Faculty of Computer Science and Information Technology

EVALUATION OF DATASET DIVERSITY TOWARD THE PERFORMANCE OF TRANSFER LEARNING MODEL FOR BIRD IDENTIFICATION

Caryn Chong Ke Xin

Bachelor of Computer Science with Honours (Computational Science)

2025

EVALUATION OF DATASET DIVERSITY TOWARD THE PERFORMANCE OF TRANSFER LEARNING MODEL FOR BIRD IDENTIFICATION

CARYN CHONG KE XIN

This project is submitted in partial fulfilment of the

requirements for the degree of

Bachelor of Computer Science with Honours (Computational Science)

Faculty of Computer Science and Information Technology

UNIVERSITI MALAYSIA SARAWAK

2025

PENILAIAN KEPELBAGAIAN SET DATA KE ARAH PRESTASI MODEL PEMBELAJARAN PEMINDAHAN UNTUK PENGENALPASTIAN BURUNG

CARYN CHONG KE XIN

Project ini merupakan salah satu keperluan untuk

Ijazah Sarjana Muda Sains Komputer dan Teknologi

Maklumat (Sains Komputer)

Fakulti Sains Komputer dan Teknologi Maklumat

UNIVERSITI MALAYSIA SARAWAK

2025

UNIVERSITI MALAYSIA SARAWAK

THESIS STATUS ENDORSEMENT FORM

TITLE EVALUATION OF DATASET DIVERSITY TOWARD THE PERFORMANCE OF TRANSFER LEARNING MODEL FOR BIRD IDENTIFICATION

ACADEMIC SESSION: 2024/2025

CARYN CHONG KE XIN
(CAPITAL LETTERS)

hereby agree that this Thesis* shall be kept at the Centre for Academic Information Services, Universiti Malaysia Sarawak, subject to the following terms and conditions:

1. The Thesis is solely owned by Universiti Malaysia Sarawak
2. The Centre for Academic Information Services is given full rights to produce copies for educational purposes only
3. The Centre for Academic Information Services is given full rights to do digitization in order to develop local content database
4. The Centre for Academic Information Services is given full rights to produce copies of this Thesis as part of its exchange item program between Higher Learning Institutions [or for the purpose of interlibrary loan between HLI]
5. ** Please tick (√)

☐	CONFIDENTIAL	(Contains classified information bounded by the OFFICIAL SECRETS ACT 1972)
☐	RESTRICTED	(Contains restricted information as dictated by the body or organization where the research was conducted)
√	UNRESTRICTED	

Validated by

(AUTHOR'S SIGNATURE) (SUPERVISOR'S SIGNATURE)

Permanent Address

No. 311 Lorong Stampin 10A, 93350
Kuching, Sarawak.

Date: 20/7/2025 Date: 27/7/2025

Note * Thesis refers to PhD, Master, and Bachelor Degree
** For Confidential or Restricted materials, please attach relevant documents from relevant organizations / authorities

DECLARATION

I, Caryn Chong Ke Xin with the matric number of 79010, hereby solemnly declare that the report titled '**Evaluation of Dataset Diversity Toward the Performance of Transfer Learning Model for Bird Identification**' is prepared and completed authentically by me in partial fulfilment of the requirements for the degree of Bachelor of Computer Science with Honours (Computer Science) is a record of an original work carried out by me under the guidance of Associate Professor Dr Lee Nung Kion. I further declare that the work reported in this project has not been previously or concurrently submitted for any other degree at UNIVERSITI MALAYSIA SARAWAK (UNIMAS) or any other institutions. I declare that I have not copied or plagiarized from any other sources except for quotations and citations which have been duly acknowledged.

CARYN CHONG KE XIN (79010)

Faculty of Computer Science and Information Technology

UNIVERSITI MALAYSIA SARAWAK

ACKNOWLEDGEMENT

Above all, I want to sincerely thank Associate Professor Dr Lee Nung Kion, my supervisor, for his essential advice, inspiration, and steadfast support during this final year project. His knowledge and perceptions have been crucial in forming this project into what it is now.

Next, I want to express my sincere gratitude to Dr Wee Bui Lin, my examiner, for her insightful remarks and helpful criticism on my final year project. Her careful analysis and insightful recommendations were very helpful in improving the overall quality of my work.

Furthermore, I want to express my profound appreciation to Professor Dr. Wang Yin Chai, the coordinator of the Final Year Project, for his outstanding leadership, thorough preparation, and unambiguous direction in helping to facilitate the overall framework of this project. His detailed planning of the FYP process, helpful advice, and timely updates have made sure I stayed on course and fulfilled all deadlines.

Lastly, my family and friends deserve a particular thank you for their unwavering support and understanding during this journey. The project is completed with gratefulness. Without their support, I wouldn't have been able to keep my momentum and came so far.

ABSTRACT

The rapid advancement of machine learning techniques has significantly improved the accuracy of bird identification systems, particularly through the application of transfer learning. This project investigates the impact of dataset diversity, specifically defined as the presence or absence of image background elements on the performance of transfer learning models in bird identification tasks. By analyzing a single bird dataset with two variants (with background and without background), this project aims to establish a correlation between visual background and model efficacy. Convolutional neural networks (CNNs) that have already been trained, like EfficientNetB0, were used to refine models on datasets with various levels of diversity. The model's performance will be evaluated using metrics including accuracy, precision, recall, and F1 score. This research highlights the importance of creating balanced and diverse datasets to enhance the effectiveness of transfer learning models in bird identification, thereby facilitating better wildlife monitoring and conservation initiatives.

ABSTRAK

Kemajuan pesat teknik pembelajaran mesin telah meningkatkan ketepatan sistem pengenalan burung dengan ketara, terutamanya melalui aplikasi pembelajaran pemindahan. Projek ini menyiasat kesan kepelbagaian set data—yang secara khusus ditakrifkan sebagai kewujudan atau ketiadaan latar belakang imej—ke atas prestasi model pembelajaran pemindahan dalam tugas pengenalpastian burung. Dengan menggunakan satu set data burung yang sama dalam dua variasi (dengan latar belakang dan tanpa latar belakang), kajian ini bertujuan untuk menentukan korelasi antara variasi visual latar belakang dan keberkesanan model. Rangkaian saraf konvolusi (CNN) yang telah dilatih, seperti EfficientNetB0, digunakan untuk memperhalusi model pada set data dengan pelbagai peringkat kepelbagaian. Prestasi model akan dinilai menggunakan metrik termasuk ketepatan, ketepatan, ingatan semula dan skor F1. Penyelidikan ini menyerlahkan kepentingan mencipta set data yang seimbang dan pelbagai untuk meningkatkan keberkesanan model pembelajaran pemindahan dalam pengenalpastian burung, dengan itu memudahkan pemantauan hidupan liar dan inisiatif pemuliharaan.

TABLE OF CONTENTS

LIST OF TABLES

TABLE OF FIGURES

LIST OF ABBREVIATIONS

BMP Bitmap Image Files

CNNs Convolutional Neural Networks

CSS Cascading Style Sheets

GUI Graphical User Interface

HOG Histogram of Oriented Gradients

HTML Hypertext Markup Language

IDE Integrated Development Environment

JPEG Joint Photographic Experts Group

JPG Joint Photographic Experts Group

MFCCs Mel-frequency cepstral coefficients

OOD Out-of-Distribution

PNG Portable Network Graphic

RAM Random Access Memory

SIFT Scale-Invariant Feature Transform

SVM Support Vector Machines

UML Unified Modeling Language

VS Code Visual Studio Code

WEBP Web Picture

CHAPTER 1: INTRODUCTION

1.1 Project Title

Evaluation of Dataset Diversity Toward the Performance of Transfer Learning Model for Bird Identification

1.2 Introduction/ Background

Bird identification is a crucial task in the fields of ecology, conservation, and wildlife management. Accurate and efficient bird species classification provides valuable insights into biodiversity, habitat monitoring, and environmental changes. The rise of machine learning, particularly deep learning techniques, has significantly advanced the capabilities of automated bird identification systems. One such approach that has shown promising results is transfer learning, where pre-trained models developed for one task are adapted and fine-tuned for the specific problem of bird species classification.

However, the effectiveness of transfer learning models in bird identification can be influenced by specific characteristics of the training data. This includes visual features such as background context. Usually, models may unintentionally learn from irrelevant background information rather than focusing solely on the bird species. This raises important questions about how the presence or absence of background in training images, as a specific form of dataset diversity, affects model performance and generalization. In this study, "dataset diversity" is narrowly defined as the visual presence or absence of background elements in otherwise identical bird image datasets.

While previous research indicates that models trained on diverse datasets tend to exhibit better generalization to unseen data compared to those trained on homogeneous datasets (Yang, Shen, & Xu, 2024), few have specifically isolated the effect of background elements in otherwise identical datasets. Therefore, this study fills the gap by evaluating how background content influences the performance of transfer learning model, EfficientNetB0 when trained on a single bird dataset in two settings, one with image backgrounds, and the other without image backgrounds.

Moreover, birds often serve as flagship species for conservation efforts due to their visibility and public appeal. This widespread interest can drive funding and support for conservation initiatives, making bird research not only scientifically relevant but also socially impactful (Garnett, Ainsworth, & Zander, 2018).

To address these interconnected challenges, this project investigates the impact of background variation on transfer learning model performance in bird species classification. By analyzing model behavior across two dataset variants, the study aims to provide practical insights into dataset preparation strategies that support more robust and generalizable bird identification systems.

1.3 Problem Statement

The identification of birds is crucial for tracking biodiversity and supporting conservation initiatives. Accurate bird identification is essential for monitoring ecosystem health, as birds are often indicators of environmental changes. Historically, bird identification relied on expert knowledge and manual observation, but the advent of machine learning has transformed this field, offering automated solutions that can analyze vast amounts of data.

However, the performance of these models depends not only on algorithmic strength but also on the characteristics of the training data. One key factor is the presence of background information in bird images, which can either assist or hinder the model's learning. For example, if the model unintentionally relies on background cues, such as trees, sky, or lighting patterns, it may fail to generalize when the background context changes.

Although previous research has shown that models trained on homogeneous datasets often struggle to generalize to new species and environmental conditions (Yang, Shen, & Xu, 2024), few have specifically examined how removing or retaining background information affects the classification performance. This gap raises the need to evaluate the impact of background context on model accuracy and robustness in a controlled setting.

Therefore, this project seeks to address the research question:

How to prepare and diversify the dataset to improve the performance of transfer learning models in identifying bird species?

By investigating this relationship, the study aims to provide insights that could enhance the robustness and applicability of automated bird identification systems.

1.4 Objectives

1. To assess the performance of the EfficientNetB0 model in identifying bird species across a dataset with varying background characteristics.
2. To investigate the relationship between dataset diversity and model performance,

determining which characteristics of the datasets most significantly impact the accuracy of bird species classification.

1.5 Brief Methodology

This project used an Agile-inspired approach methodology to support structured experimentation and iterative development. Although the project is mainly focusing on research-based, the iterative nature of model development, including dataset preparation, training configuration adjustments, and performance evaluation aligned well with the Agile principles. This method allows for ongoing improvement based on performance metrics and experimental findings, which is particularly helpful in deep learning processes involving image classification. Furthermore, Agile allows for the parallel development of a minimal local deployment, ensuring the project remains adaptable and focused on delivering incremental outcomes (Chan, 2024).

The dataset used in this study is the “20 UK Garden Birds” image dataset sourced from Kaggle. It consists of 2841 images across 20 species (Mahony, 2023). Two dataset variants were prepared for experimentation:

1. With Background: Original bird images with natural elements such as trees, branches or sky.
2. Without Background: The same images as the “With Background” dataset setting, just the background content was removed, focusing solely on the bird itself.

Since Kaggle notebook was used throughout the training process, and the dataset used is sourced from Kaggle as well, the dataset was directly added to the Kaggle notebook. After data collection part, data processing will be performed. This includes image resizing, rotation, flipping and so on to enhance robustness.

The EfficientNetB0 model was selected for this study due to its state-of-the-art performance in image classification tasks. According to Tan and Le (2019), EfficientNet has been shown to achieve top performance on various image classification benchmarks.

To evaluate the impact of background information, each dataset setting was trained with three different model configurations using EfficeintNetB0 model, pre-trained on ImageNet:

1. Configuration 1: Dropout 0.3, Batch Size 16
2. Configuration 2: Dropout 0.5, Batch Size 32
3. Configuration 3: Dropout 0.7, Batch Size 64

5-fold cross validation was used to ensure the robustness and generalizability of the results. The performance was then evaluated using the standard classification metrics, which are accuracy, precision, recall, and F1-score.

Among the with background dataset setting, the best-performing model after evaluation will be selected to proceed with the local deployment as a simple web application using Streamlit. This minimal deployment enabled users to upload the UK bird image and receive a predicted species label, serving as a demonstration of the model's real-world applicability.

For the data analysis part, data was analysed using statistical methods to determine the relationship between dataset diversity and model performance. Comparative analysis was conducted to identify trends in model performance metrics across different datasets backgrounds. The detail explanation of the methodology will be discussed in the Chapter 3.

1.6 Scope of Project

This project investigates the impact of dataset diversity on the performance of transfer learning models, specifically the EfficientNetB0 model, for bird species identification.

The scope has been purposefully reduced to ensure a controlled and research-oriented evaluation with a single dataset in two scenarios. The project does not aim to develop a fully functional bird recognition application, but rather to demonstrate the results of the model evaluation process.

Things that will be included in the project:

1. Use of a single dataset: 20 UK Garden Birds from Kaggle.
2. Creation of two dataset settings: with background and without background.
3. Model training using EfficientNetB0 with three different configurations (changing of dropout and batch size)
4. 5-fold cross validation and evaluation of model performance using metrics such as accuracy, precision, recall, and F1 score.
5. Comparative analysis to compare model performance across dataset variants.
6. Minimal local deployment of the best-performing model among the with background dataset setting using Streamlit for demonstration.

Things that will not be included in the project:

1. Use of multiple bird datasets.

2. Exploration of models other than EfficientNetB0.
3. Large-scale user testing or usability studies on the deployed app.

1.7 Significance of Project

This project contributes to the field of machine learning for biodiversity applications by addressing an underexplored factor: the significance of background information in training datasets for bird species classification. As transfer learning has always been performing well in image identification tasks, limited research or studies examined how visual background features influence the model generalization and accuracy.

By comparing the performance of a transfer learning model, EfficientNetB0 across two variants of the same dataset, with and without background, this project aids in providing practical insights into dataset preparation strategies. These findings are useful for both research and practical applications when generalization and data variability pose significant challenges.

Overall, the project's findings should help guide future dataset curation and model training procedures for wildlife image classification tasks, particularly when training data is limited or highly variable in visual context.

1.8 Project Schedule

Table 1. 1: Table for Project Schedule

Task	Start Date	Duration (days)	End Date
Project Planning	25/11/2024	21	23/12/2024
Data Collection and Preparation	24/12/2024	26	28/01/2025

Model Development	29/01/2025	31	12/03/2025
Model Evaluation	13/03/2025	21	10/04/2025
Results and Analysis	11/04/2025	21	09/05/2025
Deployment	10/05/2025	16	30/05/2025
Documentation	31/05/2025	16	20/06/2025

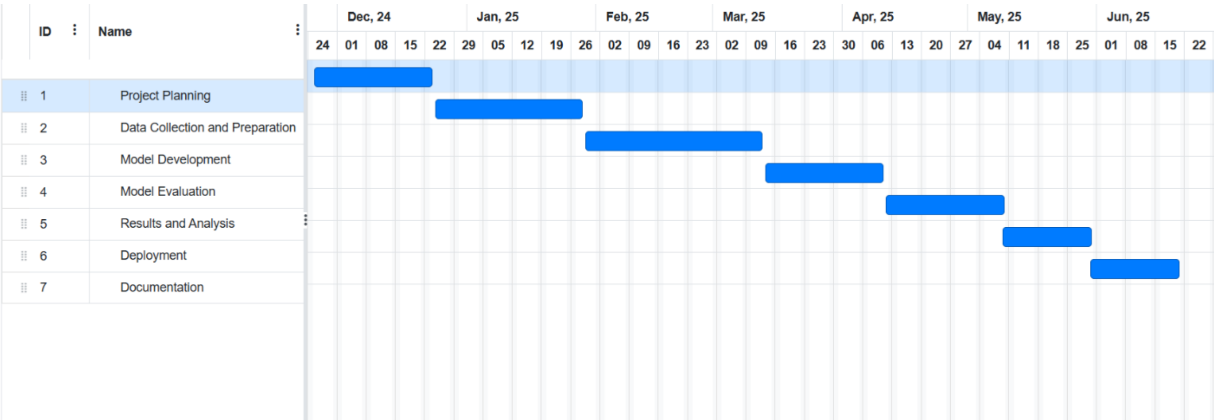

Figure 1. 1: Gantt Chart for Project Timeline

1.9 Expected Outcome

The project aims to comprehensive analysis of how dataset diversity, defined in this study as the variation between image datasets with and without background, affects the performance of the EfficientNetB0 model in bird identification. Through systematic experimentation, the study will evaluate classification metrics across both dataset versions to determine the impact of visual context on model accuracy and generalization.

1.10 Project Outline

This section will briefly give an overview of the report structure, which have a total of six chapters and a brief description of what each chapter did.

1.10.1 Chapter 1: Introduction

This project categorized into two phases, Final Year Project 1 (FYP 1) and Final Year Project 2 (FYP 2). In chapter 1, this project starts off with Introduction, Problem Statement, Objectives, Methodology, Scope of Project, Significance of Project, Project Schedule, Expected Outcome, and lastly Project Outline.

1.10.2 Chapter 2: Literature Review

This chapter evaluated key literature on bird identification using deep learning, focusing on the benefits of transfer learning and EfficientNetB0 for fine-grained categorization. It also demonstrated how dataset properties, notably background context, affect model performance, which supports the study's aim. In addition, existing bird identification systems and deployment approaches have been reviewed to provide practical context. These ideas form the basis for the research methodology in the following chapter.

1.10.3 Chapter 3: Requirements Analysis and Design

The chapter examines the bird identification project's requirements and design considerations, such as Agile process, model architecture, software and hardware requirements, functional and non-functional requirements, and a user-friendly prediction interface. It also demonstrates the system's use case and GUI using Figma as the prototype, as well as how users can recognize bird photographs with the local Streamlit program.

1.10.4 Chapter 4: Implementation and Results

Chapter 4 outlines the implementation of the model training pipeline, dataset preparation, and evaluation process. It also presents the experimental results and concludes with the local deployment of the best-performing model.

1.10.5 Chapter 5: Result and Analysis

Experimental results and analysis of the model performance across different configurations and dataset settings of the project will be conducted here. It compares the performance of models trained with and without background data, assesses classification metrics, and analyses trends. It also compares the findings to those from similar studies to help contextualize the project's contributions.

1.10.6 Chapter 6: Conclusion

In Chapter 6, it summarizes the key contributions, limitations, future works that can be implemented, and a conclusion.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

This chapter reviews the existing work on bird species identification using machine learning, with a special emphasis on the use of transfer learning. It investigates how different dataset properties, particularly the presence or lack of background elements, affect the performance and generalization capabilities of deep learning models. This review provides a theoretical and empirical framework for assessing how dataset variety affects classification accuracy in bird identification tasks.

2.2 Background Study

Manual observation of physical characteristics like as plumage color, beak shape, and size were common in the early phases of bird identification. While instructive, these methods relied heavily on observer knowledge and ambient variables, resulting in inconsistency and misidentification (Lederer, n.d.; Mertins, n.d.).

With advances in computer vision, standard machine learning approaches such as Support Vector Machines (SVM), Decision Trees, and Random Forests were used to automate bird species classification. These techniques rely primarily on constructed image features such as colour histograms and edge descriptors (Caruana & Niculescu-Mizil, 2006). However, their performance was frequently constrained by the quality and specificity of feature extraction approaches as Scale-Invariant Feature Transform (SIFT) and Histogram of Oriented Gradients (HOG) (Dalal & Triggs, 2005; Lowe, 2004).

Recent advances in deep learning, notably Convolutional Neural Networks (CNNs), have greatly improved image classification skills by learning visual information automatically. CNN-based models such as VGGNet, ResNet, and EfficientNet have been found to outperform classical techniques, particularly when used with transfer learning. Transfer learning enables pre-trained models on big image datasets such as ImageNet to be fine-tuned on smaller, domain-specific datasets, saving training time while maintaining high accuracy (P. D. et al., 2019).

Tan and Le (2019) developed EfficientNet, which uses a compound scaling strategy to balance model depth, width, and resolution, resulting in great performance with fewer parameters. Its versatility and efficiency make it ideal for resource-constrained contexts and tiny datasets, such as bird identification taks.

Modern bird classification systems have significantly improved both accuracy and generalizability by employing pre-trained CNNs and carefully prepared datasets, particularly when tackling dataset-specific challenges such as background variation—a central focus of this study.

2.3 Role of Deep Learning in Bird Image Identification

Recent developments in deep learning, notably convolutional neural networks (CNNs), have significantly improved automated bird species recognition. These systems use large-scale annotated datasets to extract visual elements like plumage patterns, forms, and colors from raw pixel data. CNNs, unlike typical machine learning algorithms that rely on hand-crafted features, may derive hierarchical representations from input images, which improves classification accuracy (Krizhevsky, Sutskever, & Hinton, 2012).

CNNs are particularly useful in bird identification since they can discern small variations between species. However, the performance of these models might be highly dependent on dataset properties like lighting, resolution, and background context. Background components, such as trees or the sky, may contribute noise or unwanted correlations into the model's learning, limiting its capacity to generalize across contexts (Beery, Van Horn, & Perona, 2018).

To solve these issues, transfer learning has emerged as an effective technique. Using pre-trained models like EfficientNet, researchers may fine-tune deep architectures using bird-specific datasets even when data availability is low. EfficientNet's compound scaling mechanism allows it to balance accuracy and computing economy, making it ideal for smaller, domain-specific applications such as the bird identification task investigated in this paper (Tan & Le, 2019).

This study specifically investigates the impact of background context in bird images, comparing model performance on two datasets: one with natural backgrounds and one with backgrounds removed. This work, which focuses on image-only inputs and measures performance changes owing to background fluctuation, gives useful insights for dataset design in image classification tasks involving fine-grained visual categories such as bird species.

2.4 Comparative Analysis of CNN Architectures in Bird Classification

In fine-grained image classification tasks like bird species recognition, choosing an appropriate CNN architecture has a considerable impact on model accuracy, generalizability, and computational performance. Numerous studies have compared the performance of architectures such as VGG, ResNet, MobileNet, and EfficientNet.

In a thorough evaluation of several deep learning architectures on a bird classification dataset, Santiago Martinez (2024) found that EfficientNet-B0 outperformed ResNet-50, VGG-16, and Inception-v3 in terms of top 1 accuracy and training efficiency, highlighting its optimal trade-off between depth, width, and resolution. Similarly, Farou and Reslan changed EfficientNet-B0's top layer for fine-tuning and discovered that it consistently beat MobileNet, DenseNet201, and ResNet152V2 in fine-grained bird species classification (Farou & Reslan, n.d.).

Chaturvedi et al. (2024) further showed that EfficientNet's lightweight nature and compound scaling make it particularly successful for datasets with high inter-class similarity and slight visual distinctions, such as those found in agricultural bird populations. Meanwhile, Neeli et al. (2023) demonstrated that, while models such as VGG and ResNet provide competitive performance, EfficientNet-B0 provided higher accuracy while maintaining computational efficiency, making it suitable for both research evaluation and minimal local deployment contexts (Neeli, 2023).

These comparison studies support the choice of EfficientNet-B0 for this project, which balances performance and complexity while allowing for scalable experimentation. Furthermore, its ability to adapt to dataset variability makes it ideal for assessing the influence of background variety in image datasets.

2.5 Impact of Dataset Characteristics on Transfer Learning Performance

The quality and structure of picture datasets have a significant effect on transfer learning model performance, particularly for fine-grained classification tasks like as bird species identification. CNN models are known for learning both relevant and irrelevant visual cues. This includes the background features, which, if related with certain classes, can mislead the model toward shortcut learning by focusing on background rather than subject features (Ghani et al., 2025).

Several research studied how background variability in training data influences generalization. For example, Yang et al. (2022) discovered that CNNs performed much better on test sets with similar backgrounds to the training data, indicating overfitting to background context. To address this, they tested dataset augmentation and background-blurred images to force models to acquire object-centric attributes (Yang et al., 2022).

Besides that, image resolution also has a significant impact on classification accuracy. Wang (2024) conducted an optimization study comparing different input resolutions and discovered that both extremely high and very low resolutions lowered accuracy, underlining the necessity of resolution scaling that matches the pre-trained model's expectations (Wang, 2024).

Another challenge is class imbalance, which arises when some bird species are overrepresented in the collection. Manna et al. (2023) shown that imbalance leads to biased classification performance that benefits the majority classes. They used oversampling and class-weighting techniques to balance training signals (Manna et al., 2023).

Additionally, poor lighting, motion blur, and compression failures are all examples of image noise that can affect performance. Alswaitti et al. (2022) found that models trained on clean datasets performed badly when evaluated on noisy or low-light images, and they proposed adopting noise-robust augmentation during training.

In short, these findings collectively validate the study's rationale, which was to experimentally investigate how the presence or absence of background elements in the dataset affects classification performance using a transfer learning-based CNN architecture. This method assures that performance gains are truly the result of model generalization rather than background correlation.

2.6 Existing Bird Identification Web Application

In this section, there are three existing Bird Identification Web Application/ Mobile Application, namely Netvue Web Application, Nyckel Web Application and Bird Identifier by Picture Mobile Application are reviewed and compared. Then, a comparison table is created to evaluate the existing system with the proposed system. The subsections and tables that follow serve the purpose to provide an overview of existing bird identification aids. This gives background for the research's minimal deployment component. These systems are not benchmarked against this project, and they are only reviewed to understand common deployment features.

2.6.1 Netvue Web Application

Netvue Web Application Link: Free AI Recognition Tool on Birds! Learn About What Birds are Coming to You.

According to Netvue (2024), Netvue established in 2010, has developed a strong presence in the smart home and security camera market. One of their notable initiatives is the Birdfy sub-brand, which contains 6000 and more bird species recognition, focuses on connecting users with nature through innovative bird feeder cameras. This brand features a bird identification function on its website that utilizes advanced AI technology, allowing users to identify various bird species by uploading photos of the birds. The features are explained with brief descriptions in the following.

Features of Netvue Web Application

The web application allows users to upload bird photos in JPEG, JPG, and PNG formats as shown in Figure 2.1.

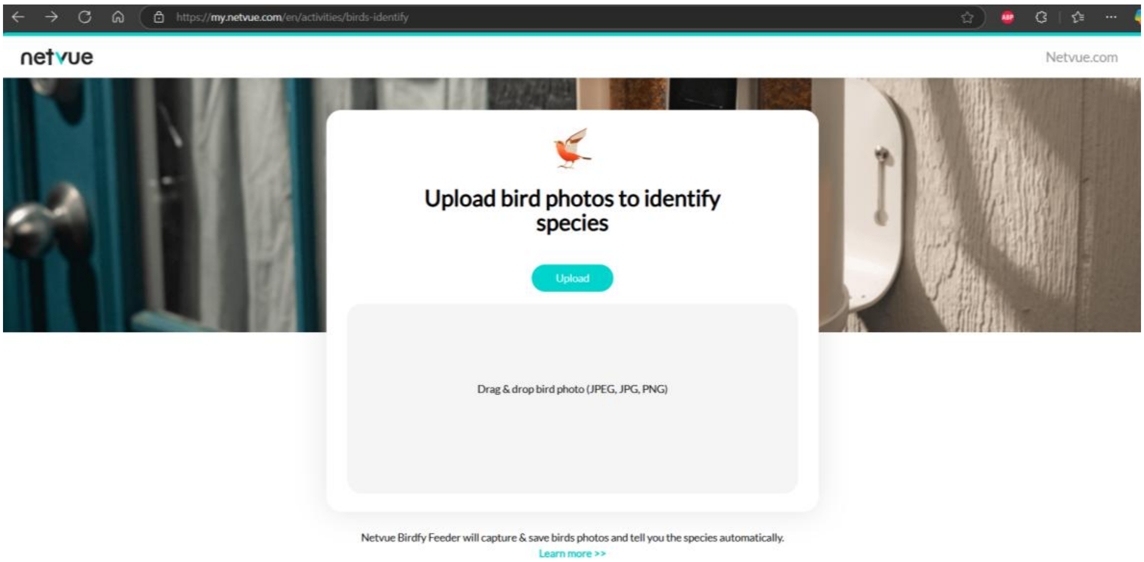

Figure 2. 1: User Interface of Netvue Web Application

After uploading, it utilizes AI algorithms or community input to provide accurate bird identification results. Then, the output displays the name of the identified bird species along with the model's confidence level in the identification, indicating how certain it is about the result, shown in Figure 2.2.

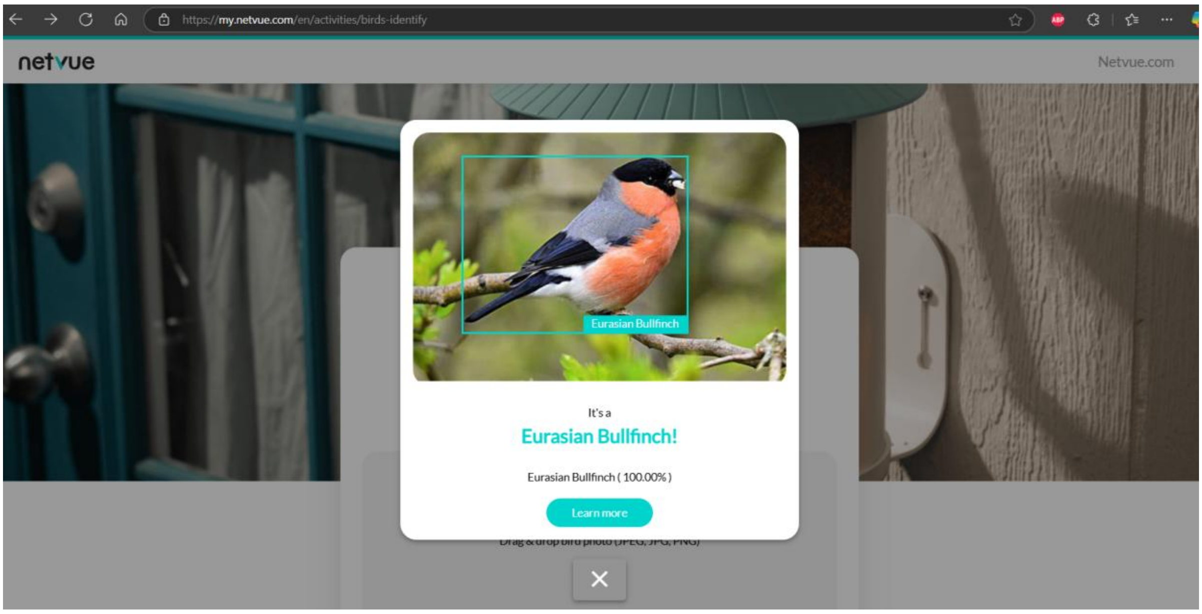

Figure 2. 2: Sample Output Result for Netvue Web Application

2.6.2 Nyckel Web Application

Nyckel Web Application Link: Identify birds using AI | Nyckel

Nyckel Web Application is a bird identification tool offered by Nyckel, which is a company founded in 2021. Nyckel has developed a machine learning research and engineering platform that allows for the rapid development and integration of ML capabilities into applications. The platform facilitates the classification and semantic search of images, text, and tabular data, and it can be deployed immediately, enabling clients to invoke functions in real time using an API (PitchBook, 2024). The features are explained with brief descriptions in the following.

Features of Nyckel Web Application

As you can see from Figure 2.3 below, the website allows the user to upload an image to identify birds quickly in the format of JPG, PNG, BMP or WEBP.

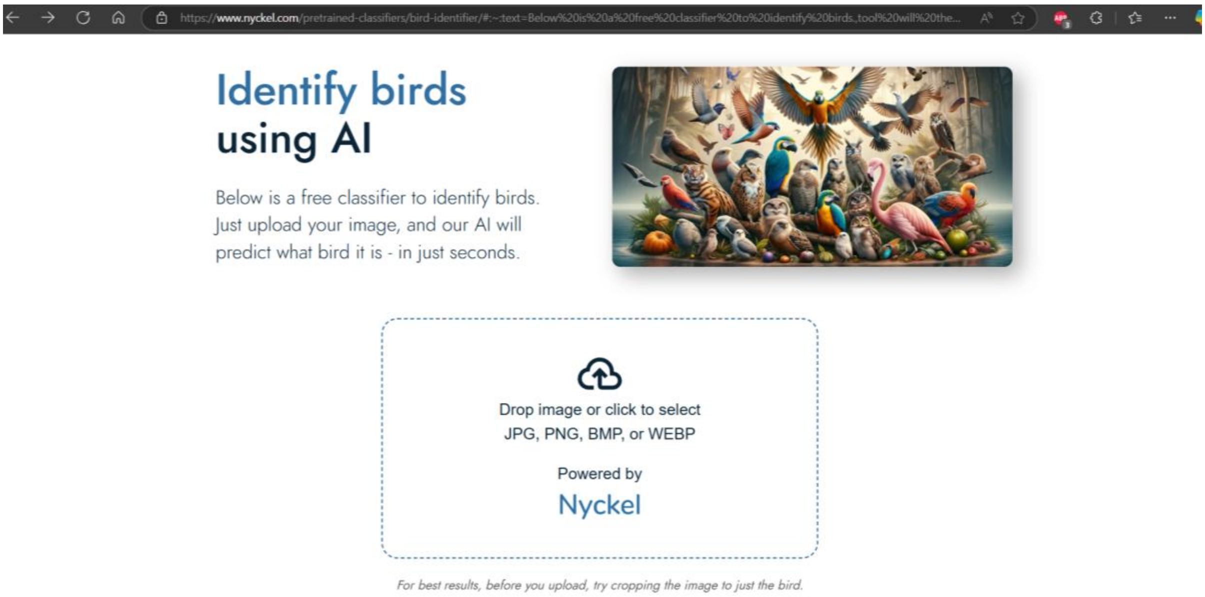

Figure 2. 3: User Interface of Nyckel Web Application

After uploading the image of the bird, the AI will predict the species based on its training with a dataset of 525 species with 291 labels (Nyckel, n.d.). It will display a confidence score for its prediction on the model it used as shown in Figure 2.4.

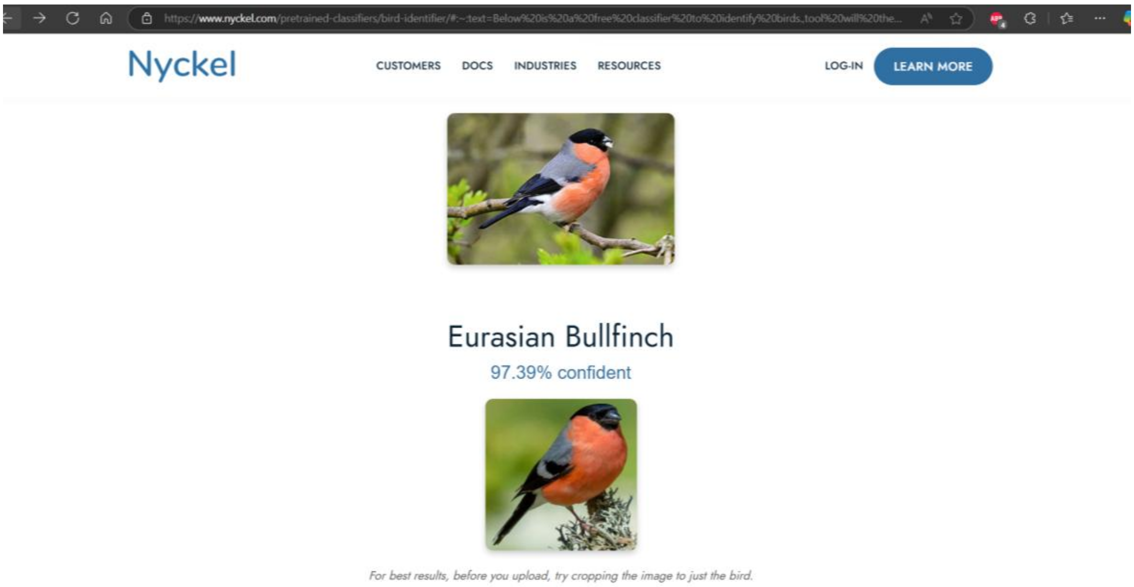

Figure 2. 4: Sample Output Result for Nyckel Web Application

2.6.3 Bird Identifier by Picture Mobile Application

Installation Link: Bird identifier by picture by Harsh Vardhan

Bird Identifier by Picture is a mobile application designed to help users identify various bird species using artificial intelligence. Developed by Harsh Vardhan, it was released on February 29, 2024. The app allows users to upload photos of birds, utilizing advanced machine learning algorithms to analyse these images and provide detailed information about the identified species (Vardhan, 2024). While the app offers a limited number of credits for free usage, users may need to upgrade or make in-app purchases for unlimited access. The following outlines the features along with concise descriptions.

Features of Bird Identifier by Picture Mobile Application

From Figure 2.5, it can be shown that there is a '+' icon, which is used for the upload of the bird image.

Figure 2. 5: User Interface of Bird Identifier by Picture Mobile Application

After selecting the picture of the bird to be uploaded, the AI analyses the unique characteristics of the bird depicted in the image. Then, the application provides detailed information about the identified bird species as shown in Figure 2.6.

Figure 2. 6: Sample Output Result for Bird Identifier by Picture Mobile Application

2.7 Comparison Features of Existing Systems and Proposed System

Table 2. 1: Comparison Features between the Existing Systems and Proposed System

Features	Netvue	Nyckel	Bird Identifier by Picture	Proposed System
Web application	✓	✓	✗	✓
Mobile application	✗	✗	✓	✗
Uses AI/ Deep learning	✓	✓	✓	✓
AI chat box	✗	✗	✓	✗
Upload image	✓	✓	✓	✓
Take picture	✗	✗	✗	✗
Analyse single bird per picture	✓	✓	✓	✓
Analyse multiple birds in single picture	✓	✗	✓	✗
Show accuracy/ Confidence level	✓	✓	✗	✓
Details of the species	✗	✗	✓	✓
View history data	✗	✗	✓	✗
In-app purchases	✗	✗	✓	✗

Table 2.1 shows the comparison features between the existing bird identification system with the proposed system.

2.8 Proposed Solution

The evaluation on the pros and cons for the three existing Bird Identification Systems are stated in the Table 2.2. Proposed Bird Identification System will consider adopting the useful and effective features from existing systems.

Table 2. 2: Comparison Pros and Cons between the Existing Systems and Proposed System

Types of system / Characteristics	**Netvue**	**Nyckel**	**Bird Identifier by Picture**	**Proposed System**
Advantages	-Free to use -Enable to analyse multiple birds in single picture -Display accuracy	-Free to use -Display accuracy	-Contains AI chat box -Enable to analyse multiple birds in single picture	-Free to use -Contains details of the bird species
Disadvantages	-Does not contain details of the bird species	-Does not contain details of the bird species	Contain in-app purchases	NA

2.9 Summary

This chapter investigated the evolution of bird species identification methods, emphasizing the transition from traditional manual and classical machine learning approaches to deep learning-based solutions. It underscored the importance of convolutional neural networks (CNNs), namely EfficientNetB0, in fine-grained picture categorization applications with subtle inter-species variations.

Comparative studies of CNN designs proved EfficientNet's superior balance of accuracy and processing efficiency, which justified its selection for this study. Furthermore, recent research has shown that dataset characteristics like as background presence, image resolution, and class imbalance have a substantial impact on the performance of transfer learning models. This contributes to the study's experimental focus on assessing background context effects in bird image datasets.

While not crucial to the research methodology, existing bird identification web applications and deployment approaches were reviewed to give context for real-world applications. These references serve as an outline for the project's final deployment, which includes a simple web-based prototype for species identification.

This literature, adopted together, serves as the foundation for the proposed research methodology, which will be discussed in the following chapter.

CHAPTER 3: REQUIREMENT ANALYSIS AND DESIGN

3.1 Introduction

This chapter gives an overview of the system and research requirements for the project, including both functional and non-functional aspects. It also discusses on the model training design, methodology of the process, and minimal deployment flow.

3.2 Agile-Inspired Approach

Although this project mainly focused on research-based, an Agile-inspired approach was adopted to guide the workflow throughout the experimental process. Tasks such as dataset preprocessing, model training, evaluation, and results analysis were completed iteratively in specific cycles. Model performance was examined after each cycle, and adjustments (such as hyperparameter tuning or training parameters) were performed in response to evaluation results. This iterative process facilitated systematic refining and allowed for effective time and milestone management throughout the project.

Agile techniques in AI and machine learning projects have been shown to promote flexibility and continual improvement, especially in contexts where model outputs can guide development direction. Studies have highlighted that Agile frameworks help accelerate convergence, improve accuracy, and better align experimental design with performance insights in AI model development (Kancharla & Pannala, 2019; Nikfal, 2024).

3.3 Methodology

This section will be discussed on the system methodology, basically the workflow of the whole experimental process starting from data collection to the selection of best-performing model for local deployment among the with background dataset after evaluation.

3.3.1 Process Flow Diagram

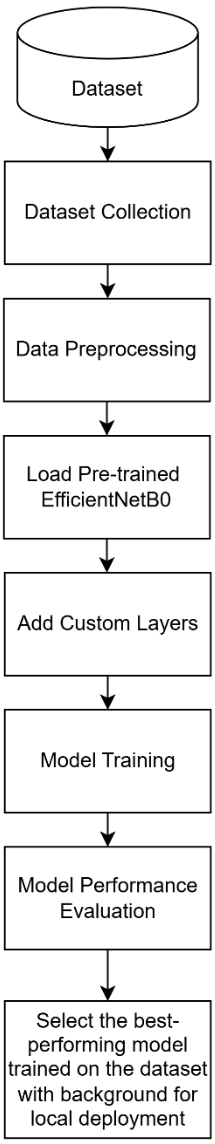

Figure 3. 1: Process Flow Diagram for Bird Identification System

Figure 3.1 shows the process flow diagram for the bird identification system. The model's images will be evaluated using the pre-trained CNN model, EfficientNetB0. This model will help leverage transfer learning to improve accuracy and efficiency in image classification tasks. The detailed explanation for every process will be written below in each subsection from 3.3.1.1 to subsection 3.3.1.7.

3.3.1.1 Dataset Collection

The datasets used this for bird identification system was sourced from Kaggle, which is the "20 Uk Garden Birds" dataset which offers a total of 2841 images across 20 species, with 150 images each for most of the species. The dataset consists of two different variants, "With Background" and "Without Background" dataset. The images are grouped by species and background presence. Some sample of bird datasets with background and without background are shown in the Figure 3.2 to Figure 3.5.

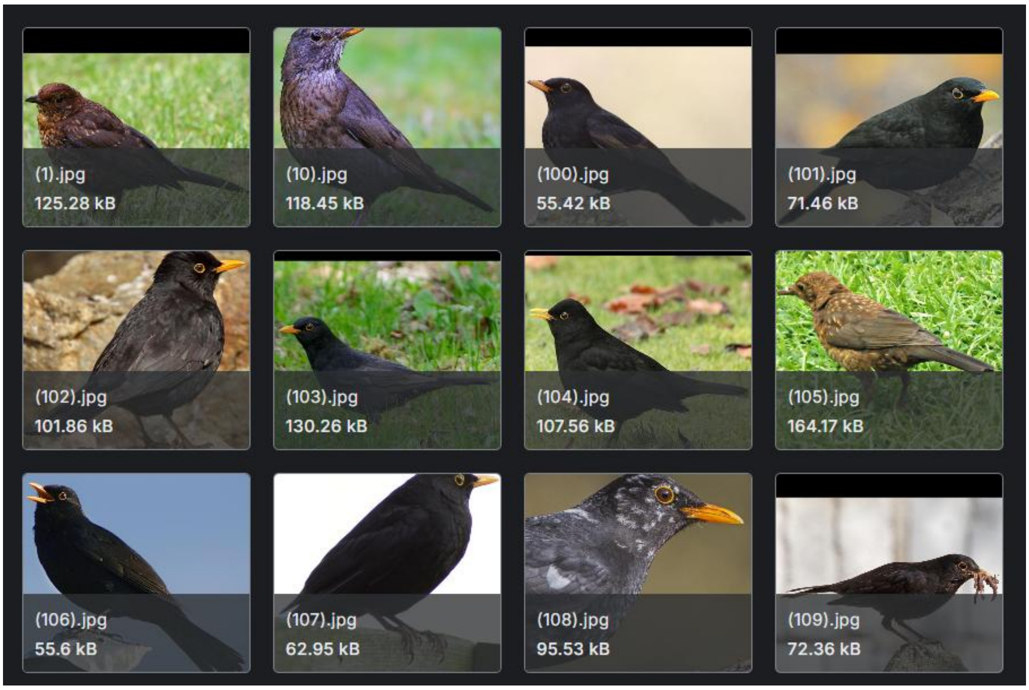

Figure 3. 2: Blackbird Dataset (with Background)

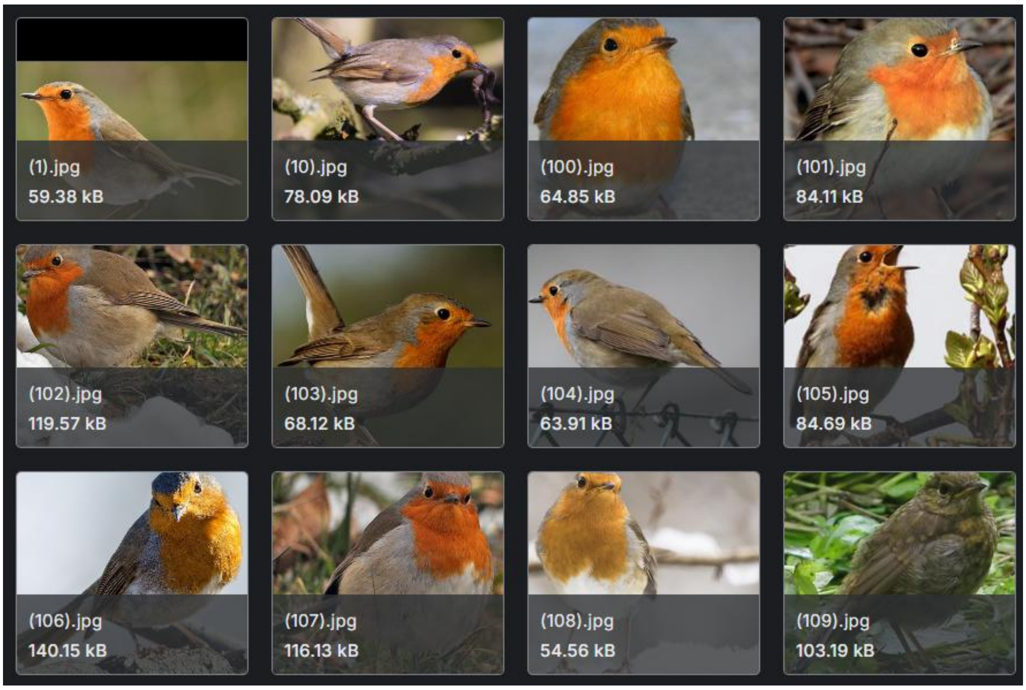

Figure 3. 3: Robin Dataset (with Background)

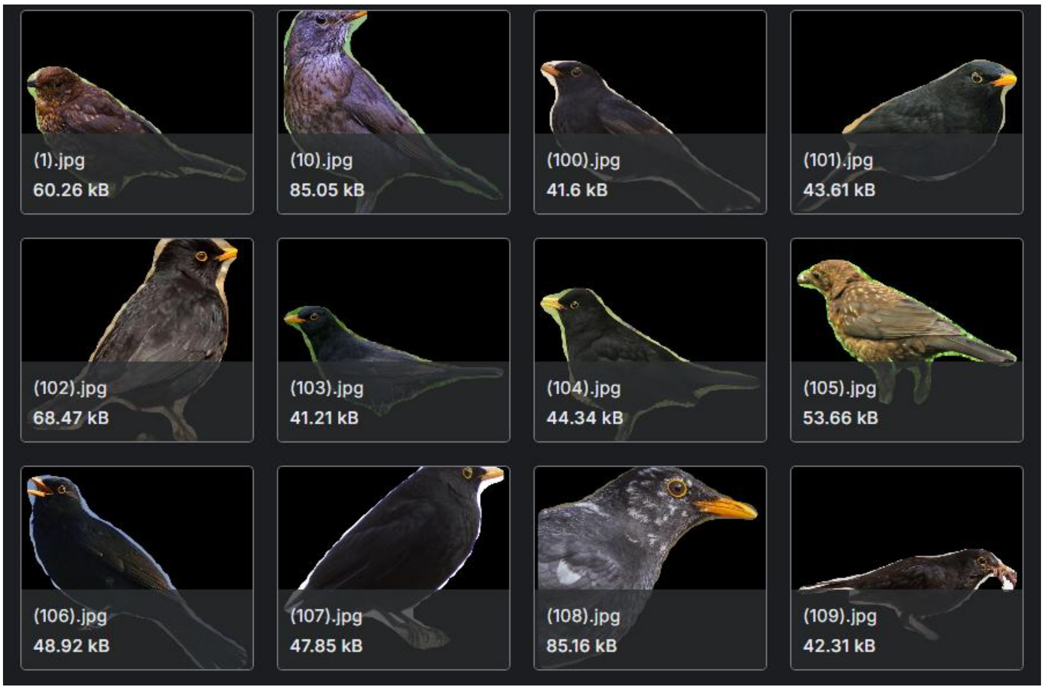

Figure 3. 4: Blackbird Dataset (without Background)

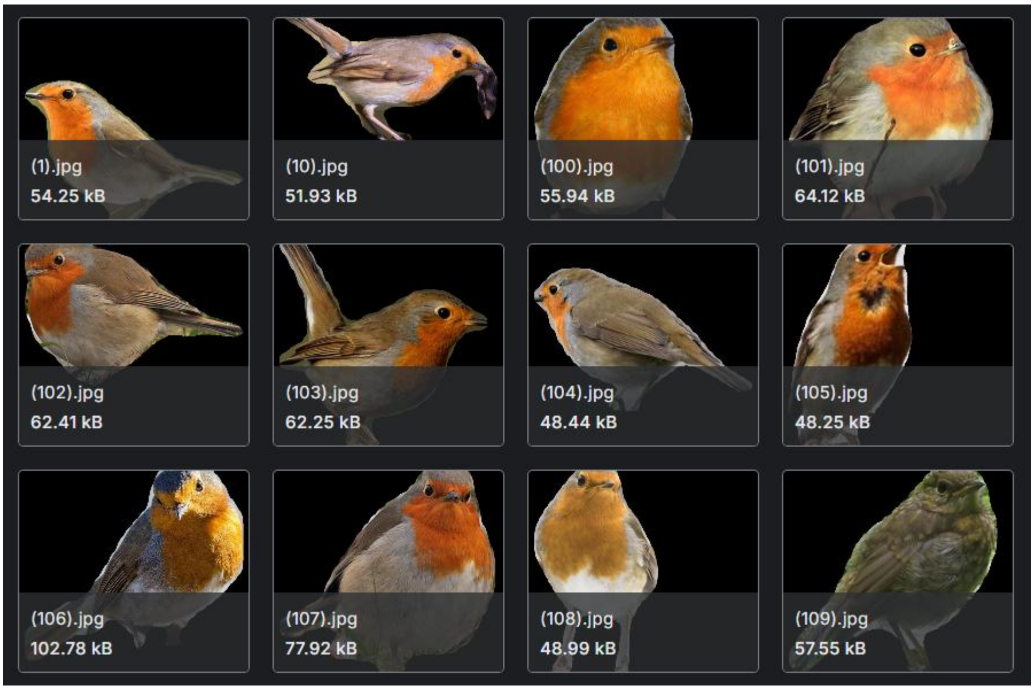

Figure 3. 5: Robin Dataset (without Background)

3.3.1.2 Data Preprocessing

After data collection, the images from the dataset were extracted and converted them into a format suitable for training and testing the model. Data augmentation is applied, which involves zooming, shifting, flipping, and resizing. This can enrich the dataset and enhance its diversity (Neeli et al., 2023). Once all the images were resized to a uniform size of 224 x 224 pixels, the next step is to split the entire dataset using K-fold cross-validation, whereby in this study, 5-fold cross-validation was used where the entire dataset is divided into five sections that rotate between training and validation to ensure a robust performance evaluation.

3.3.1.3 Load Pre-trained EfficientNetB0

In this project, the EfficientNetB0 model was used as the base architecture for bird species classification. The model was initialized with pre-trained weights from the ImageNet dataset, allowing it to utilize the benefit from the rich visual features learnt from large-scale image data. This transfer learning approach is especially useful when working with relatively limited or domain-specific datasets, as it enhances convergence and minimizes training time by reusing underlying patterns (Chollet, 2023).

3.3.1.4 Add Custom Layers

Since the top of the base was not included in the training, custom layers were added on top of the frozen EfficientNetB0 base model. To preserve previously learned ImageNet features, the original pre-trained layers were made non-trainable. A Global Average Pooling layer was used first to lower the output feature maps, followed by a Dropout layer to reduce overfitting. Finally, a Dense layer with softmax activation was utilized to classify the input image to one of the 20 bird species. This method enables the model to function as a feature extractor while adapting to the bird dataset through the newly added layers, ensuring efficiency and good baseline performance (Chollet, 2023).

3.3.1.5 Model Training

In this model training process, 5-fold cross-validation was used to ensure the robustness and reduce bias. The dataset was partitioned into five subsets at random, with each fold alternately training and validating. During each fold, the EfficientNetB0 with base frozen was trained with an Adam optimizer, with categorical cross-entropy as the loss function and accuracy as the performance metric. The training was carried out over a maximum of 20 epochs, with two callback techniques utilized, which are the “EarlyStopping” to stop the training from continuing if the validation performance stop improving and “ReduceLROnPlateau” to decrease the learning rate when performance plateaued.

Both dataset variants (with background and without background) were trained under three configurations:

1. Configuration 1: Dropout 0.3, Batch Size 16
2. Configuration 2: Dropout 0.5, Batch Size 32
3. Configuration 3: Dropout 0.7, Batch Size 64

Data augmentation and preprocessing were used during training with ImageDataGenerator to improve generalization and reduce overfitting. Following each fold, the model's validation loss and accuracy were recorded, and predictions were gathered for further evaluation. This cross-validation layout ensures that model performance is assessed across diverse subsets of the dataset, which improves the generalizability of findings.

3.3.1.6 Model Performance Evaluation

Throughout this project, the evaluation of model performance used metrics such as accuracy, precision, recall, and F1 score. Accuracy was used to evaluate how well the models predicted the target variable overall. Recall measured the percentage of correctly anticipated positive outcomes to all occurrences of positivity in the dataset, whereas precision measured the ratio of correctly predicted positive outcomes to all positive predictions. The F1-score, which combines precision and recall, was used to give a fair assessment of the model's performance, especially when classes were unbalanced (Vo, Nguyen Thien, & Mui, 2023). We were able to make well-informed decisions on the model's efficacy by using a variety of evaluation measures, which gave us a thorough grasp of its performance. Below are the formulas for each of the metrics (Vo, Nguyen Thien, & Mui, 2023).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F_1\ Score = \frac{Precision * Recall}{Precision + Recall}$$

Whereby,

TP: True Positive

TN: True Negative

FP: False Positive

FN: False Negative

Throughout training, performance was measured per fold, and validation accuracy and loss were recorded after each iteration. After completing all five folds, the results were combined to calculate average accuracy and loss across the folds, yielding a reliable assessment of model generalizability.

Besides that, learning curves (training vs. validation accuracy and loss over epochs) were also displayed to help visualize convergence. A final classification report and confusion matrix were created to assess class-level performance and common misclassifications of the bird species.

This multi-metric evaluation approach guaranteed that both overall accuracy and class-specific performance were measured, allowing for meaningful comparisons between dataset variants and training settings.

3.3.1.7 Selection of Best-performing Model for Local Deployment

After evaluating all the model configurations across both dataset variants, the best-performing model for the “with background” dataset was selected for local deployment. The selection was mainly based on the average validation accuracy across the folds. Before integrated into Streamlit, the model was trained on the entire dataset again to ensure its performance optimization. The chosen model was then saved and integrated into a lightweight Streamlit web application, which enables the user to upload the bird image and receive the predicted species label. This deployment provides as a practical demonstration of the model's applicability in real-world circumstances.

The background-free model was not chosen for deployment because it does not accurately depict real-world images, which commonly includes natural backgrounds. While the 'without background' dataset provides a clearer evaluation environment, the 'with background' dataset more closely reflects real-world circumstances such as images taken by users or citizen scientists in the field. Thus, using the best-performing model from the 'with background' setting provides realism and broader application.

3.3.2 Parameters

The hyperparameters used to train deep learning models have a major impact on their performance. According to Ilemobayo et al. (2024), characteristics like learning rate, batch size, and network depth all have a direct impact on convergence speed and accuracy.

In this research, three model configurations were examined with different dropout rates and batch sizes to see how they affected classification results. Furthermore, data augmentation techniques such as flipping and zooming were used to increase model resilience.

Awasthi and Verma (2023) underlined that optimal model performance required a combination of parameters, not just one. To produce the best results, numerous elements, such as augmentation methods and image resolution, were carefully optimized.

3.3.3 Programming Environment

The project was implemented in Python, using libraries such as TensorFlow and Keras for model development and training. All the training of the two dataset variants across three configurations were carried out using Kaggle Notebook, with the NVIDIA Tesla P100 GPU, while local deployment and testing were supported using VS Code and Streamlit. ImageDataGenerator was used for image preprocessing and data augmentation, while scikit-learn and Matplotlib were used for integrating the training pipeline and evaluation metrics.

3.4 Requirement Analysis and Design

This section provides an insight on identifying the system requirements, user roles, and technical environment required to support both the research and deployment phases of this project. Given the project's research focus on evaluating how dataset diversity (in terms of background presence) impacts the performance of a transfer learning model, the system needs revolve around model experimentation and lightweight image classification deployment.

3.4.1 Identify Users and Characteristic

<u>Casual Users</u>

General public users who may have a casual interest in birds or nature.

3.4.2 Functional Requirement

The following functions were designed to support both experimentation and basic deployment:

1. Dataset Input: Load two dataset variations: "with background" and "without background".
2. Model Configuration: Allows training with various dropout and batch size settings.
3. Model Evaluation: Output accuracy, precision, recall, F1-score, learning curves, and confusion matrix.
4. Photo Upload Feature (Streamlit Deployment): Users have a button to upload images of birds from their device and return the predicted species. The system supports common image formats (e.g., JPEG, PNG).

3.4.3 Non-Functional Requirement

The non-functional requirements serve as benchmarks for the system's behaviour and its interactions with the environment in which it was developed. The non-functional requirements for Bird Identification Systems had been identified and listed below:

1. Performance: The bird identification system should load the user interface screen and return the queries result within 30 seconds.
2. Usability: The site should be easy to use for all age groups and skill levels, minimizing the learning curve for casual users.
3. Simplicity: The system should be simple; deployment is intended for demonstration rather than commercial scalability.

3.4.4 Software Requirement

The software requirements emphasize the operating system, programming language, libraries and IDE or Notebook. The software requirements necessary for the efficient operation of the training and local deployment have been identified and are stated below in Table 3.1:

Table 3. 1: Specification of Bird Identification System Software Requirement

System/Software	**Description**
Operating System	Windows 10 or Kaggle Cloud Environment
Programming Language	Python
Libraries	TensorFlow, Keras, NumPy, Pandas, OpenCV, Streamlit
IDE / Notebook	Visual Studio Code (local) or Kaggle Notebook (training)

3.4.5 Hardware Requirement

The hardware requirement focuses on the processor, RAM, storage, and GPU have been identified and are outlined below in Table 3.2:

Table 3. 2: Specification of Bird Identification System Hardware Requirement

Hardware	**Description**
Processor	Intel i5 or AMD Ryzen 5 (or better)
RAM	Minumum 16GB
Storage	SSD with at least 512 GB (for faster data access and storage of models and datasets)
GPU	GPU acceleration on Kaggle (if needed)

3.5 Design

Use case diagram and graphical user interface design will be presented in this section.

3.5.1 Use Case Diagram

3.5.1.1 Overview of Bird Identification System Use Case

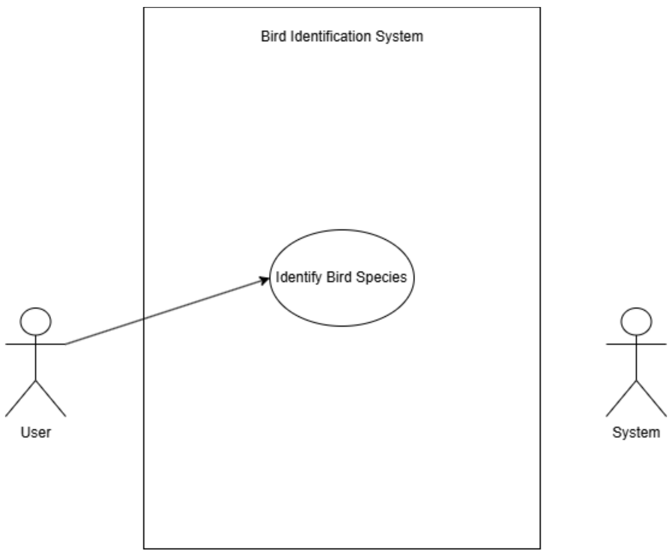

Figure 3. 6: Use Case Diagram

Figure 3.6 above shows the use case diagram for the bird identification system local deployment. In Unified Modeling Language (UML), a use case diagram is a graphic depiction of how users (actors) interact with a system. It demonstrates how diverse users interact with distinct use cases, or functionalities, within a system, capturing its functional needs (GeeksforGeeks, 2025). The Figure 3.6 depicts how users interact with the system to fulfil a single, fundamental function: uploading an image for species identification.

3.5.1.2 Use Case Scenario 1: Identify Bird Species

The deployed model handles only one functional situation, as described in Table 3.3 below:

Table 3. 3: Use Case Description Table

Use Case ID	UC 1
Use Case Name	Identify Bird Species
Actors	User
Description	User can identify bird species by uploading an image using the feature located on the Home Page.
Preconditions	User accesses the image upload interface.
Postconditions	The system displays the identified bird species label.
Main Flow	1. User uploads an image. 2. The system processes the image. 3. Model predicts the bird species. 4. The system displays the identification result.
Alternate Flow	If the image cannot be identified: 1. System displays an error message indicating that the bird species could not be identified.

3.5.2 Minimal GUI for Local Deployment

The graphical user interface (GUI) of the bird identification system was initially prototyped with Figma to visualize the layout and structure of the user interface. This prototype helped outline the interface flow prior to development. The actual implementation was carried out using Streamlit, a lightweight Python-based web framework for deploying machine learning models.

Given the primary research focus of this project, the GUI for local deployment was designed with minimal features. The interface was designed solely for demonstration purposes and features a single use case: uploading a bird image and receiving the predicted species label with confidence score and the information of the predicted bird species. Detailed explanation will be explained in each of the sections below.

3.5.2.1 Main Page

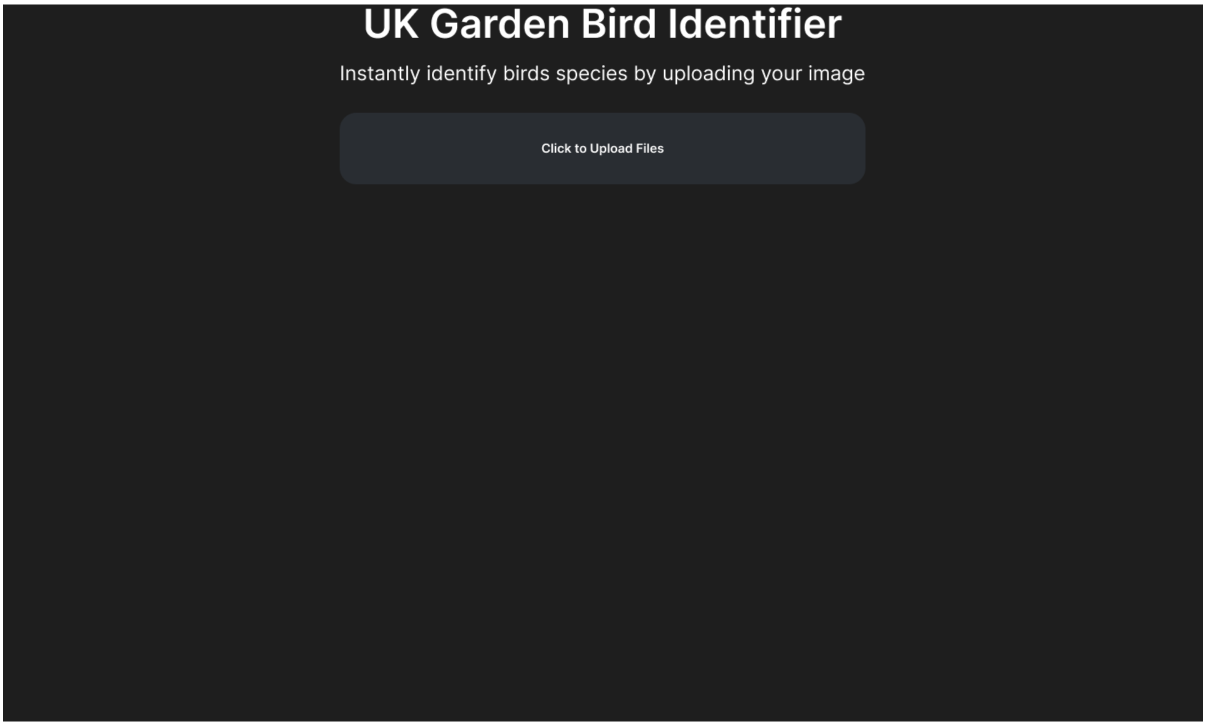

Figure 3. 7: GUI for Main Page

Figure 3.7 shows the main page of the Bird Identification System local deployment features a clean and minimalistic graphical user interface (GUI) designed for ease of use. At the top, the central area prominently displays the title "UK Garden Bird Identifier" along with a subtitle that explains the site's purpose: to instantly identify bird species through image uploads. A large, inviting upload button labelled "Click to Upload Files" serves as the primary call to action, encouraging users to easily submit their bird images for identification. Overall, the design effectively guides users through the identification process with clarity and simplicity.

3.5.2.2 Main Page After Selecting Files

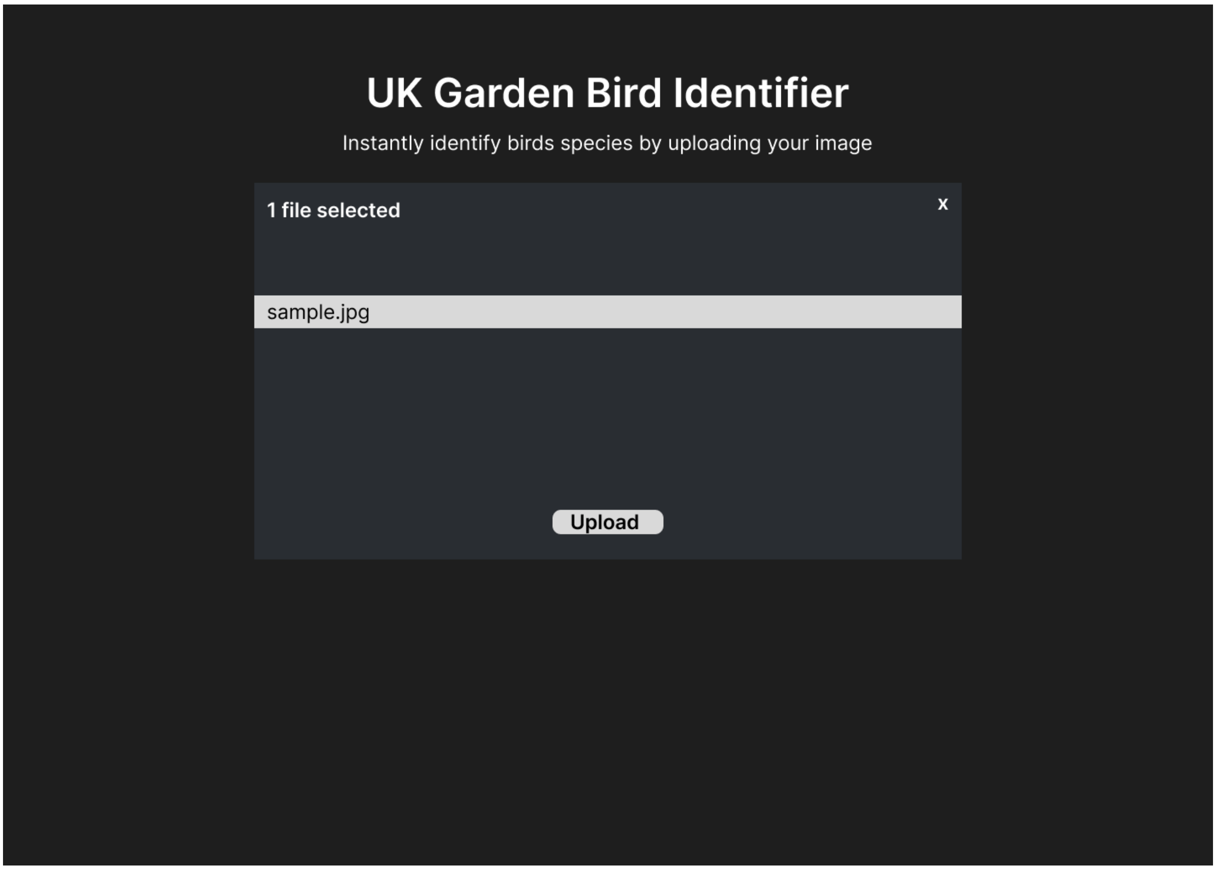

Figure 3. 8: GUI for Main Page after Selecting the Uploaded File

The updated main page GUI as depicts in Figure 3.8 above shows that a file has been successfully selected for identification. A delete icon allows users to remove the file if necessary. Below, a prominent "Upload" button invites users to start the identification process after done uploading.

3.5.2.3 Main Page Showing the Results

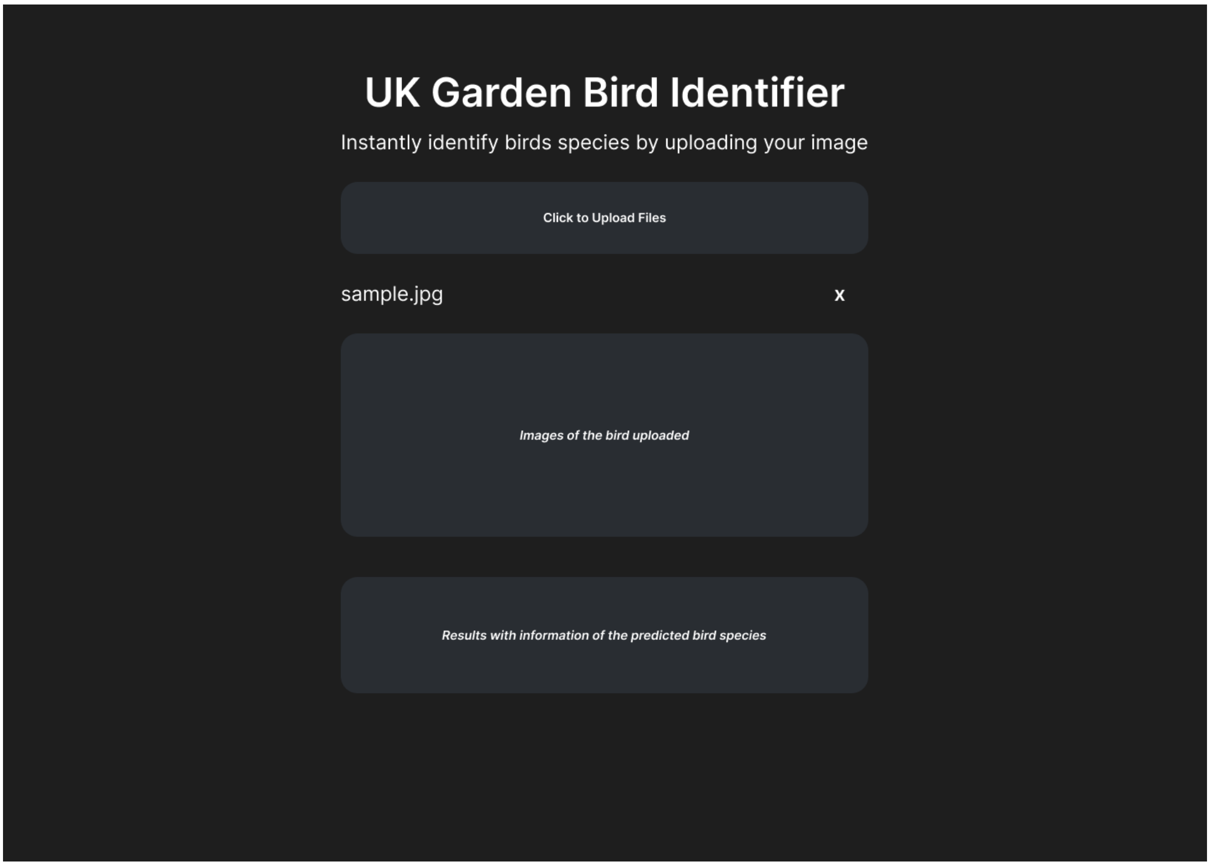

Figure 3. 9: GUI for Results Page

Figure 3.9 shows the main page GUI after the identification results is displayed. At the centre, the title "UK Garden Bird Identifier" is prominently displayed, accompanied by the subtitle explaining the site's purpose. Below this, two key sections are presented: the "Image of the Bird Uploaded," where the submitted image is shown, and the "Results with Information," which outlines the identification results with information stated there.

3.6 Summary

This chapter described the fundamental requirements and design concerns for the bird identification project. It began by implementing an Agile-inspired workflow to manage the iterative training and evaluation processes. The chapter next went over the model architecture based on EfficientNetB0, including data collection, preprocessing, training setup, evaluation metrics, and the process for selecting the best-performing model for local deployment. Key software and hardware requirements were also defined, ensuring compatibility with both experimental processes and lightweight deployment. The functional and non-functional needs were clearly established to facilitate for both research testing and a user-friendly prediction interface. Finally, the system's use case and GUI were presented, exhibiting how end users identify bird images using the local deployed Streamlit application. This foundation helps with the development, testing, and real-world demonstration of the proposed solution.

CHAPTER 4 IMPLEMENTATION AND RESULTS

4.1 Introduction

The findings of the trials carried out for the task of identifying bird species are presented and examined in this chapter. It outlines the implementation strategy employed to evaluate the impact of dataset diversity on the performance of a transfer learning model, EfficientNetB0 for bird species identification. The core objective is to investigate how background information in image data affects classification accuracy and generalization. In order to evaluate the impact of background information on model performance, two dataset settings: one with background and one without background are sourced from the same dataset provider.

For each dataset variant, three model configurations with different dropout rates and batch sizes were trained for each dataset setting:

(1) dropout 0.3 with batch size 16

(2) dropout 0.5 with batch size 32

(3) dropout 0.7 with batch size 64

Accuracy, precision, recall, and F1-score which are the standard classification metrics, were used to test each configuration's performance over 20 epochs using 5-fold cross-validation. To further understand the behaviour of the model, confusion matrices and classification reports were produced.

A comparative analysis follows the results' organization by dataset configuration and setting. Based on this evaluation, the best-performing model is chosen and later integrated into a simple web application prototype for local deployment.

4.2 Hardware and Software

The experiments and implementation in this project were conducted using both cloud-based and local environments to support different stages of the workflow.

4.2.1 Model Training Environment

Kaggle Notebooks was used for the model training and evaluation in this project. Kaggle Notebooks are a sophisticated cloud-based platform for training machine learning models. GPU acceleration, specifically the NVIDIA Tesla P100 GPU was used throughout all the model training and evaluations. This cut down on training time and improved performance for deep learning tasks.

4.2.2 Local Deployment Environment

For local deployment purposes, a simple web application prototype was developed using Streamlit, an open-source Python framework widely recognized for its capabilities in the creation of interactive web applications. This web application enables users to predict and identify the bird species using the trained model by just uploading a bird image to it. The local deployment and testing of the Streamlit app were done on a local machine using Visual Studio Code (VS Code), a flexible and popular code editor that makes it easy to write Python code and work with Jupyter notebooks. Python with a version of 3.10.0 and Streamlit with a version of 1.40.1 were used throughout the implementation and testing.

4.2.3 Programming Language and Libraries

The entire project was developed in Python programming language since it has a large number of machine learning and data processing packages. During the project, the following frameworks and libraries were used:

1. TensorFlow / Keras: TensorFlow with a version 2.18.0 and Keras with a version of 3.5.0 were used for implementing and training the EfficientNetB0 throughout the project.
2. Numpy and Pandas: For manipulating data and doing numerical calculations.
3. Matplotlib and Seaborn: For visualizing performance metrics such as confusion matrices and training curves.
4. Scikit-learn: For evaluation metrics such as classification reports and confusion matrices and also for K-fold cross-validation.
5. OpenCV (cv2) and os / pathlib: For managing path management, dataset loading, and image preparation.

In addition to providing flexibility in experimenting with various model configurations and dataset sets, this hardware-software combination guaranteed the EfficientNetB0 models' training was effective and repeatable.

4.3 Dataset Collection and Preprocessing

This project will utilize two dataset settings, which from the "20 UK Garden Birds" image dataset, sourced from Kaggle. The goal was to evaluate the impact of background information on bird species identification the EfficientNetB0 model performance. Therefore, two distinct dataset settings were employed:

1. With Background: Normal and original bird images which remains its' natural background elements such as branches, sky, tree and so on.
2. Without Background: The same type of images as the one with background, but with

all the background removed, just focuses solely on the bird itself.

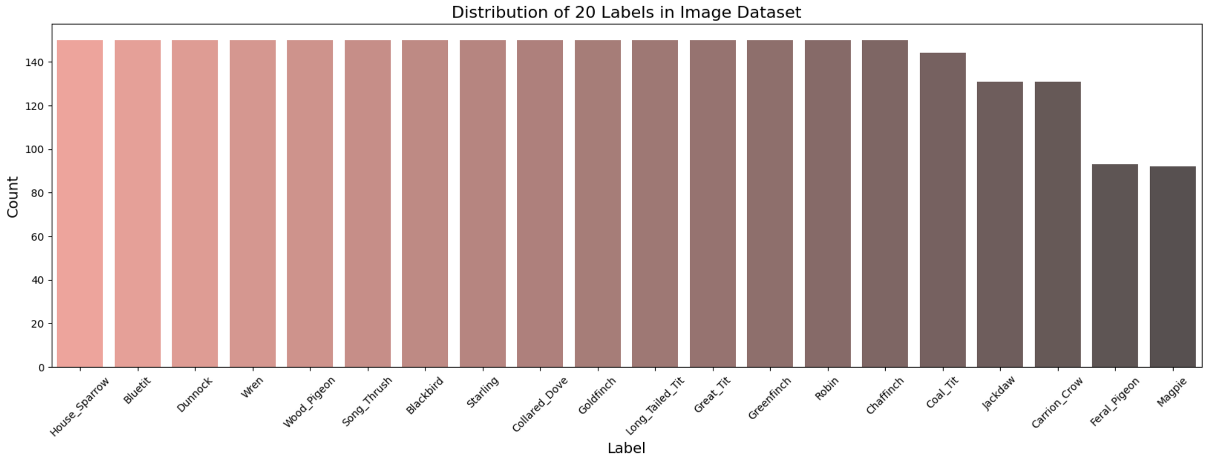

Figure 4. 1:Distribution of Image Samples Across 20 Species

Each of the dataset has images of 20 bird species, with a total of 2841 images per dataset. As you can observe from Figure 4.1, which is the distribution of image samples across 20 bird species for both the with background and without background dataset. The image distribution per class in overall is balanced at 150 images per species, except 5 bird species with slightly fewer samples, for example the Jackdaw with a number of 131 images per dataset and Magpie with a number of 92 images per dataset.

4.3.1 Dataset Organization

The data was organized into two primary columns in a Pandas DataFrame:

- Filepath: The path to the image file
- Label: Name of the corresponding bird species

The entire dataset was shuffled before training.

4.3.2 Image Preprocessing

The "ImageDataGenerator" class from Keras was used to resize all the photos to a consistent input shape, 224 to be compatible with EfficientNetB0. Data Augmentation was also used on training the data, including:

- Horizontal flipping
- Rotation range
- Zoom, width, and height shifts

4.4 Model Architecture

This project utilizes EfficientNetB0, a transfer learning model known for its effectiveness and efficacy on image classification tasks. For transfer learning on datasets with constrained size and computational capacity, EfficientNetB0 is ideal (Tan & Le, 2019).

EfficientNetB0 was loaded with pretrained weights from ImageNet but excluding the top classification layers. A custom classification head was added to the base model. It has a global average pooling layer, a dropout layer for regularization, and a dense output layer with softmax activation to predict the 20 bird species classes.

Unlike what was usually being practiced during transfer learning, instead of freezing the base model layers, this project opted to maintain all layers of the EfficientNetB0 trainable. This approach made it possible to fine-tune the pretrained backbone, using the dataset for bird identification. When background variation is a crucial factor being assessed, this method allows the model to better adapt to domain-specific features found in the training data, even though it increases the computational effort.

Across model configurations, dropout was performed at varying rates (0.3, 0.5, and 0.7) to promote generalization and lessen overfitting. In addition to early stopping and learning rate reduction callbacks, the Adam optimizer with a learning rate of $1 * 10^{-4}$ was employed. But since "ReduceLROnPlateau" is being applied, the learning rate will automatically decrease during training to aid the model keep learning when the improvements slow down, possibly enhancing model performance and convergence. The model was trained over 20 epochs with 5-fold cross-validation to guarantee the model's robustness across various dataset subsets

4.5 Training and Validation

This section will be discussed on the training and the validation of the model across each of the model configurations for both dataset settings (with and without background).

4.5.1 Training

A consistent and structured training approach was used to fully study how different datasets, especially the presence or absence of background information, affect the performance of a transfer learning model. To guarantee fair and reliable comparisons, this approach was used in the same way for both dataset settings (with and without background). And for each of the dataset settings, three different configurations were carried out per dataset.

4.5.1.1 Model Configuration

As mentioned in the above section, there were three different model configurations being evaluated based on different dropout rates and batch sizes:

- Configuration 1: Dropout rate 0.3, Batch size 16

- Configuration 2: Dropout rate 0.5, Batch size 32
- Configuration 3: Dropout rate 0.7, Batch size 64

Six different experiments, namely BG1, BG2, BG3, NBG1, NBG2, and NBG3 were conducted by individually testing each configuration on each dataset. Table 4.1 below are the training setup with two dataset settings with three configurations per dataset (different dropout and batch sizes).

Table 4. 1: Training Setup/ Configurations

Name	Dataset Version	Dropout	Batch Size
BG1	With Background	0.3	16
BG2	With Background	0.5	32
BG3	With Background	0.7	64
NBG1	Without Background	0.3	16
NBG2	Without Background	0.5	32
NBG3	Without Background	0.7	64

4.5.2 Cross-validation

As mentioned in the previous section above, 5-fold cross-validation were used in this project to ensure the robustness and unbiased evaluation of the EfficientNetB0 model's performance in both dataset scenarios (with and without background). Each of the entire dataset settings were split into 5 equal folds, and the datasets were randomly shuffled before splitting. 80% of the entire dataset will be used as training set, while 20% will be used as the validation set.

For my case, for each of the dataset settings (with background and without background) will have a total of 2841 images. The splitting will be slightly uneven as no fraction of an image was allowed. Therefore, after splitting the entire dataset into 5 equal folds, 4 folds will have a total of 568 images, and the remaining 1-fold will have a total of 569 images due to the remainder, as shown at Table 4.2 below:

Table 4. 2: Splitting of Dataset Across 5-Folds

Fold / Part	Training Set Size	Validation Set Size
1	2841-569 = 2272	569
2	2841-568 = 2273	568
3	2841-568 = 2273	568
4	2841-568 = 2273	568
5	2841-568 = 2273	568

Every fold took turns serving as the validation set, meaning each image appeared four times in the training set and exactly once in the validation set. For example, for the first iteration (Fold 1), training indices will be having 2272 images (all except the 569 images) while validation indices will be having 569 images (which is the 569 images from Fold 1). The same process was repeated, but a new set of 568 images was used for validation each time and each image has been validated exactly once after all folds, as shown at Table 4.3 below:

Table 4. 3: Splitting Training and Validation Set

Fold	Training Set		Validation Set	
1	Parts 2,3,4,5	2272 images	Part 1	569 images

2	Parts 1,3,4,5	2273 images	Part 2	568 images
3	Parts 1,2,4,5	2273 images	Part 3	568 images
4	Parts 1,2,3,5	2273 images	Part 4	568 images
5	Parts 1,2,3,4	2273 images	Part 5	568 images

4.6 Performance Evaluation

As an example, this section evaluates the performance of BG1 under Configuration 1 (dropout 0.3, batch size 16) as the same procedure will be applied to rest of the configurations, with only the parameter values varying. This section will be showing how the model's performance was accessed using key metrics such as accuracy, loss, precision, recall, F1-score, and confusion matrix. A comprehensive comparison of all the model configurations will be further discussed in Chapter 5.

4.6.1 Evaluate Model

For the model evaluation part, the EfficientNetB0 model used the evaluate() function on the validation datasets for each fold. It will then return the key metrics for loss and accuracy. The metrics were recorded using the "val_loss" and "val_acc" values. The evaluation was performed using 5-fold cross-validation, where training and validation data were partitioned differently for each fold.

4.6.2 Predict Model

The predict() method which generates a probability distribution across the 20 bird species classes for every validation image, was used to make predictions for each fold. The final predicted label was obtained by applying the argmax() function to each prediction in order to identify the class with the highest probability. The model's performance was then evaluated by comparing these predictions (y_pred) with the validation generator's ground truth labels (y_true). The all_preds and all_trues lists were used to compile all predictions and their matching true labels for each of the five folds. The creation of final evaluation metrics, such as the confusion matrix and classification report, which offer a more thorough picture of model performance throughout the whole dataset, depended heavily on this accumulation.

4.6.3 Learning Curve

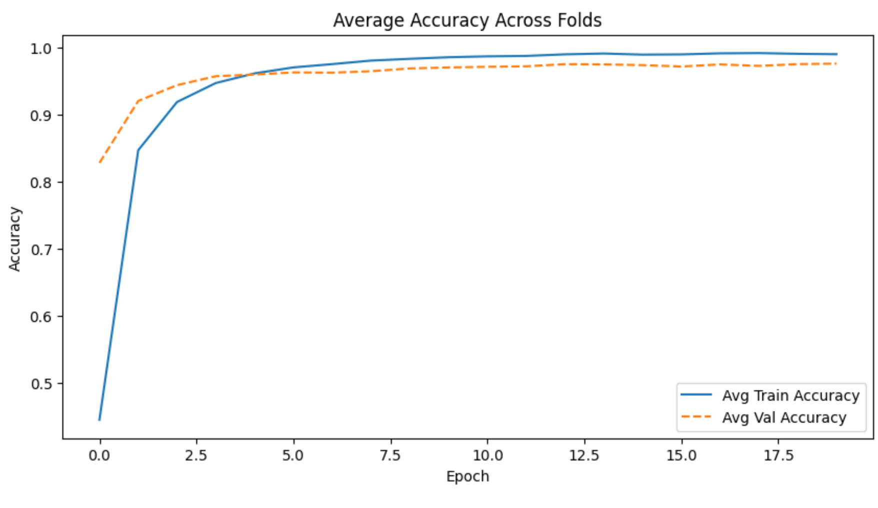

Figure 4. 2: Learning Curve on Average Accuracy Across Folds

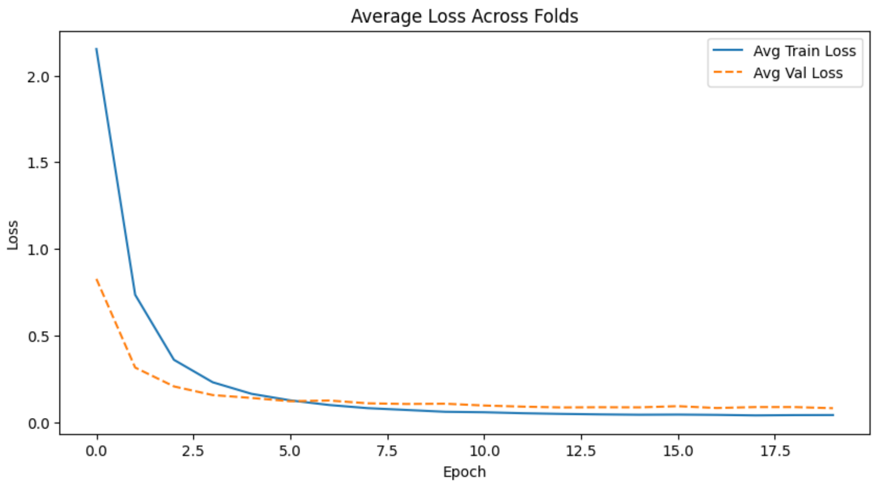

Figure 4. 3: Learning Curve on Average Loss Across Folds

Figure 4.2 and Figure 4.3 were the learning curve of the EfficientNetB0 model for BG1 under Configuration 1 (dropout 0.3, batch size 16). The accuracy and loss metrics were recorded for both the training and validation sets throughout all five folds in order to gain a better understanding of the model's learning behaviour during training. A smooth and realistic learning curve for the overall model performance was then created by averaging these metrics epoch by epoch. While the loss curve shows how the model's prediction errors dropped with time, the accuracy curve demonstrates how effectively the model learned about classifying bird species accurately on the training and validation sets. These plots aid in evaluating possible signs of underfitting (performs poorly on both training and validation data) or overfitting (training accuracy rising while validation accuracy declines). In general, the learning curves offer a thorough visualization of the training dynamics of the model, assisting in confirming the stability and efficacy of the selected model configuration.

4.6.4 Classification Report

Final Combined Classification Report:

	precision	recall	f1-score	support
Blackbird	0.94	0.97	0.95	150
Bluetit	0.97	0.98	0.98	150
Carrion_Crow	0.90	0.98	0.94	131
Chaffinch	0.99	0.99	0.99	150
Coal_Tit	0.99	0.99	0.99	144
Collared_Dove	0.97	0.97	0.97	150
Dunnock	0.97	0.97	0.97	150
Feral_Pigeon	1.00	0.96	0.98	93
Goldfinch	0.99	1.00	1.00	150
Great_Tit	0.99	0.96	0.97	150
Greenfinch	0.99	0.98	0.98	150
House_Sparrow	0.98	0.97	0.97	150
Jackdaw	0.97	0.93	0.95	131
Long_Tailed_Tit	0.99	0.99	0.99	150
Magpie	1.00	0.99	0.99	92
Robin	0.99	0.99	0.99	150
Song_Thrush	0.99	0.97	0.98	150
Starling	1.00	0.95	0.98	150
Wood_Pigeon	0.95	0.99	0.97	150
Wren	0.97	0.99	0.98	150
accuracy			0.98	2841
macro avg	0.98	0.98	0.98	2841
weighted avg	0.98	0.98	0.98	2841

Figure 4. 4: Classification Report

The Scikit-learn library's `classification_report()` method can be used to create the model's classification report, as illustrated in Figure 4.4. This report includes metrics for each class, including precision, recall, F1-score, and also the macro and weighted average of the model.

4.6.5 Confusion Matrix

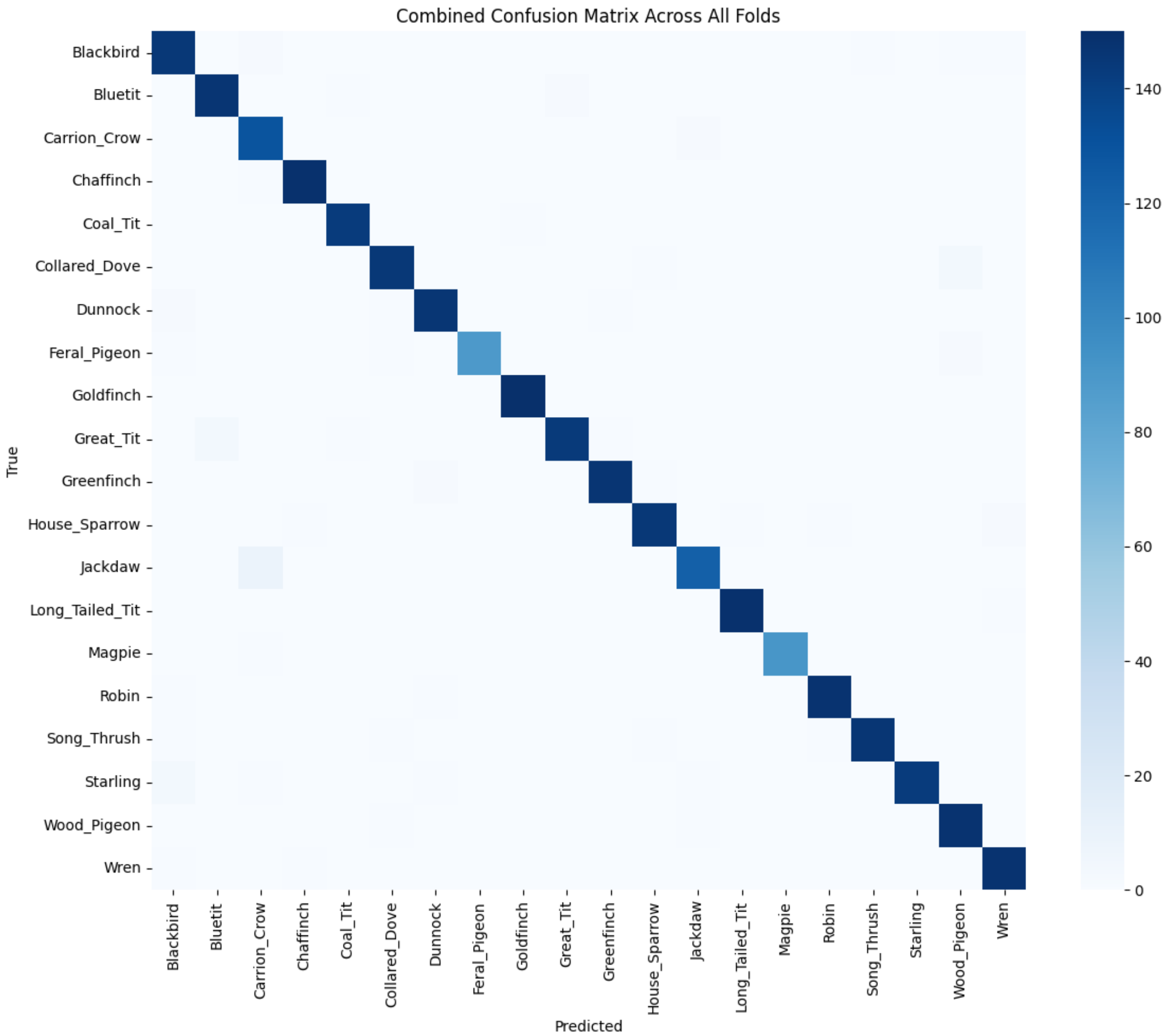

Figure 4. 5: Confusion Matrix

By using the confusion_matrix() function and taking the true labels and predicted labels, a matrix showing the correct and incorrect predictions were constructed. The plotting of the results makes it easier to see how accurately the model can predict each class and aids in identifying which classes are often being confused.

4.7 Summary of Model Performance

To provide an overview of all experimental setups, Table 4.4 below presents a performance summary for the six trained models. Each model was evaluated using 5-fold cross-validation, and average metrics such as accuracy, precision, recall, and F1-score were measured.

The findings clearly reveal that models trained on the dataset with background consistently outperformed those trained on the background-removed variation. BG1 (dropout 0.3, batch size 16) performed the best across all measures, supporting its selection as the final model for deployment. This summary allows for a direct performance comparison and sets the stage for deeper interpretation and discussion in the next chapter.

Table 4. 4: Summary of Performance Across All Configurations

Configuration	Dataset	Dropout	Batch Size	Avg Accuracy	Avg Loss	Precision	Recall	F1-Score
BG1	With Background	0.3	16	0.9771	0.0778	0.98	0.98	0.98
BG2	With Background	0.5	32	0.9704	0.0916	0.97	0.97	0.97
BG3	With Background	0.7	64	0.9616	0.1209	0.97	0.96	0.96
NBG1	Without Background	0.3	16	0.9666	0.0997	0.97	0.97	0.97
NBG2	Without Background	0.5	32	0.9616	0.1369	0.96	0.96	0.96
NBG3	Without Background	0.7	64	0.9588	0.1391	0.96	0.96	0.96

4.8 Local Deployment

Figure 4.6 below shows the interface of the bird identification local deployment interface using Streamlit.

Figure 4. 6: Interface of Local Deployment (Streamlit)

The best-performing model configuration among the with background dataset was chosen and retrained on the entire dataset for local deployment in order to facilitate real-world use. This process guarantees that the final model makes use of all the data that is available, which could improve its overall accuracy and resilience when used in practice.

As mentioned earlier in the above section, the "with background" dataset setting, notably BG1, which used a dropout of 0.3 and a batch size of 16, produced the model that was chosen for deployment. This model was chosen as it performed the best in terms of average accuracy during the cross-validation studies. The "without background" dataset was not considered for deployment since it does not reflect real-world bird imagery where backgrounds are always naturally present.

Using the same training setup as previous folds, the last training step used the entire dataset as input. To maximize performance and avoid overfitting, the model was assembled and trained using early stopping and learning rate reduction callbacks. The data generator was configured to load and augment all labeled bird images.

A simple web application prototype was then developed using Streamlit for local deployment and was carried out on a local machine using VS code.

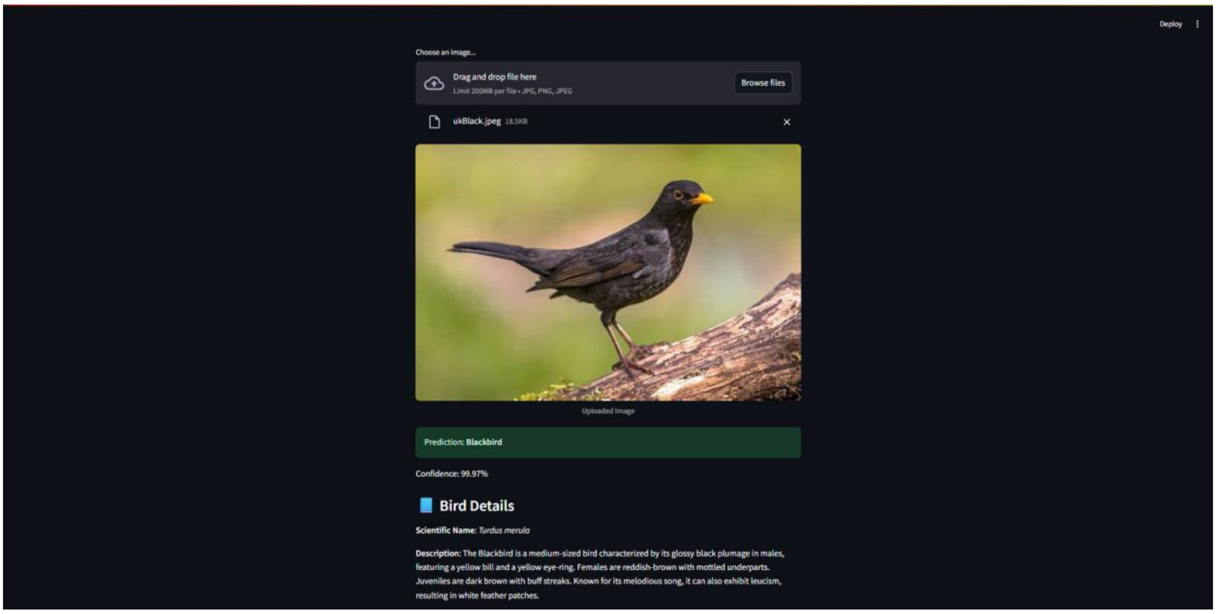

Figure 4. 7: Sample Output after Uploading Bird Image

Figure 4.7 above depicts the sample output of the system after successfully uploading the bird image for identification. As shown, the uploaded bird image will show in the system, with the predicted label displayed too. The confidence level in percentage as well as the bird details were shown to give users a better understanding of the bird species.

It is crucial to highlight that the deployed model was only trained on the 20 UK bird species included in the dataset. If a user uploads an image of a bird that is not in this category (for example, an Indian bird species), the system will nevertheless return a predicted label from one of the known classes, along with confidence level and species information, even if the result is not accurate. Such a behaviour is typical of closed-set classification algorithms, which are not designed to reject or detect unknown inputs.

4.8 Summary

This chapter provides an overview of the implementation process on the EfficientNetB0 model with six configurations in total, combining different dropout rates and batch sizes across two dataset settings which are the dataset with background and the dataset without background. Five-fold cross-validation was used to train each model configuration in order to guarantee reliable evaluation and generalization. Gathering validation accuracy and loss over folds, making predictions, and examining important performance indicators like confusion matrices and classification reports were all part of the model's evaluation process. In order to investigate training dynamics and identify indications of overfitting or underfitting, learning curves were also plotted. The best-performing model from the with background dataset setting was then chosen for local deployment. Overall, this chapter laid the groundwork for experimental comparison and offered comprehensive insights into the behaviour of the model, providing the foundation for the analysis and discussion of the results in the following chapter.

CHAPTER 5 RESULT AND ANALYSIS

5.1 Introduction

In this chapter, the experimental findings from training and testing the bird species identification model with two dataset settings: one with background and one without, are thoroughly evaluated and analyzed. The aim is to investigate the impact of dataset diversity, namely the presence or absence of background features in the images, on the performance of transfer learning model which is EfficientNetB0 in this case.

For each of the dataset setting, the model's performance was evaluated in three different configurations. These setups differ according to batch size and dropout rate, specifically:

1. Configuration 1: Dropout 0.3, Batch Size 16
2. Configuration 2: Dropout 0.5, Batch Size 32
3. Configuration 3: Dropout 0.7, Batch Size 64

Accuracy, loss, precision, recall, F1-score, and confusion matrix are used to measure the results of evaluating each configuration using 5-fold cross-validation. This made it possible to fully comprehend the impact of each dataset and parameter setting on model generalization.

A comparison analysis was then performed across configurations and dataset types, which includes:

1. Trends in performance within the same dataset setting (for example, how does the increasing of dropout or batch size affect the accuracy or performance)
2. Trends in performance within the different dataset setting
3. Comparison among the obtained results with the relevant existing studies

As mentioned in the previous chapter, the model that performed the best will be chosen for local deployment. This chapter provides a clear and thorough explanation of the findings to support that choice made.

5.2 Performance on Dataset with Background

This section presents the performance of the EfficientNetB0 model trained on the with-background dataset across three configurations (BG1, BG2, and BG3) using 5-fold cross-validation. The evaluation metrics include accuracy, precision, recall, and F1-score, along with classification reports and confusion matrices. The findings and analysis for each configuration were shown in the subsequent sections.

5.2.1 Configuration BG1 (Dropout: 0.3, Batch Size: 16)

The first experiment (Configuration BG1) was performed on the dataset with background and trained on EfficientNetB0 model using a dropout rate of 0.3 and a batch size of 16. The performance was evaluated using 5-fold cross-validation over 20 epochs. Table 5.1 summarizes the accuracy and loss across all five folds.

Table 5. 1: Accuracy and Loss Across for BG1

Fold	Accuracy	Loss
1	0.9807	0.0669
2	0.9789	0.0643
3	0.9771	0.0693

4	0.9771	0.0730
5	0.9718	0.1154
Average	**0.9771**	**0.0778**

As from the results above, it can be clearly seen that the model achieved a mean accuracy of 0.9771 (97.71%) with an average loss of 0.0778, indicating a high classification performance. A dropout rate of 0.3 seems to be a good compromise between retaining learning capacity and avoiding overfitting. Due to more frequent weight updates, the smaller batch size of 16 might have improved generalization.

Final Combined Classification Report:

	precision	recall	f1-score	support
Blackbird	0.94	0.97	0.95	150
Bluetit	0.97	0.98	0.98	150
Carrion_Crow	0.90	0.98	0.94	131
Chaffinch	0.99	0.99	0.99	150
Coal_Tit	0.99	0.99	0.99	144
Collared_Dove	0.97	0.97	0.97	150
Dunnock	0.97	0.97	0.97	150
Feral_Pigeon	1.00	0.96	0.98	93
Goldfinch	0.99	1.00	1.00	150
Great_Tit	0.99	0.96	0.97	150
Greenfinch	0.99	0.98	0.98	150
House_Sparrow	0.98	0.97	0.97	150
Jackdaw	0.97	0.93	0.95	131
Long_Tailed_Tit	0.99	0.99	0.99	150
Magpie	1.00	0.99	0.99	92
Robin	0.99	0.99	0.99	150
Song_Thrush	0.99	0.97	0.98	150
Starling	1.00	0.95	0.98	150
Wood_Pigeon	0.95	0.99	0.97	150
Wren	0.97	0.99	0.98	150
accuracy			0.98	2841
macro avg	0.98	0.98	0.98	2841
weighted avg	0.98	0.98	0.98	2841

Figure 5. 1: Classification Report for BG1

The final combined classification report, aggregating predictions across all folds, was shown in Figure 5.1. The model achieved a macro and weighted F1-score of 0.98, with multiple classes such as Goldfinch, Chaffinch, Long_Tailed_Tit, and Magpie reaching perfect or near-perfect scores.

As mentioned earlier in Chapter 4, this model was being chosen as the model for local deployment as it was the best-performing model which gives the best accuracy among the other configurations (BG2 and BG3) for the with background dataset setting, due to its reliable and constant performance on all evaluation metrics.

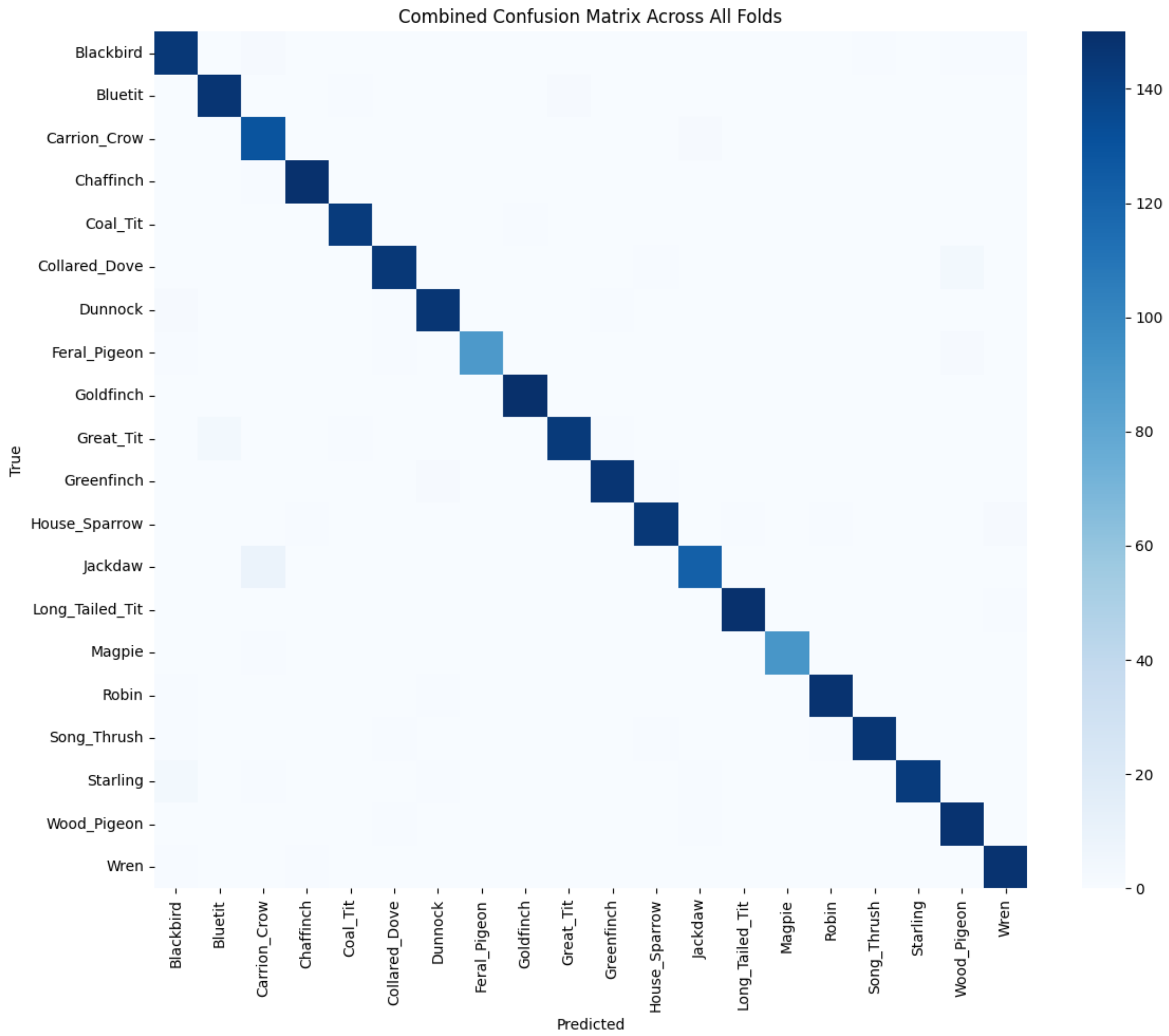

Figure 5. 2: Confusion Matrix for BG1

The confusion matrix in Figure 5.2 displayed the performance of the model across all folds. Whereas off-diagonal elements showed misclassifications, diagonal elements showed instances that were correctly classified. The model's robustness was demonstrated by the confusion matrix, which displayed that most of the bird species were accurately recognized with few cases of misclassification. However, misclassifications were observed between the Jackdaw and Carrion_Crow classes, likely due to both appearing to be in black color.

5.2.2 Configuration BG2 (Dropout: 0.5, Batch Size: 32)

In the second experiment (Configuration BG2), the EfficientNetB0 model was trained on the background dataset with a dropout rate of 0.5 and batch size of 32. The 5-fold cross-validation was used to assess the performance. The accuracy and loss for each of the five folds were summarized in Table 5.2.

Table 5. 2: Accuracy and Loss Across for BG2

Fold	Accuracy	Loss
1	0.9631	0.1195
2	0.9736	0.0783
3	0.9648	0.0948
4	0.9771	0.0798
5	0.9736	0.0856
Average	**0.9704**	**0.0916**

With an average loss of 0.0916, the accuracy is 0.9704 (97.04%) with a little greater variance in performance between folds. This consistency showed that the model worked consistently across different dataset subsets. Due to the increased regularization caused by the higher dropout rate of 0.5, there may have been a slight underfitting effect in comparison to BG1. The moderate batch size of 32 may have reduced the frequency of updates, causing the model to take longer to adapt to the data. Although performance was slightly decreased due to BG2's relatively excessive regularization, it was still consider performing well.

Final Combined Classification Report:

	precision	recall	f1-score	support
Blackbird	0.92	0.95	0.93	150
Bluetit	0.96	0.99	0.98	150
Carrion_Crow	0.87	0.96	0.91	131
Chaffinch	0.97	0.99	0.98	150
Coal_Tit	0.99	0.98	0.98	144
Collared_Dove	0.98	0.97	0.98	150
Dunnock	0.95	0.98	0.97	150
Feral_Pigeon	0.98	0.95	0.96	93
Goldfinch	1.00	1.00	1.00	150
Great_Tit	0.97	0.95	0.96	150
Greenfinch	0.99	0.97	0.98	150
House_Sparrow	0.98	0.95	0.96	150
Jackdaw	0.94	0.89	0.91	131
Long_Tailed_Tit	0.98	1.00	0.99	150
Magpie	1.00	0.99	0.99	92
Robin	0.98	0.97	0.97	150
Song_Thrush	0.98	0.98	0.98	150
Starling	1.00	0.94	0.97	150
Wood_Pigeon	0.96	0.97	0.96	150
Wren	0.99	0.97	0.98	150
accuracy			0.97	2841
macro avg	0.97	0.97	0.97	2841
weighted avg	0.97	0.97	0.97	2841

Figure 5. 3: Classification Report for BG2

From the final combined classification report generated above in Figure 5.3, it can be told that the overall accuracy is 0.97, indicating that 97% of the predictions were correct. The macro average of 0.97 showed that a good overall performance across classes without class size bias while the weighted average of 0.97 reflects performance while considering the distribution of classes. For example, the Goldfinch, Magpie and Starling reach a perfect score of 1.00. The model performs well overall, with high scores across most classes.

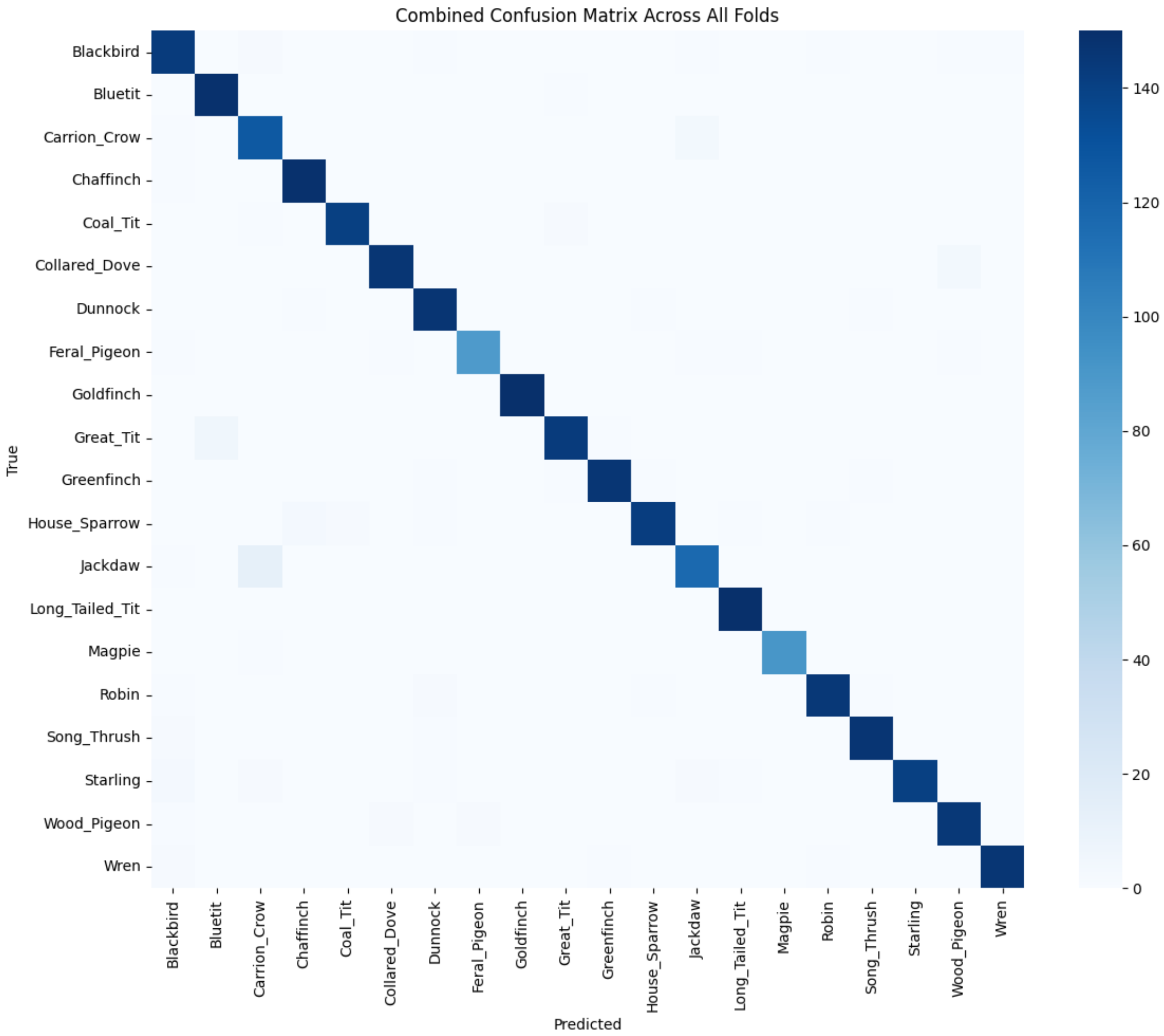

Figure 5. 4: Confusion Matrix for BG2

From Figure 5.4 above, the matrix showed a strong diagonal pattern, indicating that most predictions were correct. The high numbers along the diagonal indicated accurate forecasts for each class. There was a chance of misclassification between House_Sparrow and Chaffinch as there might be some visual similarities, especially the pattern they had on the wings, which can confuse the model.

5.2.3 Configuration BG3 (Dropout: 0.7, Batch Size: 64)

The EfficientNetB0 model was trained on the background dataset in the third experiment (Configuration BG3) with a batch size of 64 and a dropout rate of 0.7. The performance was evaluated using 5-fold cross-validation. Table 5.3 provides a summary of the accuracy and loss for each of the five folds.

Table 5. 3: Accuracy and Loss Across for BG3

Fold	Accuracy	Loss
1	0.9701	0.1055
2	0.9525	0.1324
3	0.9472	0.1664
4	0.9648	0.1149
5	0.9736	0.0854
Average	**0.9616**	**0.1209**

Out of the three, BG3 had the largest average loss (0.1209) and the lowest average accuracy 0.9616 (96.16%). As a dropout rate of 0.7 can be considered as very high, the model most certainly underfitted, failing to identify significant patterns. Besides that, a large batch size of 64 might have also led to less frequent updates which caused the model to learn slower. Due to the large batch size and high dropout rate, which reduced the model's capacity for learning, it made this the least effective configuration among the with background dataset.

Final Combined Classification Report:

	precision	recall	f1-score	support
Blackbird	0.96	0.94	0.95	150
Bluetit	0.97	0.97	0.97	150
Carrion_Crow	0.87	0.94	0.90	131
Chaffinch	0.99	0.97	0.98	150
Coal_Tit	0.97	0.99	0.98	144
Collared_Dove	0.94	0.99	0.96	150
Dunnock	0.94	0.97	0.95	150
Feral_Pigeon	0.99	0.92	0.96	93
Goldfinch	1.00	0.99	1.00	150
Great_Tit	0.99	0.95	0.97	150
Greenfinch	0.98	0.97	0.98	150
House_Sparrow	0.92	0.95	0.93	150
Jackdaw	0.91	0.92	0.92	131
Long_Tailed_Tit	0.99	0.99	0.99	150
Magpie	0.99	0.99	0.99	92
Robin	0.99	0.96	0.98	150
Song_Thrush	0.98	0.97	0.97	150
Starling	1.00	0.95	0.98	150
Wood_Pigeon	0.98	0.96	0.97	150
Wren	0.96	0.97	0.97	150
accuracy			0.96	2841
macro avg	0.97	0.96	0.96	2841
weighted avg	0.97	0.96	0.96	2841

Figure 5. 5: Classification Report for BG3

This setup had a macro F1-score of 0.96 and an overall accuracy of 0.96 (96%) as observed from Figure 5.5. Class Goldfinch had nearly prefect recall and perfect precision. This implies that Goldfinch was predicted by the model with zero false positives and just one false negative. While for the Carrion_Crow, it had the lowest precision among all the other classes, meaning that other species were sometimes incorrectly classified as Carrion_Crow. Less unique traits or visual similarities could be the cause of this. The model's performance was consistent among frequent and less frequent bird species, as evidenced by the weighted average scores (Precision: 0.97, Recall: 0.96, F1-Score: 0.96), which roughly matched the macro averages. This consistency demonstrates even more how well the model managed class imbalance, especially there were a few classes with uneven support such as Feral_Pigeon with 93 samples, Magpie with 92 samples, while most others had 150 samples.

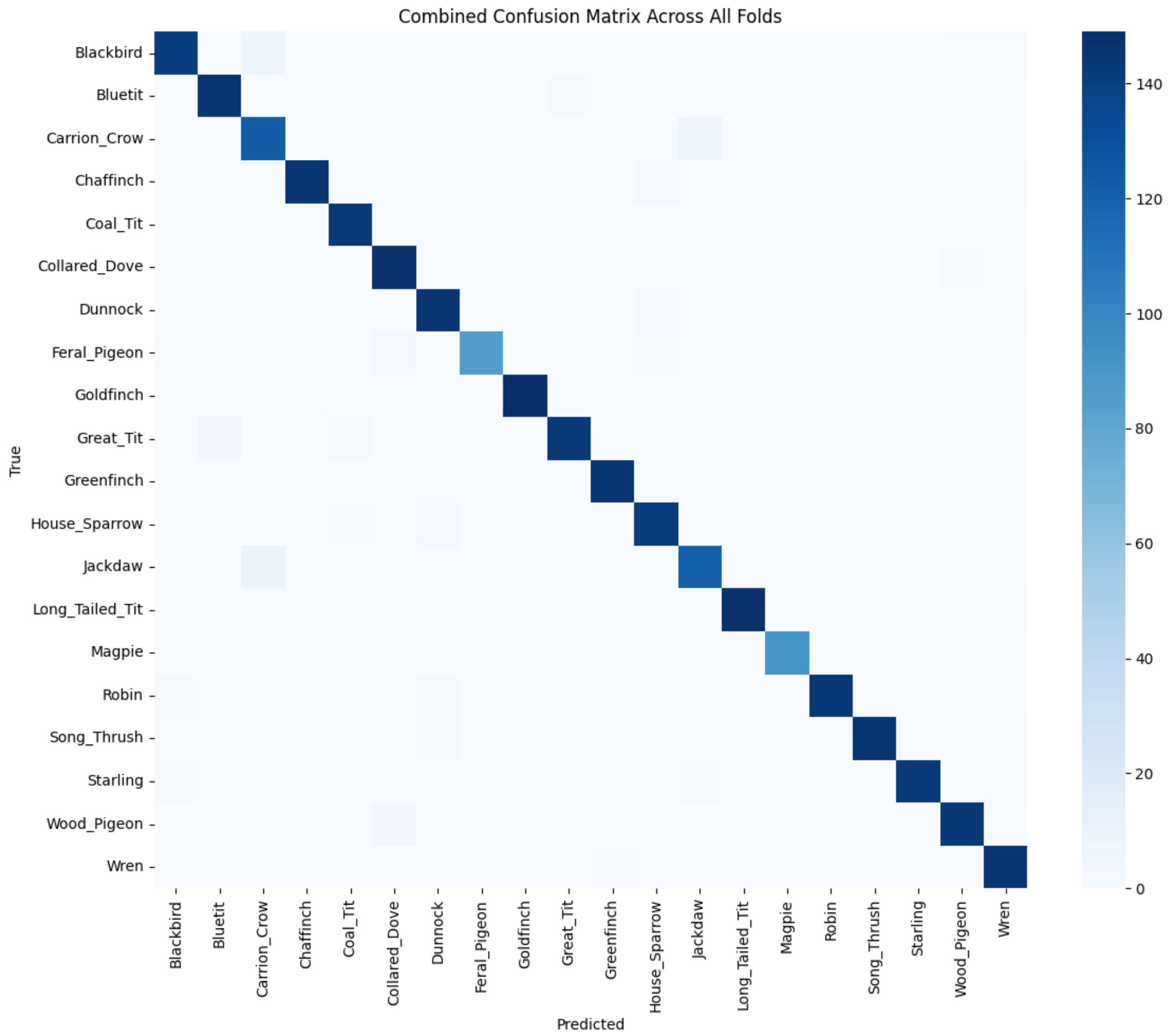

Figure 5. 6: Confusion Matrix for BG3

The 20 species' true and predicted classes were shown graphically in the confusion matrix in Figure 5.6 above. From the confusion matrix above, majority of the values line on the diagonal, indicating that most of the predictions correspond to the true class, while only a minimal off-diagonal values exist such as Carrion_Crow and Jackdaw which aligned with their slightly lower precision and recall.

5.2.4 Comparison Across Configurations with Background (BG1–BG3)

Table 5. 4: Comparison Across Configurations with Background

Configurations	Dropout	Batch Size	Avg Accuracy
BG1	0.3	16	0.9771
BG2	0.5	32	0.9704
BG3	0.7	64	0.9616

From Table 5.4, it can be clearly shown that accuracy clearly declines with an increasing batch size and dropout rate. The best configuration which gave the best performance is BG1, with an average accuracy of 0.9771 (97.71%). This suggests that this combination of lower dropout and smaller batch size is optimal for this dataset as a lower dropout rate allowed the model to learn more features effectively without overfitting while the smaller batch size may aid in providing more frequent updates during training, which leads to improvements in generalization. Performance in BG2 and BG3 decreased slightly when batch size and dropout increased. This might be due to a higher dropout having removed too much learning capacity and the larger batch size made learning less responsive.

BG1 was chosen for final training and deployment since it performed better than other configurations as mentioned earlier. These results implied that enhancing transfer learning performance on bird classification tasks with real-world background data requires careful adjustment of regularization and training dynamics.

5.3 Performance on Dataset Without Background

This section used 5-fold cross-validation to show the performance of the EfficientNetB0 model trained on the without background dataset in three different configurations (NBG1, NBG2, and NBG3). Along with classification reports and confusion matrices, the same evaluation metrics —accuracy, precision, recall, and F1-score were employed. The following sections present the results and analysis for each setting.

5.3.1 Configuration NBG1 (Dropout: 0.3, Batch Size: 16)

As performing the same way as the dataset with background, the first experiment for the without background dataset (NBG1), was trained on the EfficientNetB0 model using a dropout rate of 0.3 and a batch size of 16. The performance was evaluated using 5-fold cross-validation over 20 epochs. Table 5.5 summarizes the accuracy and loss across all five folds.

Table 5. 5: Accuracy and Loss Across for NBG1

Fold	Accuracy	Loss
1	0.9666	0.0975
2	0.9736	0.0829
3	0.9525	0.1370
4	0.9648	0.0916
5	0.9754	0.0893
Average	**0.9666**	**0.0997**

This configuration achieved an average accuracy of 0.9666 (96.66%), which was also the highest average accuracy among the three configurations of the without background dataset (NBG1, NBG2, and NBG3). This indicates that it has a strong and consistent performance. A dropout rate of 0.3 which was considered as relatively low dropout rate maintains sufficient learning capacity while providing some sort of regularization. The small batch size of 16 allowed for more frequent weight updates, which could aid in generalization. This trend can also be observed when using the "with background" setting. A minimal average loss of 0.0997 supports the model's confidence in its predictions and prevents overfitting.

Final Combined Classification Report:

	precision	recall	f1-score	support
Blackbird	0.94	0.96	0.95	150
Bluetit	0.98	0.97	0.97	150
Carrion_Crow	0.87	0.95	0.91	131
Chaffinch	0.99	0.97	0.98	150
Coal_Tit	0.97	0.99	0.98	144
Collared_Dove	0.95	0.99	0.97	150
Dunnock	0.97	0.97	0.97	150
Feral_Pigeon	0.98	0.95	0.96	93
Goldfinch	1.00	1.00	1.00	150
Great_Tit	0.98	0.98	0.98	150
Greenfinch	0.99	0.99	0.99	150
House_Sparrow	0.96	0.97	0.97	150
Jackdaw	0.95	0.90	0.93	131
Long_Tailed_Tit	0.99	0.98	0.98	150
Magpie	1.00	0.98	0.99	92
Robin	0.98	0.97	0.97	150
Song_Thrush	0.97	0.97	0.97	150
Starling	0.99	0.95	0.97	150
Wood_Pigeon	0.97	0.95	0.96	150
Wren	0.97	0.97	0.97	150
accuracy			0.97	2841
macro avg	0.97	0.97	0.97	2841
weighted avg	0.97	0.97	0.97	2841

Figure 5. 7: Classification Report for NBG1

From Figure 5.7, the model obtained an overall accuracy of 0.97 (97%), with the macro average and weighted average for precision, recall and F1-score at 0.97. This suggests that all bird classes have continuously exhibited excellent performance. A high precision of 0.97 suggests that the model had only a few false positive predictions while a high recall of 0.97 illustrates that the model had a high ability to identify most of the actual bird samples. This therefore led to a balanced F1-score (0.97), meaning that both false positives and false negatives were minimal. Overall, all classes show an excellent classification results, especially Goldfinch with a perfect score of 1.00 for precision, recall and F1-score, only a few classes like Carrion_Crow (F1-score: 0.91) and Jackdaw (F1-score: 0.93) showed slightly lower performance, possibly as a result of visual resemblance to other species or class imbalance. In short, even with the absence of background, the model under NBG1 still handled the bird classification very well, proving the configuration's efficacy in obtaining relevant features purely from bird appearances.

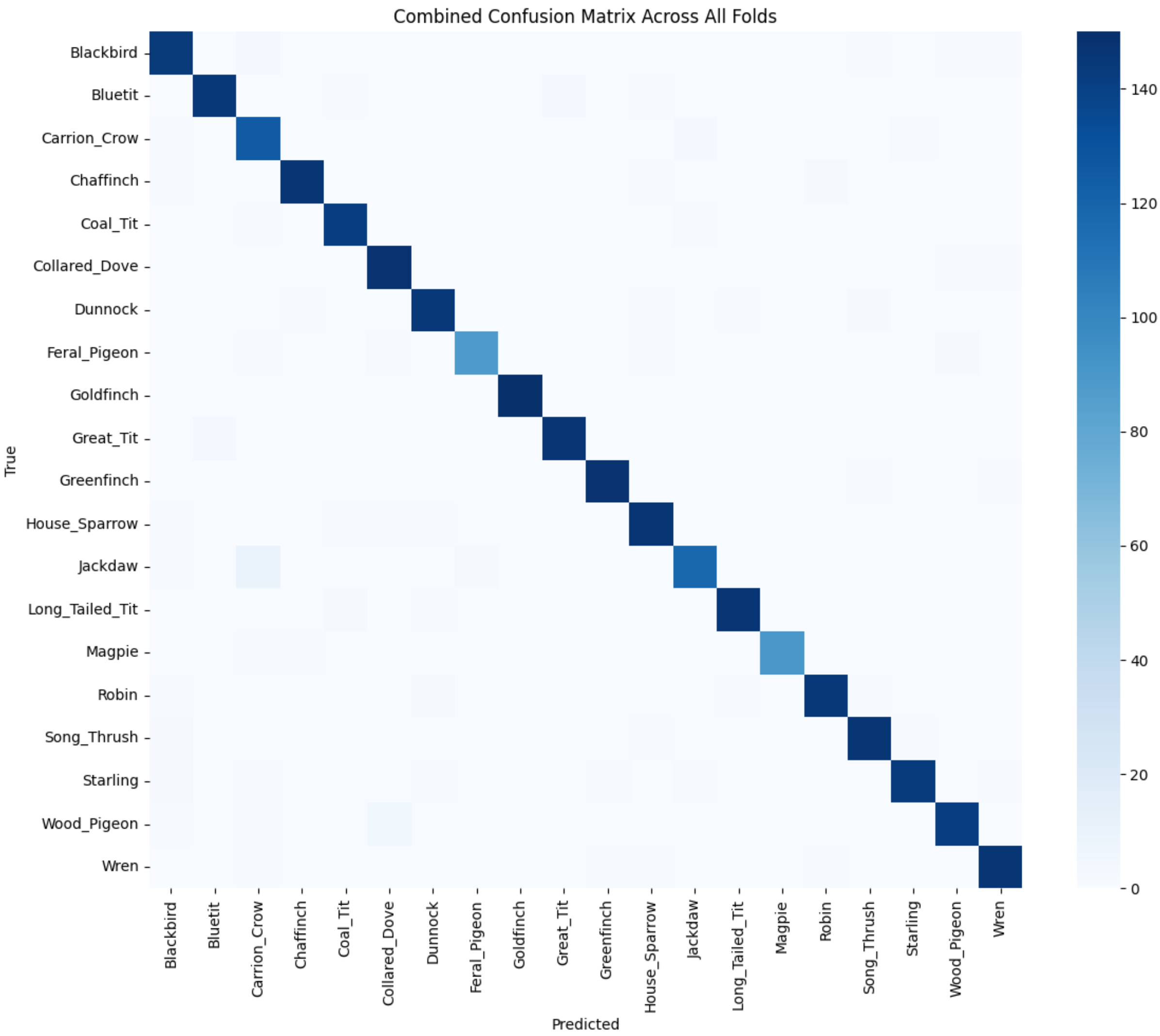

Figure 5. 8: Confusion Matrix for NBG1

Figure 5.8 above showed an overview of predictions for each class. Most values lie along the diagonal, indicates that most of the predictions match the true class. As can be clearly seen from the figure, there were only a few false positives and false negatives, as most of the matrix are entirely dark blue on the diagonal, which indicates a high accuracy for each class. Only some classes such as Jackdaw and Wood_Pigeon performed misclassifications, which also aligned with their precision, recall and F1-score in the classification report.

5.3.2 Configuration NBG2 (Dropout: 0.5, Batch Size: 32)

For this configuration NBG2, a dropout rate of 0.5 and batch size of 32 was trained the same on EfficientNetB0 model and performance was evaluated using 5-fold cross-validation over 20 epochs. The accuracy and loss across all five folds were summarized in Table 5.6 below.

Table 5. 6: Accuracy and Loss Across for NBG2

Fold	**Accuracy**	**Loss**
1	0.9613	0.1163
2	0.9683	0.1040
3	0.9595	0.1510
4	0.9577	0.1654
5	0.9613	0.1476
Average	**0.9616**	**0.1369**

With an average of 0.9616 (96.16%), this configuration had a performance slightly lower than the NBG1 and a higher average loss of 0.1369. With a slightly higher dropout rate of 0.5, it seems to bring increment in regularization, which might have the possibility to hinder the learning in a dataset as the model might not have enough capacity to learn the data effectively. A larger batch size of 32 tends to lead to a more stable gradient estimate as the model might be calculating the gradient based on more data points but at the same time, it will have less frequent updates, which might be impactful especially for smaller datasets. Overall, the model still performed well but not as optimally as NBG1.

Final Combined Classification Report:

	precision	recall	f1-score	support
Blackbird	0.92	0.95	0.93	150
Bluetit	0.97	0.97	0.97	150
Carrion_Crow	0.85	0.95	0.90	131
Chaffinch	0.99	0.92	0.95	150
Coal_Tit	0.98	0.98	0.98	144
Collared_Dove	0.94	0.99	0.96	150
Dunnock	0.95	0.96	0.95	150
Feral_Pigeon	0.97	0.92	0.95	93
Goldfinch	1.00	1.00	1.00	150
Great_Tit	0.97	0.96	0.96	150
Greenfinch	0.96	0.95	0.95	150
House_Sparrow	0.96	0.97	0.96	150
Jackdaw	0.94	0.90	0.92	131
Long_Tailed_Tit	0.98	0.99	0.98	150
Magpie	0.99	0.98	0.98	92
Robin	0.98	0.97	0.97	150
Song_Thrush	0.96	0.97	0.97	150
Starling	1.00	0.95	0.97	150
Wood_Pigeon	0.97	0.94	0.95	150
Wren	0.97	0.97	0.97	150
accuracy			0.96	2841
macro avg	0.96	0.96	0.96	2841
weighted avg	0.96	0.96	0.96	2841

Figure 5. 9: Classification Report for NBG2

Figure 5.9 above showed an overall accuracy of 0.96 (96%), which is a very strong performance. Both the macro and weighted average were 0.96, which indicates that model is balanced and performs well across all 20 bird classes. Classes like Goldfinch, Long_Tailed_Tit, and Magpie obtained precision, recall, and F1-score at nearly or at 1.00, meaning that these species can be identified by the model confidently and accurately. Carrion_Crow had the lowest precison (0.85) as sometimes it might get confused with other birds like Jackdaw which also have a darker appearance. Overall, the other classes had F1-scores between 0.95 and 0.97, indicating consistent reliability.

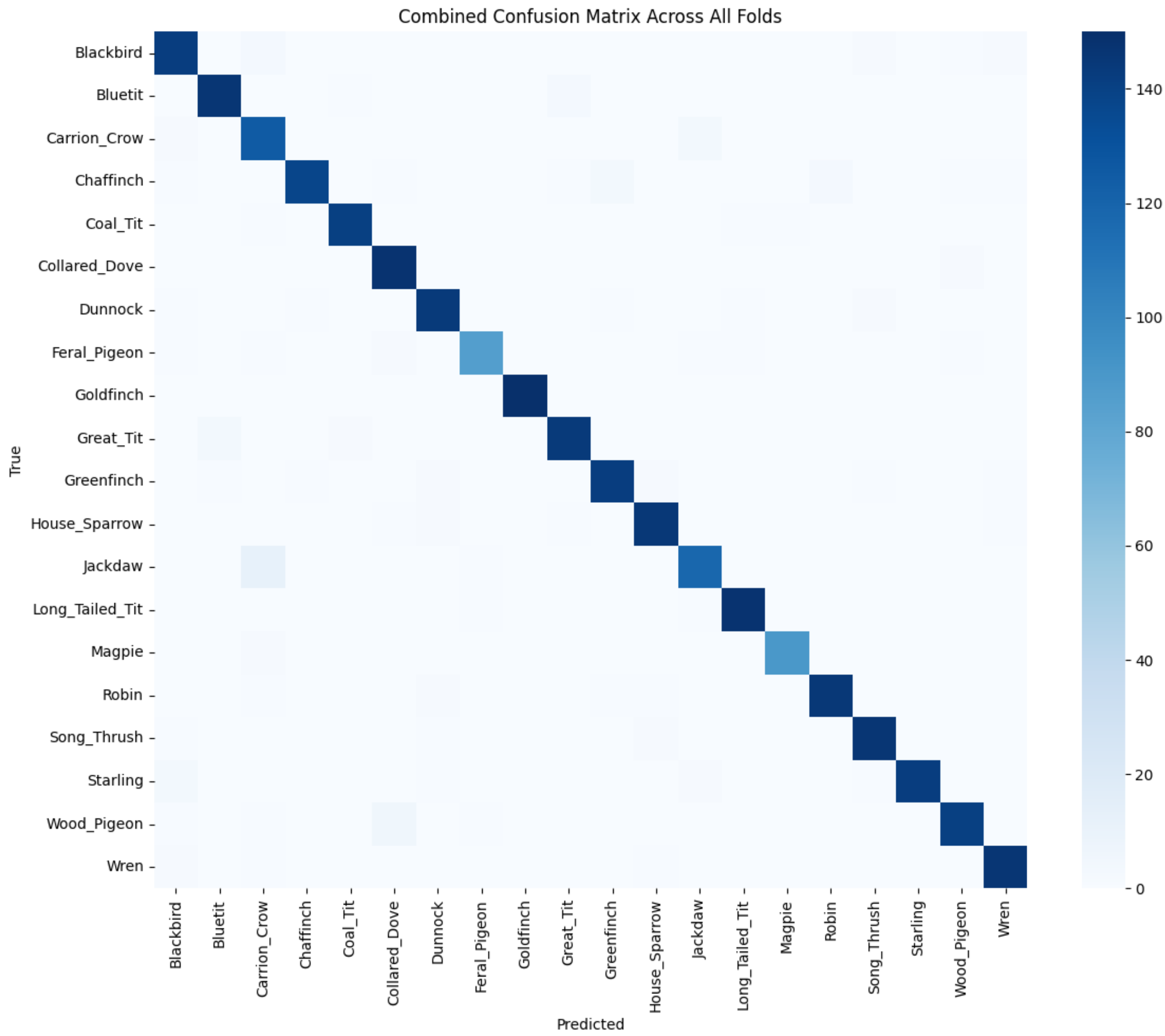

Figure 5. 10: Confusion Matrix for NBG2

Figure 5.10 above gave a picture of the model correctly predicted most bird images as most of the predictions lied along the diagonal with a dark blue color. Although most of the predictions were correctly predicted, there were still exceptions that revealed small amounts of confusion with similar-looking species. For example, the Carrion_Crow and Jackdaw with some faint-off diagonal squares. In a whole the confusion is not widespread, and the errors are still considered as minimal.

5.3.3 Configuration NBG3 (Dropout: 0.7, Batch Size: 64)

The next configuration is on the NBG3, training on EfficientNetB0 model with dropout rate of 0.7 and batch size of 64. The performance was then evaluated using 5-fold cross-validation over 20 epochs and the results on the accuracy and loss across five folds were summarized in Table 5.7 below.

Table 5. 7: Accuracy and Loss Across for NBG3

Fold	Accuracy	Loss
1	0.9649	0.1220
2	0.9525	0.1597
3	0.9630	0.1482
4	0.9577	0.1192
5	0.9560	0.1462
Average	**0.9588**	**0.1391**

This configuration gave the lowest performance 0.9588 (95.88%) across the three configurations in this dataset setting. Since 0.7 was the highest dropout rate among the three configurations, although it aids in preventing overfitting, it might still have the possibility of introducing too much noise, restricting the model's ability to learn in a dataset without background. This is because the model might rely on limited features to make predictions if the dataset has its background removed. Since many neurons are inactive during training due to a high dropout rate of 0.7 is applied, the model's ability to learn those important features during training might be limited too. Using a large batch size of 64 can lead to fewer updates and might result in the model not learning effectively from a slightly smaller and homogeneous dataset. This can then result in underfitting as the model fails to capture the necessary patterns, and therefore leading to a slightly poor performance of 0.9588 compared to the other configurations.

Final Combined Classification Report:

	precision	recall	f1-score	support
Blackbird	0.89	0.93	0.91	150
Bluetit	0.97	0.99	0.98	150
Carrion_Crow	0.88	0.91	0.89	131
Chaffinch	0.98	0.95	0.97	150
Coal_Tit	0.97	0.98	0.98	144
Collared_Dove	0.96	0.97	0.97	150
Dunnock	0.95	0.95	0.95	150
Feral_Pigeon	0.94	0.88	0.91	93
Goldfinch	0.99	1.00	1.00	150
Great_Tit	0.97	0.95	0.96	150
Greenfinch	0.94	0.95	0.95	150
House_Sparrow	0.97	0.97	0.97	150
Jackdaw	0.88	0.92	0.90	131
Long_Tailed_Tit	0.97	0.96	0.96	150
Magpie	0.97	0.99	0.98	92
Robin	0.98	0.95	0.97	150
Song_Thrush	0.97	0.98	0.97	150
Starling	1.00	0.95	0.97	150
Wood_Pigeon	0.95	0.95	0.95	150
Wren	0.99	0.97	0.98	150
accuracy			0.96	2841
macro avg	0.96	0.96	0.96	2841
weighted avg	0.96	0.96	0.96	2841

Figure 5. 11: Classification Report for NBG3

For configuration NBG3, the classification report depicted in Figure 5.11 had an accuracy of 0.96, which means that around 96% of all bird images were correctly classified. The macro average and weighted average on all metrics such as precision, recall and F1-score were also

0.96, which showed the model is balanced and performed well across all classes. Birds such as Bluetit, Goldfinch, Magpie and Wren are considered as high-performing classes as they had very high scores with all equal to or exceed a score of 0.97. This indicates that the model rarely made mistakes while identifying the birds. Blackbird with a precision of 0.89 and recall of 0.93 showed slightly lower performance as sometimes it might confuse with similar species. Other than that, Feral_Pigeon had a recall of 0.88, meaning some images of Feral Pigeon sometimes were classified as other birds. With scores ranging from 0.95 to 0.97, the majority of other classes, including Chaffinch, Collared_Dove, Dunnock, Great_Tit, Greenfinch, and Wood_Pigeon, exhibit consistently strong but slightly lower performance.

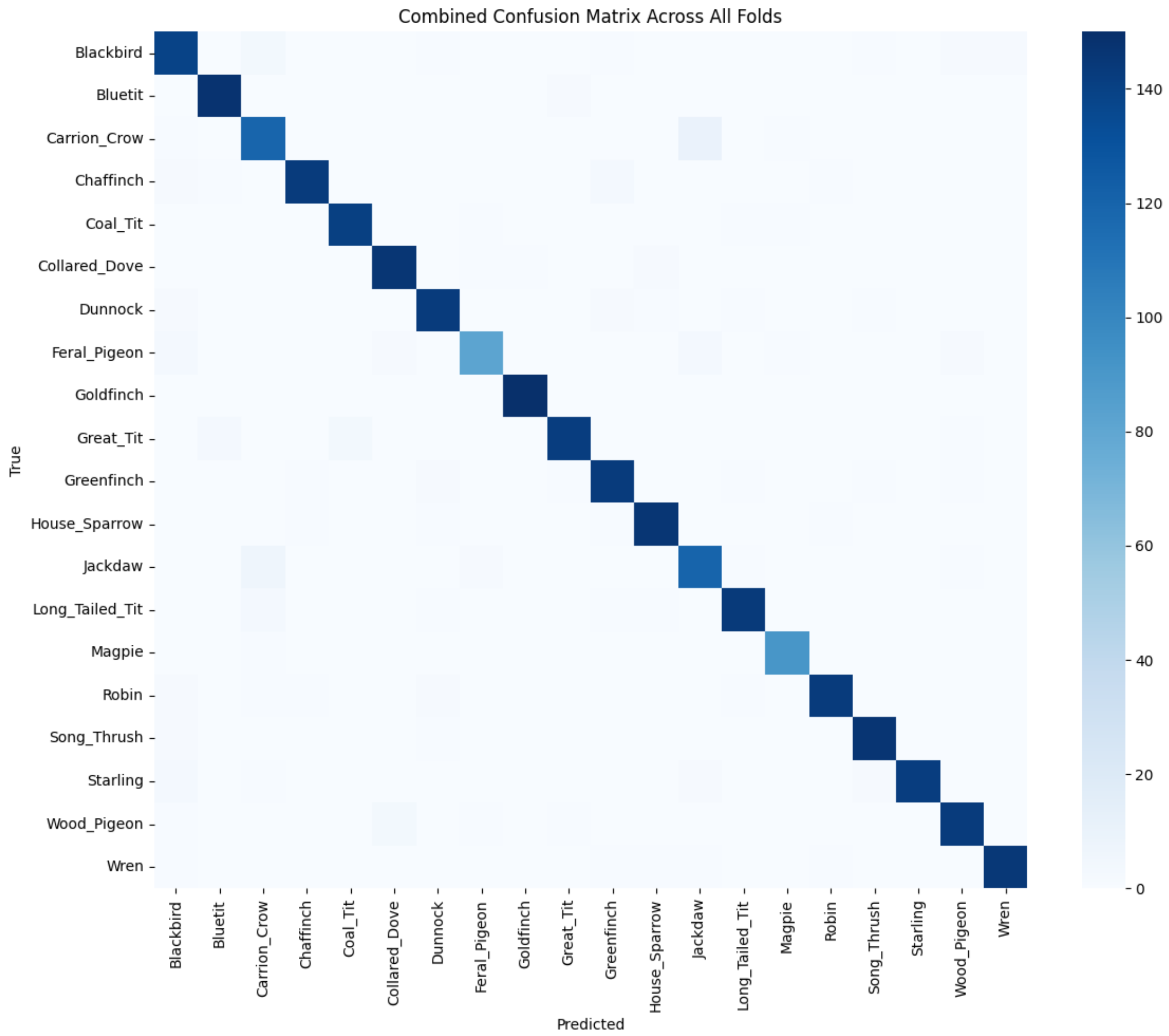

Figure 5. 12: Confusion Matrix for NBG3

Confusion matrix shown in Figure 5.12 above indicated that most of the birds' images were correctly predicted as most of them lied along the diagonal. But there were still some off-diagonal matrixes which were few misclassifications such as Carrion_Crow was predicted as Jackdaw, Jackdaw and Long_Tailed_Tit was sometimes predicted as Carrion_Crow. Besides that, Great_Tit was also sometimes misclassified as Bluetit and Coal_tit, mainly due to the similarity on the appearance of the bird species as those bird species had white patches which were alike on them. But in a whole, this configuration NBG3 still performed well across different classes.

5.3.4 Comparison Across Configurations without Background (NBG1–NBG3)

Table 5. 8: Comparison Across Configurations without Background

Configurations	Dropout	Batch Size	Avg Accuracy
NBG1	0.3	16	0.9666
NBG2	0.5	32	0.9616
NBG3	0.7	64	0.9588

From Table 5.8 above, it can be clearly shown that among the three configurations, NBG1 with a dropout rate of 0.3 and batch size of 16 achieved the highest average accuracy 0.9666 (96.66%), making it the best-performing configuration for the dataset without background. As dropout rate increased from 0.3 to 0.7, there was a slight degradation in performance as observed. This is likely due to higher dropout reduces the model's learning capacity by deactivating more neurons during training. As the without background dataset itself already provided less visual information, applying a high dropout may cause the model to overlook some important bird-specific features, leading to reduced accuracy. Similarly, increasing the batch size from 16 to 64 meant the model updated its weights less often. This could make it harder for the model to adapt to small differences between bird classes, especially because the background cues were already removed. In short, configuration NBG1 emerged to be the most effective setup for the without background dataset, demonstrating that a lower dropout and smaller batch size are better suited to obtaining the important features of bird species when background context is being removed.

5.4 Summary of Model Performance Across Configurations

Table 5. 9: Evaluation Metrics Across All Configurations

Configuration	Dataset	Dropout	Batch Size	Avg Accuracy	Avg Loss	Precision	Recall	F1-Score
BG1	With Background	0.3	16	0.9771	0.0778	0.98	0.98	0.98
BG2	With Background	0.5	32	0.9704	0.0916	0.97	0.97	0.97
BG3	With Background	0.7	64	0.9616	0.1209	0.97	0.96	0.96
NBG1	Without Background	0.3	16	0.9666	0.0997	0.97	0.97	0.97
NBG2	Without Background	0.5	32	0.9616	0.1369	0.96	0.96	0.96
NBG3	Without Background	0.7	64	0.9588	0.1391	0.96	0.96	0.96

Table 5.9 above shows a clear comparative overview of how different model configurations and dataset settings influenced performance. Besides, it presents a consolidated summary of evaluation metrics obtained using 5-fold cross-validation. Each configuration combined a particular dropout rate and batch size and was trained separately on either the "with background" or "without background" datasets.

From the Table 5.9, it shows that models trained with lower dropout rates and smaller batch sizes had higher accuracy and lower loss values, regardless of the dataset type. Notably, BG1 (with background, dropout 0.3, batch size 16) was the best-performing configuration, with an average accuracy of 0.9771 (97.71%) and an F1-score of 0.98, indicating that this combination promotes higher model generalization and less overfitting. These finding align with findings by Kandel and Castelli (2020), who found that smaller batch sizes led to improved generalizability in CNNs, especially in image classification tasks.

This section focuses on how internal model design factors affect performance across both datasets, providing the foundation for the following section's exploration of dataset-driven effects.

5.5 Comparison Across Dataset Settings: With vs Without Background

Table 5. 10: Comparison Across Dataset Setting

	Avg Accuracy With Background	**Avg Accuracy Without Background**
Dropout: 0.3, Batch Size: 16	0.9771	0.9666
Dropout: 0.5, Batch Size: 32	0.9704	0.9616
Dropout: 0.7, Batch Size: 64	0.9616	0.9588
Best Avg Accuracy	**0.9771 (BG1)**	**0.9666 (NBG1)**

In the context of evaluating the impact of dataset diversity on the model performance, a cross-dataset comparison was conducted using the same model configurations across both settings: with background (BG) and without background (NBG) as shown in Table 5.10 above. From the Table 5.10, it consistently shows that the dataset with background obtained a higher performance across all configs, and the best-performing model overall was the Configuration

BG1 with a dropout of 0.3 and batch size of 16 which obtained an average accuracy of 0.9771. This trend suggests that the presence of background information might aid in contributing additional contextual findings in improving the model's ability to identify between bird species, for example in terms of habitat, lightning or branch structure.

This finding gives a picture of how actually the inclusion or exclusion of background elements, mainly the dataset diversity, affects the performance of the transfer learning models in bird identification tasks. As seen from the table above, while models trained on background-removed images still performed strongly, the slight decrease in accuracy seen across all configurations suggests that removing background may somehow accidentally eliminate useful environmental features. These results highlight how maintaining background diversity might improve the robustness of species classification through transfer learning in especially certain real-world environments.

5.6 Comparison with Existing Bird Classification Models

This section provides a qualitative comparison of results from similar bird classification studies. The goal is not to compare models under identical conditions, but to place the proposed study's performance and design choices in the perspective of the larger research landscape. These selected works are either based on the same dataset, same model architecture, or apply transfer learning approaches similar to those used in this study. The studies include:

1. 20 UK Garden Birds Species Classification with ResNet50V2 (Mahony, 2023)
2. Bird Species Detection Using CNN and EfficientNetB0 (Neeli et al., 2023)
3. Bird Classification Using CNN on North American Bird Dataset (Sitepu et al., 2022)

The performance of the existing papers and studies will be discussed and summarized in the subsections below. Table 5.11 provides an overview of dataset properties, model types and their results.

5.6.1 Summary of Existing Models

This subsection briefly describes each of the existing studies with a focus on how their choice of model architecture and dataset size contributes to classification performance. These benchmarks provide insights into how dataset diversity and architecture choices influence transfer learning outcomes. A table summarizing the performance and key aspects of the existing and proposed models were presented in Table 5.11 below.

Table 5. 11: Performance of Existing and Proposed Models

Study	Dataset (Kaggle)	Number of Classes	Number of Images	Model (s)	Accuracy / F1
Mahony (2023)	20 UK Garden Birds	20	2841 (BG)	ResNet50V2	~94% (Accuracy)
Neeli et al. (2023)	525 Bird Species	525	84,635	EfficientNetB0	89% - 92% (Accuracy)
Sitepu et al. (2022)	200 North American Birds	200	11788	MobileNetV2	75% (F1-weighted)
				EfficientNetB0	81% (F1-weighted)
				EfficientNetB3	81% (F1-weighted)

Proposed Study (2025)	20 UK Garden Birds	20	2841 (BG) 2841 (NBG)	EfficientNetB0	97.71% (BG1) 96.66% (NBG1)

5.6.1.1 20 UK Garden Birds Species Classification with ResNet50V2 (Mahony, 2023)

This paper used the same dataset as the proposed model, which is the 20 UK Bird dataset, but applied technique of transfer learning to the ResNet50V2 model instead of EfficientNetB0. It also used the weights learned from the ImageNet dataset as a base model and fine-tuned using transfer learning techniques. Data preprocessing and image augmentation were carried out too. Image size was set to 224 and among the entire dataset, 70% was for training, 10% for validation, and 20% for testing. By using images with background throughout the experiment, the model achieved an accuracy of 94%, indicating a high performance in distinguishing the 20 bird species. Although it used a different architecture, this study validates the effectiveness of transfer learning on this specific dataset.

5.6.1.2 Bird Species Detection Using CNN and EfficientNetB0 (Neeli et al., 2023)

This study used Convolutional Neural Networks (CNNs), specifically the EfficientNetB0 architecture on a much larger dataset, which had a total of 525 species and around 84000+ images to tackle the challenge of automatic bird species classification. The EfficientNetB0 architecture in this study applied fine-tuning along with data augmentation and transfer learning. The data augmentation techniques used included rotation, scaling and cropping to enhance the robustness of the model. Training-validation split was implemented. The model was trained over 20 epochs using a batch size of 32. Besides, callbacks and early stopping were also incorporated to enhance the training. This study achieved a performance of 89.19% to 92.0% when identifying bird species, highlighting EfficientNetB0's scalability across diverse datasets.

5.6.1.3 Bird Classification Using CNN on North American Bird Dataset (Sitepu et al., 2022)

This study by Sitepu et al., 2024 was carried out to analyze comparative results from using three architectures, which are MobileNetV2, EfficientNetB0, and EfficientNetB3 with ImageNet-pretrained weights in recognizing the bird species or classes. A dataset on the North American Bird with a total of 200 species of birds was used in this study. The images were resized to 224 x 224 and ImageDataGenerator was used for data normalization. Each of the model performances were based on the weighted F1-scores: MobileNetV2 (75%), EfficientNetB0 (81%), and EfficientNetB3 (81%).

5.6.2 Comparative Analysis and Discussion

This section includes a qualitative discussion that compares the proposed model's results to those from three related bird classification studies. These comparisons are not empirical benchmarks, because each study employed different datasets, data quantities, and model architectures. In contrast, the present study uses EfficientNetB0 to isolate the effect of background variance on a single, consistent dataset. The aim of this comparison is to provide contextual understanding and to place the project's findings within the larger research landscape of bird species classification using transfer learning.

a. Difference in Dataset Scale (Same Model Architecture)

Discussing the model performance across the dataset sizes, the study by Neeli et al. (2023), used a large dataset with a total of 525 bird species. Applying the same transfer learning model as the proposed model, EfficientNetB0 with data augmentation but implementing a training-validation split instead of the K-fold cross validation, the model achieved an accuracy around 89% to 92% when identifying birds which demonstrates strong generalization, especially when it is being trained over a high number of classes. In contrast, this study used a smaller dataset (20 species) and achieved higher accuracy: 97.71% (with background) and 96.66% (without background). While this may indicate improved performance, it is most likely impacted by the more controlled experimental setting: fewer classes, more consistent images, and balanced class distributions.

b. Difference in Model Architecture (Same dataset, same variation used)

Moving on to the comparison among architecture, the study carried out by Mahony (2023) used the same dataset (20 UK Garden Bird), as the proposed model, but a different model architecture, ResNet50V2. Although the same dataset (background dataset) was used, the study by Mahony (2023) obtained an accuracy of around 94% by using the dataset with background throughout the experiment while the proposed model using EfficientNetB0 on the same background dataset obtained an accuracy of 97.71%. This difference could indicate that EfficientNetB0 is more efficient or better suited to small-class, fine-grained classification jobs. However, because exact training sets and augmentation procedures differ, this observation is only informative, not conclusive.

c. Comparison Across Multiple Architectures on a Different Dataset

In contrast, the study brought by Sitepu et al. (2024) explored on multiple models including MobileNetV2, EfficientNetB0, and EfficientNetB3 on a dataset of 200 North American bird species. Among the three models used, both EfficientNet models obtained a higher F1-weight score, 81% compared to the MobileNetV2, 75%. Although their datasets and classification contexts differ, these findings support the EfficientNet family's overall robustness in bird classification, particularly in managing class imbalance while offering consistent results across multiple classes.

d. Impact of Image Context (Background vs. No Background)

For the impact of dataset context, the proposed study is the only one that evaluated on same dataset with and without background. The accuracy dropped slightly from 97.71% to 96.66% when the background was removed. This suggests that, while background context can provide slight cues that boost performance, the EfficientNetB0 model is nevertheless robust when those cues are eliminated. This finding is valuable in real-world deployments, where background complexity does vary. It also shows that dataset diversity in terms of image context will impact the model performance.

5.7 Summary

This chapter concludes with a comprehensive summary of the results which is the performance on dataset with background and without background when experimenting through each of the configurations and the discussion made. A comparison across configurations for each of the dataset setting were discussed. The best performance for both dataset setting was from Configuration 1, with a dropout of 0.3 and batch size of 16. The model performed slightly better with background but remain robust without it. This demonstrated that the image context (background) has a minor but noticeable effect on the performance. This chapter also discussed on the other three existing studies or models with the proposed model, either based on the same dataset used, same model architecture, or applying the transfer learning approaches similar to the proposed model. Furthermore, the comparative analysis with three relevant studies demonstrated that the proposed model outperforms existing work in terms of accuracy and efficiency, even with fewer classes and a smaller dataset. The choice of choosing the best-performing configuration for final deployment among background datasets was also being justified in this chapter.

CHAPTER 6 CONCLUSION

6.1 Introduction

This chapter concludes the study by summarizing the key findings, contributions, limitations, and outlining directions for future research.

6.2 Contribution

This present project makes several significant contributions to the field of computer vision and biodiversity, especially in species identification. It was said that this project provides a comparative performance analysis of a transfer learning model, EfficientNetB0 across datasets with different contextual characteristics (with and without background). This study also demonstrated that EfficientNetB0 is highly effective in datasets with smaller sizes when dealing with classification tasks such as bird species identification. It also proved that dataset context, which is the presence of background, has a measurable but slightly minor impact on model accuracy. A structured experimental setup using the 5-fold cross validation and various configurations were proposed to evaluate the model generalization. The best-performing model among the with background configuration was then chosen to be deployed into a simple local web application to allow practical identification of birds by uploading the images.

6.3 Limitations

Despite these contributions, the study also has limitations that should be acknowledged. Firstly, the study utilized only the 20 UK Garden Bird dataset from Kaggle, which may not represent the global population of bird species. This may therefore restrict the models' generalizability to a wider range of bird species.

In addition, the system used a closed-set classification framework, which means it can only categorize input images of birds into one of the 20 known classifications. As a result, if a user uploaded an image of a bird species not included in the training dataset, such as a bird from another region, the model will still make a prediction based on the 20 known species, though inaccurately. This drawback highlights the lack of an out-of-distribution (OOD) detection method, which would enable the model to recognize when an image falls outside of its learnt categories and avoid forced misclassification.

Besides that, the study focused solely on one transfer learning model, EfficientNetB0, without exploring other methods or architectures, which may limit the applicability of the findings to a broader range of scenarios.

Although both dataset variations (with and without background) were utilized to evaluate visual diversity, all images were reasonably clean and carefully cropped, which may not accurately represent the noise and variability encountered in real-world environments. Lighting variations, weather conditions (e.g., fog, rain), and occlusion (e.g., limited visibility due to foliage) were not explicitly addressed, which could have an impact on the model's robustness in uncontrolled field deployments.

6.4 Future Works

Building upon the findings and limitations of this study, several avenues for future work are recommended. Firstly, to overcome the limitation of a relatively small dataset, efforts should be made to incorporate a larger and more diverse bird datasets, for example global species datasets, to improve on the generalization across different bird populations. Open datasets like

CUB-200-2011, NABirds, or iNaturalist can be considered to use to provide a wider variety of bird species across different regions. With the expanded dataset, better training of the model and improved generalizability to unseen data would be made possible.

Furthermore, a crucial future improvement would be to include mechanisms for recognizing out-of-distribution (OOD) inputs. Because the current model is based on a closed-set assumption and will categorize any bird input as one of the 20 known UK bird species, it cannot reject or flag unfamiliar species. Confidence thresholding, softmax-based entropy filtering, and embedding-space distance metrics can be implemented to allow the model to avoid prediction or flag uncertain inputs. Integrating OOD detection helps reduce incorrect classifications and enhance trustworthiness in real-world applications, especially in global or open-ended deployment circumstances.

Besides that, exploration of additional model architectures enables a comparative performance analysis, which allows the researchers to identify the most suitable model for specific deployment contexts, such as lightweight models for mobile applications or more powerful models for cloud-based systems.

Moreover, in order to enhance the model's robustness in real-world scenarios, real-world image variations should be included. To address this, future work should consider augmenting the existing dataset or collecting new images that reflect such challenging conditions, for example, in varying light intensities, in different weather conditions (fog, rain, cloudy), and also occlusion. The model can learn to identify birds even when the subject is not clear or fully visible by using images with blur or partial views.

6.5 Conclusion

In conclusion, this chapter summarized the main contributions of the study to computer vision and biodiversity, specifically in bird species identification using transfer learning. This study demonstrated that EfficientNetB0 is effective in small-scale datasets and resilient across different image contexts (with and without background). However, limitations exist, including the dataset scope, which was limited to UK bird species, the issue on the out-of-distribution (OOD), the lack of real-world environmental variability, and the limited model exploration. To address these limitations, future works was proposed to enhance the generalization. This includes using larger and more diverse datasets, implementing the integration of OOD detection mechanisms, testing on additional architectures, and incorporating real-world image conditions. These reflections guide how the current work might be broadened or enhanced in future studies and practical deployments.

The study successfully achieved its stated objectives. First, the EfficientNetB0 model's performance was thoroughly assessed across different configurations and dataset variants, with the aim of determining its efficacy in identifying bird species under varying background conditions. Second, by examining both qualitative and quantitative results, including accuracy trends, F1-scores, and cross-dataset comparisons, the study provided clear insights into how dataset diversity, specifically the presence or absence of background elements, influences model generalization and classification accuracy. These findings shows that model architecture and dataset composition play an important role in transfer learning-based bird identification, with the outcomes perfectly aligned with the project's initial objectives.

REFERENCES

Alswaitti, M., Zihao, L., & Alomoush, W. (2022). *Effective classification of birds' species based on transfer learning*. International Journal of Electrical and Computer Engineering, 12(4), 4172–4184. https://doi.org/10.11591/IJECE.V12I4.PP4172-4184

Awasthi, A., & Verma, A. (2023). Bird species identification using deep learning. *International Journal of Engineering Research & Technology (IJERT)*, *8*(1), Article 112. https://www.ijert.org/research/bird-species-identification-using-deep-learning-IJERTV8IS040112.pdf

Beery, S., Van Horn, G., & Perona, P. (2018). *Recognition in terra incognita*. Proceedings of the European Conference on Computer Vision (ECCV), 456–473. https://doi.org/10.1007/978-3-030-01234-2_28

Caruana, R., & Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. *Proceedings of the 23rd International Conference on Machine Learning (ICML)*, 161–168. https://doi.org/10.1145/1143844.1143865

Chan, M. (2024). *Agile development of AI defect detection using CNN*. Semiconductor Review. https://system-on-chip-soc-apac.semiconductorreview.com/cxoinsight/agile-development-of-ai-defect-detection-using-cnn-nwid-409.html

Chaturvedi, K., Skantha, M. N., & Harissh, G. (2024). *Deep learning-based identification of birds in agricultural fields*. IEEE. Retrieved from https://ieeexplore.ieee.org/abstract/document/10921862/

Chollet, F. (2023). *Transfer learning & fine-tuning*. Keras. https://keras.io/guides/transfer_learning/

Dalal, N., & Triggs, B. (2005). "Histograms of Oriented Gradients for Human Detection." *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 1, 886-893.

Farou, Z., & Reslan, A. (n.d.). *Automatic fine-grained classification of bird species using deep learning*. Retrieved from https://www.researchgate.net/publication/366192757_Automatic_Finegrained_Classification_of_Bird_Species_Using_Deep_Learning

Garnett, S. T., Ainsworth, G. B., & Zander, K. K. (2018). Are we choosing the right flagships? The bird species and traits Australians find most attractive. *PLoS One, 13*(6), Article e0199253. https://doi.org/10.1371/journal.pone.0199253

GeeksforGeeks. (2025). *Use case diagram - Unified modeling language (UML)*. https://www.geeksforgeeks.org/use-case-diagram/

Ghani, B., Kalkman, V. J., Planqué, B., Vellinga, W. P., & Gill, L. (2025). *Impact of transfer learning methods and dataset characteristics on generalization in birdsong classification*. Scientific Reports. https://doi.org/10.1038/s41598-025-00996-2

Ilemobayo, J. A., Durodola, O. I., Alade, O., Awotunde, O. J., Adewumi, T. O., Falana, O., Ogungbire, A., Osinuga, A., Ogunbiyi, D., Odezuligbo, I., Edu, O. E., & Ifeanyi, A. (2024). *Hyperparameter tuning in machine learning: A comprehensive review*. https://www.researchgate.net/publication/381255284_Hyperparameter_Tuning_in_Machine_Learning_A_Comprehensive_Review

Kancharla, A., &Pannala,A.(2019). Factors for accelerating the development speed in systems of artificial intelligence. *DiVA Portal.*https://www.diva-portal.org/smash/record.jsf?pid=diva2:1335130

Kandel, I., & Castelli, M. (2020). *The effect of batch size on the generalizability of convolutional neural networks on a histopathology dataset*. Data in Brief, Elsevier. https://www.sciencedirect.com/science/article/pii/S2405959519303455

Kensert, A., & Harrison, P. J. (2019). Transfer learning with deep convolutional neural networks for classifying cellular morphological changes. *SLAS Discovery, 24*(4), 466–475. https://doi.org/10.1177/2472555218818756

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). *ImageNet classification with deep convolutional neural networks*. Advances in Neural Information Processing Systems, 25, 1097–1105. https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf

Lederer, R. (n.d.). Identifying birds with GISS or JIZZ. Ornithology. https://ornithology.com/identifying-birds-with-giss-or-jizz/

Lowe, D. G. (2004). "Distinctive Image Features from Scale-Invariant Keypoints." *International Journal of Computer Vision*, 60(2), 91-110.

Mahony, D. (2023). *20 UK garden birds*. Kaggle. https://www.kaggle.com/datasets/davemahony/20-uk-garden-birds

Manna, A., Upasani, N., Jadhav, S., & Mane, R. (2023). *Bird image classification using convolutional neural network transfer learning architectures*. Retrieved from https://www.researchgate.net/publication/369809168

Martinez, M. F. S. (2024). *Comparative analysis of deep learning architectures for fine-grained bird classification* (Bachelor's thesis). University of Twente. Retrieved from http://essay.utwente.nl/101313/

Mertins, B. (n.d.). How to identify birds (Observation + experience). Nature Mentor. https://nature-mentor.com/bird-identification/

Neeli, S., Guruguri, C. S. R., Kammara, A. R. A., Annepu, V., Ch, V. R. R., & Bagadi, K. (2023). *Bird species detection using CNN and EfficientNet-B0*. ResearchGate. https://www.researchgate.net/publication/378269409_Bird_Species_Detection_Using_CNN_and_EfficientNet-B0

Netvue. (n.d.). About us | Netvue: Home security. Done smart. https://www.netvue.com/pages/about

Netvue. (n.d.). Birds identify. https://my.netvue.com/en/activities/birds-identify

Nikfal,N.(2024). A development process framework for artificial intelligence/machine learning (AI/ML)-based connected health informatics (Doctoral dissertation, Purdue University). *ProQuest Dissertations Publishing.*

Nyckel. (n.d.). Identify birds using AI. https://www.nyckel.com/pretrained-classifiers/bird-identifier/#:~:text=Below%20is%20a%20free%20classifier%20to%20identify%20birds.,tool%20will%20then%20predict%20what%20bird%20it%20is.

PitchBook. (2024). *Nyckel overview*. https://pitchbook.com/profiles/company/493367-41#overview

Sitepu, A. C., Liu, C.-M., Sigiro, M., Panjaitan, J., & Copa, V. (2022). A convolutional neural network bird's classification using north American bird images.International Journal of Health Sciences,6(S2), 15067–15080. https://doi.org/10.53730/ijhs.v6nS2.8988

Tan, M., & Le, Q. V. (2019). *EfficientNet: Rethinking model scaling for convolutional neural networks* (arXiv:1905.11946v5). https://arxiv.org/abs/1905.11946

Vardhan, H. (2024). *Bird identifier by picture*. AppAdvice. https://appadvice.com/app/bird-identifier-by-picture/6478655064

Vo, H.-T., Nguyen Thien, N., & Mui, K. C. (2023). Bird detection and species classification: Using YOLOv5 and deep transfer learning models. *International Journal of Advanced Computer Science and Applications*, 14(7). https://doi.org/10.14569/IJACSA.2023.0140701

Wang, Y. (2024). *Enhancing the classification of Finnish bird images through deep learning: A comparison and optimisation study of CNN architectures* (Bachelor's thesis). LUT University. Retrieved from https://lutpub.lut.fi/handle/10024/167475

Yang, C.-L., Harjoseputro, Y., Hu, Y.-C., & Chen, Y.-Y. (2022). An improved transfer-learning for image-based species classification of protected Indonesians birds. *Computers, Materials & Continua*, 73(3), 4577-4593. https://doi.org/10.32604/cmc.2022.031305

Yang, F., Shen, N., & Xu, F. (2024). Automatic bird species recognition from images with feature enhancement and contrastive learning. *Applied Sciences, 14*(10), 4278. https://doi.org/10.3390/app14104278