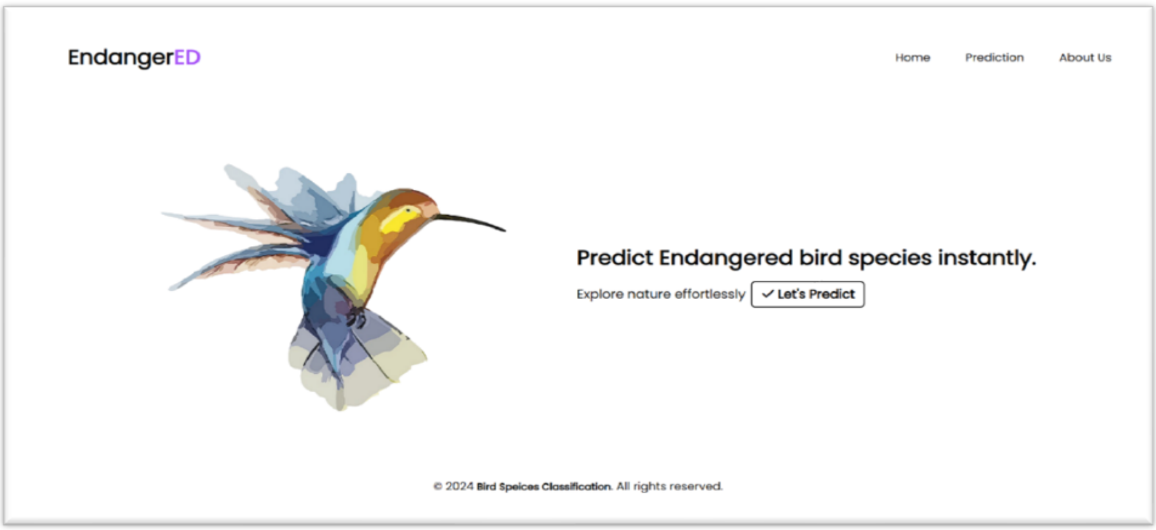

Figure 5.8: Home page

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 5, pp 190-204, May-2024

https://doi.org/10.47001/IRJIET/2024.805029

The prediction page serves as the page where the user provides input image of the bird to predict bird species.

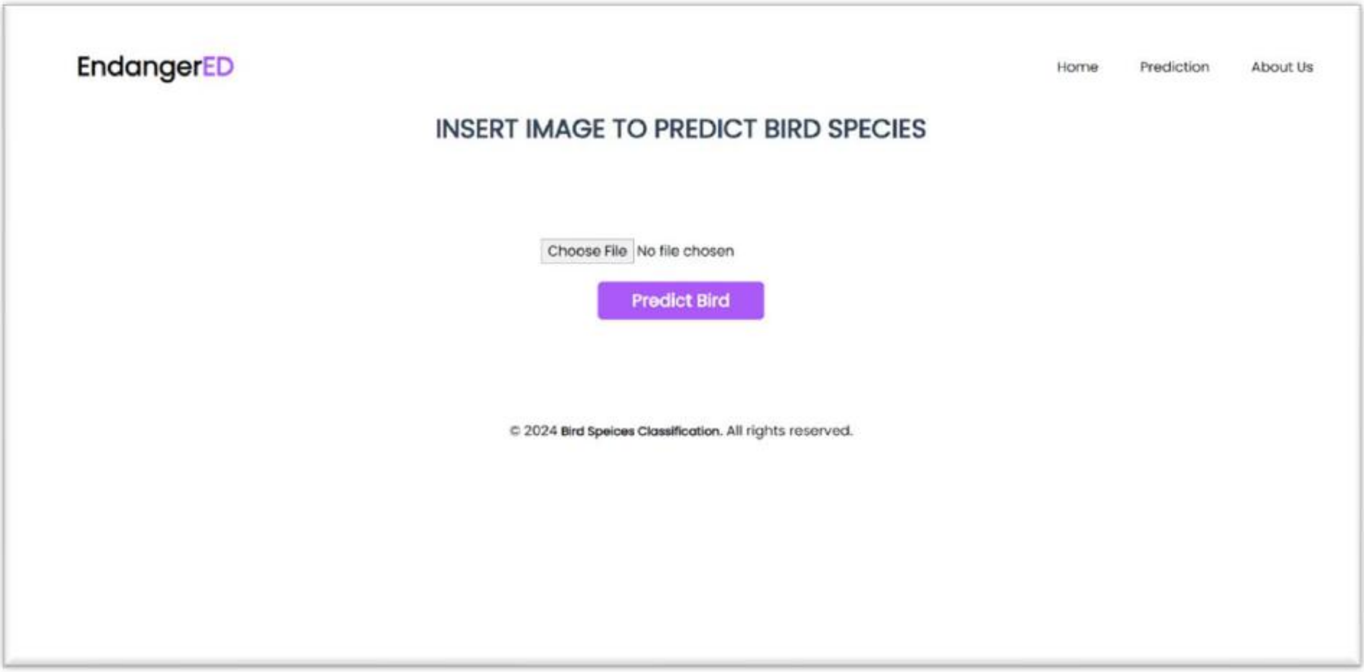

Figure 5.9: Prediction Page

The about page provide the details about the project about how the project is made and how many bird species can this project classify.

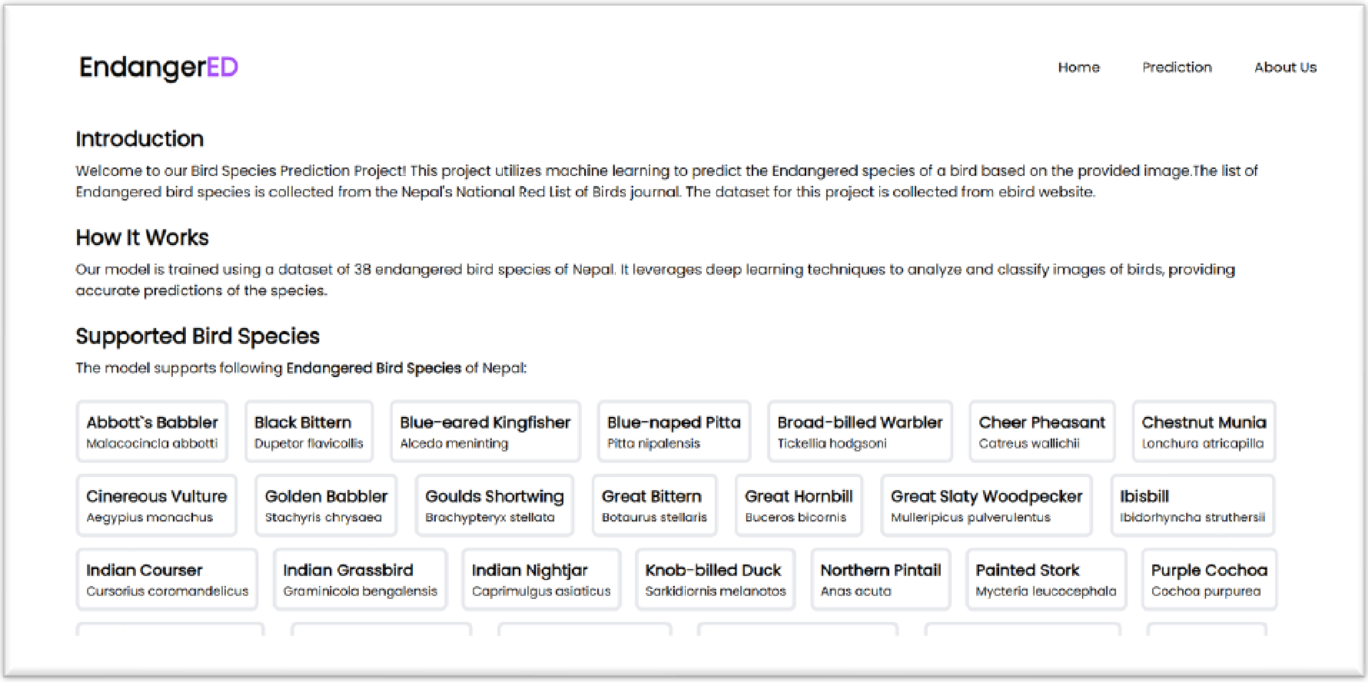

Figure 5.10: About Us page

International Research Journal of Innovations in Engineering and Technology (IRJIET)

ISSN (online): 2581-3048

Volume 8, Issue 5, pp 190-204, May-2024

https://doi.org/10.47001/IRJIET/2024.805029

5.8 Prediction

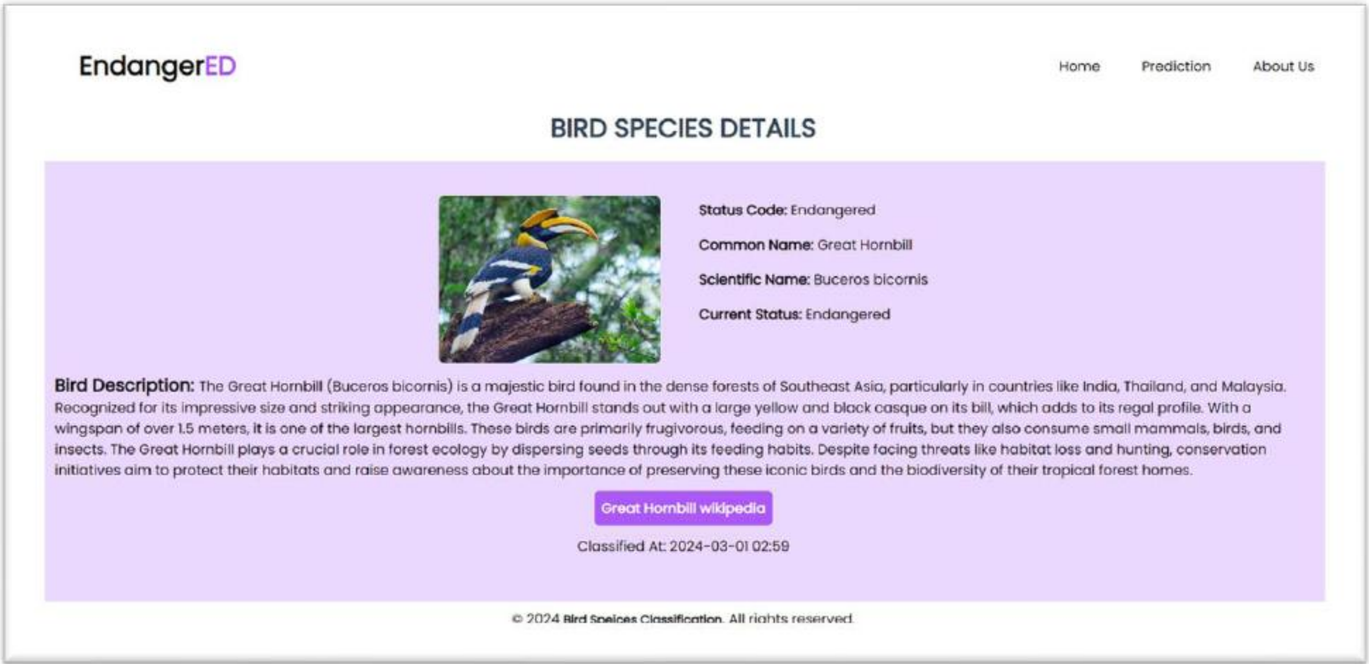

Figure 5.11: Predicted Bird Species details page

5.9 Evaluation of the Model on Non-Bird Images

Figure 5.12: Prediction on non-bird image

It demonstrated the capability to distinguish between images containing birds and those without, based on the identified bird species.

VI. CONCLUSION

In this research project, our main objective was to develop a deep learning system capable of classifying Endangered Bird Species of Nepal using CNN models. To achieve this goal, we curated a high-quality dataset consisting of images of Nepal's 38 Endangered Bird Species sourced from the ebird.org website. We then experimented with four different CNN models to identify the most effective approach for classification. The key findings of our experimentation revealed notable variations in testing accuracies across the different models. Model 2 exhibited the highest testing accuracy of 90.8%, followed closely by Model 1 with an accuracy of 89.3%. Models 3 and 4 achieved accuracies of 88.12% and 83.29%, respectively. These results demonstrate the effectiveness of CNN models in accurately classifying Endangered Bird Species based on image data. However, a major limitation encountered during the project was the availability of bird images, which impacted the diversity and size of the dataset. Despite this challenge, our findings have significant implications for the conservation of bird species in Nepal and beyond. The developed deep learning system has

ISSN (online): 2581-3048
Volume 8, Issue 5, pp 190-204, May-2024
https://doi.org/10.47001/IRJIET/2024.805029

the potential to aid conservation efforts by facilitating the identification and monitoring of Endangered Bird Species. Looking ahead, future improvements to this project could focus on enhancing the quality and diversity of the dataset. By collecting more comprehensive and representative images of bird species, we can further improve the accuracy and reliability of the classification system. Additionally, exploring advanced data augmentation techniques and incorporating transfer learning approaches may contribute to better model performance and generalization. Overall, this project highlights the promising applications of deep learning in biodiversity conservation and underscores the importance of leveraging technology to protect endangered species and preserve natural ecosystems.

ACKNOWLEDGEMENT

We must express our deep and sincere gratitude to the Institute of Engineering, Pashchimanchal Campus and the Department of Electronics and Computer Engineering for giving us this wonderful platform and an opportunity to utilize and build our skills. We are deeply indebted to all our teachers who have provided us with valuable knowledge that has really helped us to reach here and explore the horizon further. We also would like to express our sincere thanks to our supervisor, Asst. Prof. Nabin Lamichhane for providing us with the necessary guidance and his valuable time for the improvement of our research project.

We would also like to acknowledge our family and friends and would like to thank them for being helpful and supportive which has worked as a motivation and created a secure space for us.

REFERENCES

[1] C. Inskipp, H. S. Baral, T. Inskipp, A. P. Khatiwada, M. P. Khatiwada, P. L. Poudyal and R. Amin, "Nepal's National Red List of Bird," Journal of Threatened Taxa, vol. 9, no. 1, pp. 9700-9722, 2017.

[2] J. Martinsson, "Bird Species Identification using Convolution Neural Networks," Chalmers University of Technology and University of Gothenburg, Sweden, 2017.

[3] J. Niemi and J. T. Tanttu, "Deep Learning Case Study for Automatic Bird Identification," Applied Sciences, vol. 8, no. 11, 2017.

[4] M. A. Tayal, A. Mangrulkar, P. Waldey and C. Dangra, "Bird Identification by Image Recognition," Helix, vol. 8, no. 6, pp. 4349-4352, 2018.

[5] A.Marini, J. Facon and A. L. Koerich, "Bird Species Classification Based on Color Features," 2013 IEEE International Conference on Systems, Man, and Cybernetics, pp. 4336-4341, 2013.

[6] X. Xiao, M. Yan, C. Ji, S. Basodi and Y. Pan, "Efficient Hyperparameter Optimization in Deep Learning Using a Variable Length Genetic Algorithm," arXiv, vol. 1, 2020.

[7] B. Khatiwada, B. P. Subedi, N. Duwal and R. Gautam, "Samrakshyan: An Endangered Birds Recognition Portal," March 2023. [Online]. Available: https://www.linkedin.com/posts/rewangautam_research-birds-conservation-activity-7031145284067807232-HQ8o/.

[8] "Ebird," 2021. [Online]. Available: https://media.ebird.org/catalog?taxonCode=grehor1&sort=rating_rank_desc&mediaType=photo.

[9] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel and B. Thirion , "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825-2830, 2011.

[10] B. S. Poudel, "Wildlife Research and Monitoring in Nepal: An Overview," The Initiation, vol. 2, no. 1, pp. 22-32, 2006.

[11] N. Subedi, N. Paudel, M. Chhetri, S. Acharya, and N. Lamichhane, "Nepali Image Captioning: Generating Coherent Paragraph-Length Descriptions Using Transformer," J. Soft Comput. Paradigm, vol. 6, no. 1, pp. 70–84, Mar. 2024, doi: 10.36548/jscp.2024.1.006.

Citation of this Article:

Sumit Pant, Sandip Shrestha, Abhishek Aryal, Ozan Wagle, Nabin Lamichhane, "Classification of Endangered Bird Species of Nepal Using Deep Learning", Published in *International Research Journal of Innovations in Engineering and Technology - IRJIET*, Volume 8, Issue 5, pp 190-204, May 2024. Article DOI https://doi.org/10.47001/IRJIET/2024.805029

****