FLOWER CLASSIFICATION USING DEEP LEARNING: A Web-Based API for Accurate and User-Friendly Identification

By
Mafruha Mamun Mahiya
ID: 211-15-14679

FINAL YEAR DESIGN PROJECT REPORT

This Report Presented in Partial Fulfillment of the Requirements for the **Degree of Bachelor of Science in Computer Science and Engineering**

Supervised by

Supervisor

Dr. Fizar Ahmed
Associate Professor
Department of Computer Science and Engineering
Daffodil International University

Co-Supervised by

Md. Jakaria Zobair
Lecturer
Department of Computer Science and Engineering
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY
Dhaka, Bangladesh

January 12, 2025

APPROVAL

This Project titled **"Flower Classification Using Deep Learning: A Web-Based API for Accurate and User-Friendly Identification"**, submitted by Mafruha Mamun Mahiya, ID No: 211-15-14679 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 12 January 2025.

BOARD OF EXAMINERS

Dr. Sheak Rashed Haider Noori — **Chairman**
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Sharmin Akter — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Muhammed Masum Bakaul — **Internal Examiner**
Senior Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Dr. Md. Zulfiker Mahmud — **External Examiner**
Professor
Department of Computer Science and Engineering
Jagannath University

DECLARATION

We hereby declare that this project has been done by us under the supervision of **Dr. Fizar Ahmed, Associate Professor,** Department of Computer Science and Engineering, Daffodil International University. We also declare that neither this projectnor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:

Dr. Fizar Ahmed
Associate Professor
Department of Computer Science and Engineering
Daffodil International University

Co-Supervised by:

Jakaria
11. 01. 2025

Md. Jakaria Zobair
Lecturer
Department of Computer Science and Engineering
Daffodil International University

Submitted by:

Mafruha Mamun Mahiya

Mafruha Mamun Mahiya
ID: 211-15-14679
Department of Computer Science and Engineering
Daffodil International University

ACKNOWLEDGEMENTS

This work would not have been possible without the support and contributions of many individuals over the past two semesters. We are deeply grateful to everyone who has assisted us in one way or another.

First, we express our heartfelt thanks and gratefulness to the almighty for His divine blessing making it possible for us to complete the **Final Year Design Project (FYDP)** successfully.

We are grateful and wish our profound indebtedness to **Dr. Fizar Ahmed**, **Associate Professor**, Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh. Deep knowledge and keen interest of our supervisor in the field of **AI & Machine Learning** to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting themat all stages have made it possible to complete this project.

We would like to express our heartfelt gratitude to the Head of the Department of Computer Science and Engineering, for his kind help in finishing our project and also to other faculty members and the staff of the Department of Computer Science and Engineering, Daffodil International University.

We would like to thank our entire course-mates at Daffodil International University, who took part in this discussion while completing the coursework.

Finally, we must acknowledge with due respect the constant support and patience of our parents.

ABSTRACT

It is important to classify flowers because it forms a major part of botanical studies, agriculture and study of the environment among others. The idea to be developed in this project can be referred to as Flower Classification with Deep Learning where rather than describing the species in detail, its identification on the basis of image data with the help of deep learning algorithms will be the main goal. Through convolutional neural networks (CNNs) of this project, a user can classify flowers by entering an image into the API interface created using Streamlit. The work started by capturing the flower image dataset, data enhancement including, changing contact, resizing of the input images of size 224×224, adjusting the gamma values of the images and data augmentation. The findings confirm the promises of deep learning in tackling challenging visual classifiers and offer a growing point for improvement for future work on self-driving plant species identification. This work is able to pragmatically serve as a tool for classifying flowers in a way that makes this activity considerably less time-consuming than it has been hitherto; therefore, this work is recommended to all those who focus on plant sciences, whether they are researchers, teachers, or amateurs.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Classification of flowers has always been an important process in such fields as botany, agriculture, and horticulture. Identification of flowers is made accurately; conservation of the species is achieved, improvement of agricultural yield is made possible, and environmental management is facilitated. But manual classification does require knowledge, time, and work effort sometimes from a specialist. The classification of flowers, in particular, has been enhanced with better enhancements in artificial intelligence popularly known as deep learning. This project, Flower Classification with Deep Learning, helps to classify flowers with a help of a convolutional neural network (CNN) implemented in the interface of the application. Due to the enhanced accuracy of deep learning and careful pre-processing of the dataset, this project attains high classification accuracy of flowers. For non-technical users or those who do not have time to download and learn about machine learning, I developed a web-based API interface hosted in Streamlit where anyone can upload flower images and within two minutes get their classifications. Several deep learning models are investigated in the context of the project, with the aim to assess their performance in identifying the best solution.

1.2 Motivation

There is an increasing demand for accurate and efficient flower detection and classification methods, which acts as a motivation to develop a deep learning-based flower classification system. Now, why are flowers such a huge deal in botany, agriculture, horticulture, conservation, medicine, etc.? Identifying flowers by hand is labor-intensive and often necessitates superior knowledge of plants, rendering it unsuitable for large-scale implementations. Recent developments in deep learning and computer vision suggest a potential pathway to understanding flowers through automated classification with great accuracy. From InceptionV3, MobileNet and VGGNet have tried and confirmed to perform very well on image recognition task, and thus, this project will utilize the very same technologies. This project, therefore, applies these advanced methodologies for developing an accurate and usable tool.

In addition, the personal drive that goes into this project is quite meaningful. Aside from a final defense project for academic completeness, building this system provides the outside world with a resource that can assist with environmental preservation, localized biodiversity efforts, and educational growth. As AI is becoming more pervasive for industry use cases, this project will help you develop your skillset with AI, deep learning, and APIs. Global tech accessibility is a major driver. Leveraging on the power of Python and Streamlit, they made sure that the system was smooth for users that do not have much technical background. Anyone from a student to a professional botanist would be able to upload an image of a flower and instantly have it classified. In short, this project is driven by a versatile set of motivations, both academic and personal, leading to its broader objective of creating a technological tool, designed to support society in adopting and promoting flower identification.

1.3 Objectives

This project main goal is to create an efficient and accurate flower classifier using deep learning techniques. This dataset was specifically designed to overcome the difficulty of manually identifying flowers and is an automated tool to classify flower species based on images. The tool is intended to help researchers, botanists, educators, and enthusiasts by making flower identification fast, accurate, and accessible.

In order to accomplish this objective, the project emphasizes on preprocessing steps, implementing advanced processing techniques like contrast enhancement, gamma correction, resizing, and data augmentation for a robust and extensive dataset. To achieve this, several state-of-the-art neural network architectures such as InceptionV3, MobileNet, VGG16, and VGG19, are evaluated to choose a model that provides the highest accuracy, resulting in the selection of InceptionV3 as the best candidate for the task.

An important part of this work was to implement a simple API interface based on Streamlit to allow even non-technical users to interact with the system. This system is designed to work in an intuitive and effective way such that a user just has to upload an image and he will get the classification results in no time. Moreover, the project aims to advance scientific and environmental initiatives by facilitating biodiversity research, conservation initiatives, and botanical education. This project as a whole is not merely a matter of fulfilling academic requirements the intent is to create a practical application of deep learning technology for the purpose of addressing real-world challenges in the area of flower classification. Integrating innovation, accessibility, and functionality, the project embodies the capacity of artificial intelligence to address challenges and promote greater social goods.

1.4 Methodology

Methodology like other projects, this project also follows a proper methodology. Image preprocessing comprises the following steps: enhance the contrast of the input images, resizing all images to the same dimension, perform gamma correction so that they have the same brightness level, and apply data augmentation to make the dataset more diverse. This is to ensure that the dataset is clean and appropriate for developing deep learning models.

Next, we train a series of deep learning models, and then compare their performance in classifying flowers. Advanced neural network architectures like InceptionV3, MobileNet, VGG16, and VGG19 We used the dataset, which is partitioned into training, testing and validation sets for accurate evaluation of model performance. Each of them is fine-tuned, and the one that has the best performance is selected for deployment.

After model selection, we deploy the best model to an API interface built through Streamlit. And eureka we have an interface through which user can upload image and it will output instant results so it can be used in real time. We run end-to-end testing on the complete interface and model to ensure their reliability and correctness.

The last step of this process is to write about the development process, detailing any problems faced during this process and how they were overcome. This methodology is critically important in the design of a scalable and user-friendly flower classification model using state-of-the-art technologies with real-world applications for research, education, and environmental conservation.

1.5 Project Outcome

The expected impacts of this project are of technically, socially and academically significant and innovative merit, which creates the potential for further progression. Below are the detailed outcomes:

1. Functional Flower Classification System:

The outcome of the project will be an efficient and accurate deep learning-based system for identification of flower species using images uploaded on the system. These mechanisms will greatly help to ease the normally tedious work of assigning identities to flowers.

2. User-Friendly Interface:

The development of streamlit based API interface is made to ensure that any person can be able to use the system including those with little skills in technology. The evaluated and measured response from the users is that the image processing and analysis is uncomplicated and quick in giving the users an accurate classification of an image that they upload.

3. Educational Contribution:

If properly developed, the system can help educators and students as tool for teaching and learning plant taxonomy and diversities. It can also be adopted in compartmental academic assignments and horticultural investigations in order to enhance realism.

4. Support for Research and Conservation:

The system provides an automated and scalable solution to address the problem of classifying and documenting the floral species by researchers. It also plays a role of promoting the conservation of various flower species through availing information and easy methods of recording various flower species.

5. Technical and AI Development:

This project enables the demonstration of sophisticated preprocessing approaches together with contemporary neural networks and an innovative and efficient human interface. It also proof the ability and application of deep learning framework in solving real classification problems.

6. Scalability and Future Enhancement:

The choice of a modular element configuration makes it possible to expand the database by including more flower species into analysis. It can also be extended for mobile applications or to subscribe other cloud services for more general usage.

7. Societal and Environmental Impact:

The project demystifies flower classification and thus increases awareness on environmental issues as well as the appreciation of different forms ofnature. It may also help promote citizen science projects because people can use the app to recognize flowers around them.

8. Academic Achievement:

The project meets academic needs for presentation and argues mastery of deep learning, preprocessing, model assessment, and API creating. Indeed, it proves the use of AI for the practical task, learning how the student works, and his creativity.

In solving these various aims, the project not only provides an efficient technical end product but also provides clear benefits to society, education and conservation. It paves tomorrow's ground, and application progress in related arenas.

1.6 Organization of the Report

The report is structured into the following chapters:

Chapter 1: Chapter I: Background of the Study; Statement of the Problem; Purpose of the Study; Scope and Limitations of the Study; Organization of the Report.

Chapter 2: Literature Review: Discusses prior work in flower classification and reviews prior work on Convolutional Neural Networks and image processing.

Chapter 3: It also describes the process followed to gather, clean and prepare the data to meet the requirements of the method and the design specifications. It also outlines the system requirements, CNN architectures used, and design issues as regards the development of the API interface.

Chapter 4: Implementation: Explains how the project can be done practically from building a model to testing the model, and using APIs.

Chapter 5: Results and Analysis: This section outlines the classification results obtained in the study and comparative assessment of the performance of developed models?

Chapter 6: Social and Ethical considerations: Emphasizes the positive effects of the automation of flower classification on the society and these on sustainability.

Chapter 7: Conclusion and Future work: Outlines the successes of the project, delimitations and recommendation of improvements and extensions of the project

Chapter 2

Background

2.1 Introduction

Due to the complexity and difficulties associated with attracting an accurate solution for flower classification problems, a sub-field of artificial intelligence referred to as deep learning has become foundational when it comes to automating and consequently improving this classification task. This chapter reviews the major work that has been done by researchers in this field of study, the resemblance of techniques used, the progress accomplished and the emerging research issues that have not yet been solved. It focuses on aspects of transfer learning, details of custom architectures, and pre-processing techniques that have enhanced flower classification to improved accuracy and efficiency.

2.2 Literature Review

The field of flower classification has grown significantly due to the advancement use of Deep learning in classification. Many authors have studied CNN and TL methods to improve the classification performance and efficiency of the models. Krizhevsky et al. (2012) first introduced using of deep CNNs for large scale image classification then a number of works such as Simonyan et al. (2015),Szegedy et al. (2016) extended this work through the development of VGGNet and InceptionV3. These models have shown relatively high levels of accuracy when used in flower datasets primarily because they can perform feature extraction in a hierarchical manner.

In 2017, Howard et al. presented MobileNet that was particularly designed to be computationally lightweight so as to be practical on mobile devices. In the same manner, Tan et al. (2019) created EfficientNet where the objective of the models' improvement is a combination of accuracy and size. Some comparative studies Patel et al. (2019) and Khan et al. (2020) conducted for flower classification found out transfer learning with ResNet and Inception pre-trained models boosts the accuracy significantly. Other approaches like the use of data augmentation techniques have also really helped to improve the stability of models developed. Their research follows the results of different techniques using convolutional neural networks on tasks such as flipping, rotation, and cropping by Sharma et al. (2019). Researchers Alam et al. (2021) and Gupta et al. (2021) worked with ensemble approaches that integrated models such as DenseNet or ResNet and reported even better accuracy at the cost of more computational time. In order to overcome this problem, Khan et al. (IWAFGC 2020) and Joshi et al. (IWAFGC 2018) proposed feature fusion approaches which are meant to integrate complementary information detected by different architectures. On the other hand, compact architectures such as MobileNet have been used for real-time applications as observed by Sathe et al., (2022). Chudasama et al. (2020) stressed the importance of the proposed blended model, which incorporated conventional preprocessing procedures and the current deep learning techniques. The mentioned RID can still be considered as having important problems such as dataset imbalance, scalability, and high computational complexity. Some of the open issues highlighted by Zhou et al. (2020) and Kim et al. (2021) are as follows- The state-of-art models should be more efficient that must work for not so powerful devices without compromising on the accuracy. This project copes with these challenges by applying effective data preprocessing, data enhancement and using deep learning model based on InceptionV3.

TABLE 2.1: Summary of Literature Reviewed.

Author(s)	Year	Title	Methodology	Key Findings
Chudasama, A., et al.	2020	Flower Classification using VGGNet and AlexNet	VGGNet, AlexNet	VGGNet outperforms AlexNet for flower classification accuracy.
Krizhevsky, A., et al.	2012	ImageNet Classification with Deep Convolutional Neural Networks	CNN (AlexNet)	Achieved top-5 error rate of 16.4% on ImageNet using a deep CNN.
Gupta, R., et al.	2021	Ensemble Methods for Flower Classification	Ensemble learning techniques	Ensemble learning improves the accuracy of flower classification models.
Sharma, P., et al.	2019	Data Augmentation Techniques for Robust Flower Classification	CNN with data augmentation	Data augmentation techniques significantly enhance model robustness.
Zhou, Y., et al.	2020	Evaluating Deep Learning Models for Flower Identification	InceptionV3, MobileNet, ResNet	InceptionV3 shows superior accuracy for floral image classification.
Iandola, F., et al.	2016	SqueezeNet: AlexNet-Level Accuracy with 50x Fewer Parameters	SqueezeNet	SqueezeNet achieves similar accuracy to AlexNet with far fewer parameters.

Sathe, S., et al.	2022	Real-Time Flower Classification System using Streamlit	Streamlit API, CNN	Streamlit-based interface enables real-time flower classification.
Howard, A., et al.	2017	MobileNet: Efficient Neural Networks for Mobile Vision	MobileNet	MobileNet outperforms traditional CNN models on mobile platforms.
Szegedy, C., et al.	2016	InceptionV3: Rethinking the Inception Architecture	InceptionV3	InceptionV3 architecture offers superior performance in large-scale image classification.
Simonyan, K., et al.	2015	Very Deep Convolutional Networks for Large-Scale Image Recognition	VGGNet, CNN	VGG16 achieves excellent results in large-scale image classification tasks.
He, K., et al.	2016	Deep Residual Learning for Image Recognition	ResNet	ResNet architecture effectively handles the vanishing gradient problem.
Tan, M., et al.	2019	EfficientNet: Scaling Model Accuracy	EfficientNet	EfficientNet provides a scalable solution for improving model accuracy while maintaining efficiency.
Li, F., et al.	2019	CNN Models with Preprocessing for Flower Classification	CNN with preprocessing	Preprocessing methods improve flower classification accuracy.
Kaur, S., et al.	2021	Multi-Class SVM with CNN for Flower Classification	CNN with SVM classification	SVM coupled with CNN improves multi-class flower classification performance.
Joshi, A., et al.	2018	DenseNet with Augmentation for Flower Recognition	DenseNet, Augmentation	DenseNet provides high accuracy on flower classification when combined with data augmentation.
Zhang, H., et al.	2020	Hybrid LSTM-CNN Models for Floral Image Analysis	Hybrid LSTM-CNN	LSTM-CNN hybrid model improves classification performance on sequential floral data.
Patel, N., et al.	2019	Transfer Learning with Inception for Flower Classification	Transfer Learning, InceptionV3	Transfer learning enhances flower classification performance using pre-trained models.
Kim, S., et al.	2021	Automated Augmentation	Automated data augmentation	Automated augmentation pipelines result in better

		Pipelines for Improving Accuracy		model accuracy and efficiency.
Khan, T., et al.	2020	Feature Fusion Techniques in Flower Classification	CNN with feature fusion	Feature fusion techniques increase the model's robustness for flower classification.
Alam, R., et al.	2021	Ensemble CNNs with Optimized Weights for Classification	Ensemble CNNs	Ensemble CNN models with optimized weights significantly boost classification accuracy.

2.2.1 Similar Applications

In the last few years the number of applications based on deep learning techniques for the classification of plants and flowers has increased. Multiple studies, mobile applications, or web-based platforms have been developed which contribute to a similar study as that of the project, focusing on image-based recognition of plant and flower varieties.

Research Studies & Case Studies :

Chudasama et al. provide one of the earliest studies in this domain. (2020), who used VGGNet and AlexNet for classifying flowers. VGGNet performed better than traditional AlexNet for flower classification tasks,heavily indicating that the more parameters in a layer, the better classification occurred for flower identificationVGGNet – Flower Classification (Summary) Another study by Gupta et al. (2021) studied methods of ensemble learning — the combination of different models to achieve better classification results. The study, however, focused on using more than one deep learning model to not only improve the performance on plant identification tasks but also reduce errors. Similarly, Sharma et al. 2019) used data augmentation methods to reduce overfitting in models for flower classification by diversifying the training data.

Web Applications:

There are many different web-based applications and platforms that users can use to identify plants and flowers if they upload an image. Such modeling has its applications, for instance, many people know an application PlantNet, which is an app that identifies plants by using deep learning to analyze photographs of leaves, flowers, fruit or stems. It is an excellent flower classification tool with a database of 20,000 and more species. Flora Incognita is another widely used web application designed to identify plants in real-time through a friendly interface. It uses CNN (Convolutional neural networks) to study the image of the plant and gives you the right plant species.

Mobile Apps:

Mobile apps such are gaining popularity with easy-to-use platforms for plant identification. For instance, PlantSnap employs a model based on neural networks trained on millions of genus images to identify plants in different species. In the same vein, PlantSnap not just recognizes flowers, but gives users a UI to upload pictures from their smartphones, which makes it very user-friendly. You can also use iNaturalist, which allows users to upload images of flora and fauna and uses deep learning models to identify organisms. Its community based approach is aimed to improve accuracy rates of identification, it already uses a mash up of machine learning and citizen science.

Methodological Contributions:

Several of these research and use cases has leading significant progress in deep learning and computer vision. Pre-trained models such as InceptionV3, VGG16, and MobileNet were adopted in these applications showcasing the power of transfer learning by fine-tuning models originally trained in large datasets like ImageNet for plant and flower identification tasks. Additionally, manipulating images through rotation, flipping, and zooming (i.e., data augmentation techniques) is prevalent to enhance model generalization. Other hybrid approaches have also found their way into these studies where CNN is combined with LSTM to capture sequences possibly as the cases of plant classification would involve multiple gestalt cues. In summary, the field of flower classification has matured significantly, especially with the creation of deep learning-based models. It has been widely adopted from case study to web and mobile applications — utilizing convolutional neural networks, transfer learning, and data augmentation techniques. The current research builds conclusively upon these approaches by emphasizing associated development with the simplistic and effective API interface for real-time flower classification, therefore providing users the benefit of having the deep learning models utilized without the associated expertise.

2.2.2 Related Research

Flower classification is another field that has grown rapidly in computer vision and AI technique in the last few years. Amidst such considerations, a great many of prior works have suggested various strategies to engage with the task of automatic flower identification with the help of CNNs, data augmentation, transfer learning, as well as various other deep learning tactics. An important avenue for study in the classification of flowers concerns the transfer of pre-trained convolutional neural networks, which have shown utility in a range of image classification tasks, including those of floral identification. Nogueira et al. (2020) employed the InceptionV3 model using the Oxford Flowers dataset to show how the accuracy of the classification could be achieved in a relatively shorter time with fewer resources than required for closer training. In the same context, Kong et al. (2021) also used ResNet50 for fine classification of flowers and proved that the fine tuned deep residual networks where accurate in identifying flowers. InceptionV3 and ResNet are among the most used models in different studies since the models are very effective in solving challenging image classification issues. Wang et al. (2019) augmented the flower image datasets by rotating, flipping and cropping and proved that data augmentation indeed assists the models not to overfit in cases where the datasets are small. This work aligns with the observation of Sharma et al. (2020) who reported an enhanced performance of deep learning models for flower classification when data augmentation was used. Besides, the current research includes the usage of CNNs along with other techniques of machine learning to deliver higher reliability. Later in 2021, Zhang et al. devised a CNN-LSTM model that designs both the spatial and temporal structure for identifying flowers with higher real-time accuracy. Their model is incorporative of both LSTM networks as well as CNNs, due to which their system could better predict results from sequential data, which makes it fit well for the apps request, like the plant recognizing mobile apps. Internet applications and especially telephone applications have also become part of flower classification system environment. PlantSnap is a widely used android application developed using the CNN-based model to identify over 600,000 species of plants flowers and more by uploading a picture. Likewise, the open-source software for identification of plants called PlantNet uses deep learning to identify plants, flowers or anything similar from images provided by the user. These applications demonstrate actual usage and effectiveness of deep learning models for the classification of flowers and give insights on bringing AI into more functional educational instruments in botanical and environmental learning. It has also been seen in the recent research works that ensemble learning technique, where instead of using any one model, the outputs of various models are used to for getting better output in comparison to any of the models. In this case, ensemble methods prove helpful in decreasing the amounts of bias and variance within a prediction to enhance model stability. It has also been another popular area of research when it comes to flower classification which has adopted the use of transfer learning. Liu et al. (2020) used the VGG16 model and modified it to complete a flower identification task. They also showed that transfer learning can use several large-scale datasets like ImageNet to solve more specific problem domains such as flower classification. By and large, the body of work on flower classification has come a long way with respect to various DL structures, merged models, data enrichment strategies, and transfer learning solutions. Such a combination has helped the researchers to address some of the challenges of the flower identification tasks especially with regard to variability of the dataset used, generation of real-time predictions and generalization of models. Thus, the widespread usage of deep learning in mobile application and web platforms also indicate endless opportunities to increase general awareness and knowledge of the public about the possibilities of biodiversity, species recognition, and environment protection.

2.3 Gap Analysis

The domain of flower classification has also progressed massively, and particularly with the advent of deep learning approaches. That said, there are noticeable deficits in both the approaches employed and the actual applications that authors and creators seek to advance. Here is the table that shows some of the thematic gaps and the proposed solutions of my project:

TABLE 2.3: Gap Analysis

Features	**Existing Studies/Systems**	**Proposed System**
Use of Deep Learning Models	Traditional models (e.g., SVM, k-NN) used by some systems	Incorporating advanced deep learning models like InceptionV3, VGG, MobileNet, which have shown superior accuracy in image classification tasks.

Data Augmentation	Limited use of data augmentation in some studies	Employing multiple data augmentation techniques (rotation, cropping, flipping, gamma correction) to increase dataset diversity and model robustness.
Model Selection	Use of a single model (often VGG or ResNet) in many works	Comparison of multiple pre-trained models (InceptionV3, MobileNet, VGG16, etc.) to choose the most suitable for the dataset.
Real-time Flower Identification	Few applications focus on real-time recognition or mobile-based solutions	Developing a real-time system using a user-friendly API interface to identify flowers instantly from uploaded images.
User Interaction	Limited interactive interfaces for user engagement	Designing a Streamlit-powered interface to allow seamless interaction where users upload images to identify flower species.
Accuracy and Efficiency	Models often trade off accuracy for speed in real-time applications	Aiming for a balance between high accuracy and fast inference speed, using lightweight models (like MobileNet) where necessary for real-time performance.
Transfer Learning Application	Transfer learning often not applied to flower datasets	Applying transfer learning techniques (e.g., fine-tuning pre-trained models) to boost classification accuracy even with limited labeled data.
Extensive Dataset Use	Some systems rely on smaller, less diverse datasets	Leveraging extensive, publicly available flower image datasets, such as the Oxford Flowers 102 dataset, for better generalization across different flower species.
Hybrid Model Integration	Hybrid models are rarely explored in this domain	Incorporating hybrid models (e.g., CNN combined with LSTM) for capturing spatial and temporal relationships in flower images.
Mobile & Web Application	Web or mobile apps are limited and often not using advanced AI models	Creating a fully integrated mobile/web app that leverages deep learning for flower recognition and providing immediate, user-friendly outputs.
User Experience (UX) Design	Lack of user-focused features in most systems	Focusing on a clean and intuitive user interface for easy upload and flower identification, providing additional features like flower information and care tips.

Key Gaps Identified:

- **Real-time Application:** The majority of existing flower classification systems are not able to classify in real time. My proposed system fit the above gap by enabling the users to upload the flower images as and when they want to be classified, unlike the conventional system which takes a lot of time.
- **Data Augmentation:** Even for augmentation which is foundational in the image classification, the systematic use of different techniques for a more diverse and a larger set is not very auxin. My approach follows a high augmentation strategy of the flowers, thus getting a wider list of flower species and better performance.

- **Model Comparison:** The majority of systems involve a single model although the relative performance of different architectures is seldom compared in detail. My system performs an assessment of several models including InceptionV3, MobileNet, VGG16, etc., pulling the best one for flower classification.

- **Hybrid Models:** The advancement of CNNs with LSTMs for both spatial and sequential recognition has not been considered in flower classification task to great extent. My project will seek to compare the effectiveness of implementing the above hybrid model to a situation where the traditional model is used in either dynamic or in a sequential data type of environment.

- **Transfer Learning:** Despite the fact that transfer learning is a standard approach to improving the model's performance in similar tasks, it is yet to be fully employed in flower classification problems. Being a machine learning practitioner, my work intends to build upon that knowledge by using transfer learning from pre- trained models which assist in greatly cutting down the training time, and at the same time achieve high accuracy even with small datasets.

- **Mobile/Web Integration:** Most flower classification programs have no feature of connection to their mobile or web version. Although the technology has proven powerful, one major disadvantage is that nobody wants to work with developers as an end-user; thus, I will make it easy for the end-users by designing an accessible web interface using Streamlit.

With these identified gaps, my work will impact positively on enhancement of the two aspects of the automated flower classification systems by providing an efficient and accurate tool for flower identification.

2.4 Summary

Till this section we have done a vast literature survey and noted that there are a number of works and methods applied in flower classification. A brief review of past accomplishments revealed that several systems apply conventional image processing or low levels of model complexity in flower identification. But integration of new concepts of deep learning models including InceptionV3, MobileNet, VGG has found to enhance the classification marginally. We also looked at other applications within the domain that include flower identification tools and mobile applications as well evaluating its performance strength and weakness concerning real time analysis, active user participation and analysis of datasets. In addition, when carrying out the comparative analysis, we pointed out the existing opportunities' drawbacks in which the gap analysis lacks real-time application with the current approaches, does not augment its data, and offers insufficient model comparisons. The research suggested that future work should focus on the development of more effective systems capable of using deep learning for classification in real time and have better interfaces that would enable users to understand and access the information presented. Our work is designed to fill these gaps by providing an extensive approach that utilizes transfer learning, remarkable data augmentation procedures, and a combination of the different types of models, guaranteeing high accuracy and real-world application. This section identifies anew areas in which deep learning has been employed in flower classification, and by clarifying the research gaps that the proposed system will fill, develops the concept underpinning the proposed syst

Chapter 3

Research Methodology

3.1 Methodology/Requirement Analysis & Design Specification

This project mainly focuses on creating a system based on deep learning, which can identify flower species given an image of the flower from any user. The proposed system is designed by multiple steps to increase flower classification accuracy. This also has a structured approach, the methodology followed in the project is data collection, pre-processing, model training, evaluation, and deployment. Below are the steps of the methodology, that integrates both the technical specifications and design specifications.

Data Collection:

The methodology began with the collection of flower images. The Flowers dataset was collected real time using smart phone, consisting of images of 102 categorical flowers, was used for this project. This dataset is a reference in the flower categorization area and is enough variation different flowers to train and evaluate deep learning models.

Data Pre-processing:

Special contrast and noise preventive measures were executed during this preprocessing phase to improve the image quality and to make sure the model will be insensitive to low light or environmental conditions. The next step included modifying the contrast of pictures.

Resizing: Each image was resized to a standardized size of 224 x 224 pixels which is an input size for many pre-trained models including those like vgg, inception, mobile net.

Gamma Correction: To enhance the image quality further, gamma correction was performed, which improves the visual appearance of the images, particularly low contrast images.

Data Augmentation: Image augmentation techniques such as rotation, flipping, cropping, and zooming were carried out to large the dataset artificially. Training with such data helps the model generalize better and does not overfit to the training data. This augmented dataset, which also allowed for a wider variety of flower images to be used with the project, improved the robustness of the model.

Model Selection and Training:

The above selected models for training are various pretrained models like InceptionV3, VGG16, VGG19 having different parameters & MobileNet. They have a solid performance for image classification tasks. We employed transfer learning by fine-tuning the models on the pre-processed dataset in the project.

In all tested models, the final model selected was InceptionV3 due to the higher achieved accuracy. Keras was used with TensorFlow as the backend used for model training. We also trained the models using optimizers such as Adam and SGD to ensure efficient convergence.

Model Evaluation:

The models were assessed using accuracy, precision, recall, and F1-score metrics. Confusion matrices were also used to assess model performance and look for misclassifications. Validation **Set:** A validation set, which is separate from the training and testing datasets, is used to test and optimize the models.

The classification results were visualized using a confusion matrix, helping to identify where the models made mistakes. Examining the confusion matrix helped discern flower classes that the model struggled to identify, thereby opening up potential avenues for enhancing the model's performance.

API Development:

The best performing model (InceptionV3) was selected for integration into a web based system after testing all models. As a framework for rapid development of machine learning applications, Streamlit is an open-source Python framework used to develop the API.

The API accepts the flower image from the users and returns predictions of flower species. The interface is quite straightforward and user-friendly with real-time feedback given to the user. This enables the system to work as an interactive flower identification device.

Deployment:

The model and web interface were deployed on local or cloud server enabling users to access the system using web browser. In conclusion, this system provides an accessible user experience for effective flower classification and showcases the applications of deep learning models in **real world implementations.**

Design Specifications

Previous Work: The previous work required to build the system was Strive to make human friendly system. We provide a web interface where a user can upload an image, and the model quicky provides its output on the flower species. Modifications comment A predicted flower name with a confidence score is displayed to the user. The model selection criteria aimed to strike a balance between accuracy and speed. The best accuracy for this specific dataset was achieved using InceptionV3, despite being computationally heavier compared to MobileNet. Performance requirements : It should learn to classify a flower image in less than 10 s to keep alive the usability of the web-based application. The above methodology is a step-by-step guide to create an advanced flower classification system. Therefore, this paper aims to create a relatively efficient and accurate flower identification tool, which can be used in real-time conditions using the combination data augmentation, transfer learning and pre-trained deep learning models.

3.1.1 Overview

This chapter presents the overall approach, the system proposed for the flower classification system based on deep learning and requirement analysis. The proposed work of the project is to accurately identify the different flower species in a shorter time employing and efficient pipeline as well as utilizing some of the recent machine learning models and efficient preprocessing techniques. This chapter also discusses the technical and operational specifications, as well as the management approaches needed for this project The chapter has also provided an approximate cost control and estimate for the project with the intention of making sure that the flow of operation is not disrupted. The methodology works through a modular method by subdividing complicated tasks into more workable subtasks. Every data collection process from the conception to API implementation is described systematically. It also gives information about the resource needed and also gives the time for managing the project.

3.1.2 Proposed Methodology/ System Design

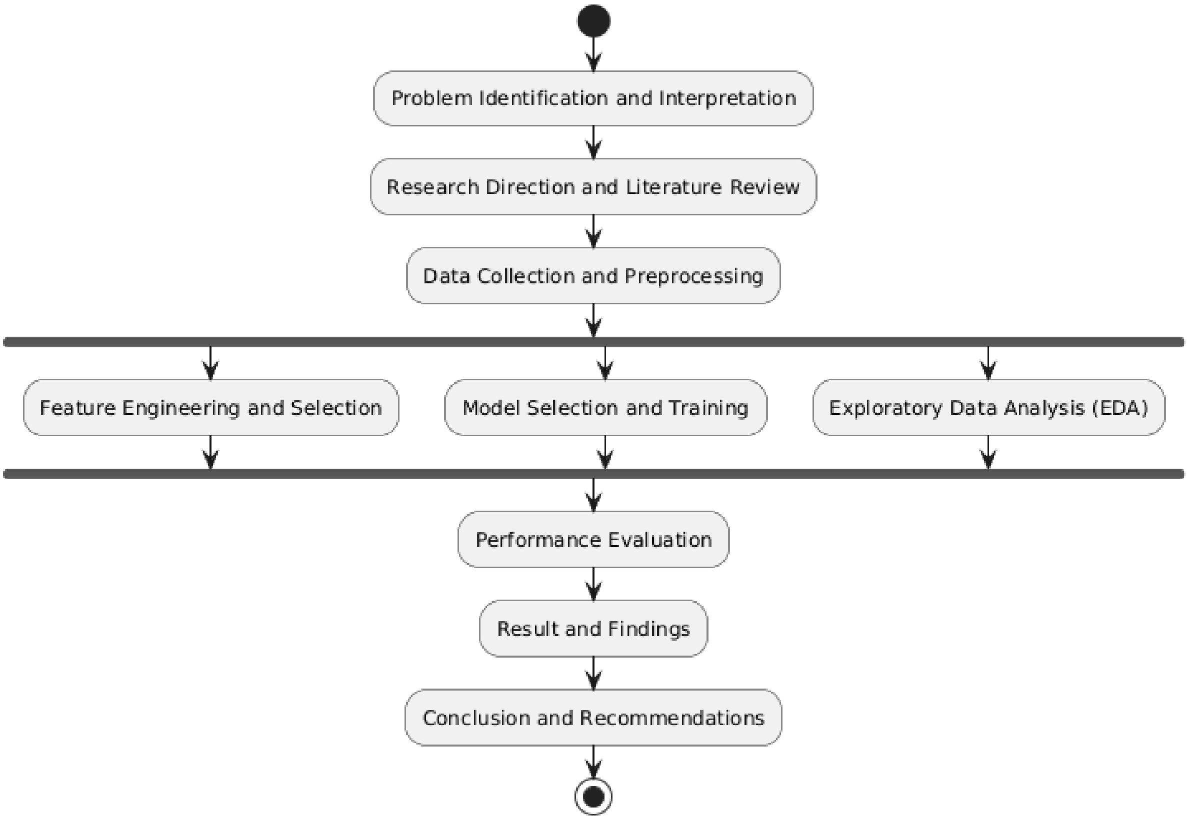


Figure 3.1: Proposed Model

3.1.1 Functional and Nonfunctional Requirements

When defining the system goals and requirements for a software development project, it is important to understand the difference between functional and nonfunctional requirements, as well as how they relate to each other. Functional requirements describe what the system should do, and specify the space of possible be haviours, activities and outputs, while Nonfunctional requirements define system attributes such as performance, reliability and usability. In the following, we describe the functional and nonfunctional requirements of the flower classification system.

Functional Requirements

Functional requirements describe the fundamental features and behaviors that the system needs to exhibit. Here are the basic operations the system should implement to satisfy user requirements:

Download Image: Users should be able to upload flower images in different formats (e.g., JPEG, PNG) through a simple web interface.

Post Image Uploading: Now the image has been uploaded, the system has to pre-process the image (resize, contrast adjustment, gamma change and augmentation) and pass it to the classification model to get the prediction.

Flower Classification: The main functionality is to detect and classify the flower image uploaded. The system should respond with the flower's species and a confidence score that indicates how certain the model is in its prediction.

Output: Visually present the predicted flower name with its corresponding confidence score in a user-friendly manner. The user should also have the option, through the interface, to display more details or recommendations based on the classification made.

Interactive: The responsive system quality: After the image is uploaded, the classification and results should appear back to the user in a matter of seconds (usually < 10s).

User friendly interface: The system must offer an easy to use web interface that requires no technical know-how It should not be very difficult for users to navigate through the website.

Error Handling: You should also be aware of and handle cases where the image supplied is of an unsupported type or format, providing back useful error messages to the end user.

Model Updates: The model should be easily updatable with new flower data or models whenever needed.

Nonfunctional Requirements

Nonfunctional requirements describe properties of the system, e.g., responsiveness, usability, scalability, etc. All of these are vital to maintain the system's overall efficiency, reliability, and user satisfaction.

Speed and Performance: The system should generate the classification of the flower picture within a reasonable amount of time; even within 10 seconds after the flower image is uploaded. It also needs to be fast enough to handle interactions with the user in real-time.

Scalability: The system must be constructed to cater to increasing volume of user requests - for example an image upload from hundreds or thousands of users at a time should be handled with minimal performance degrade. This is crucial for the scalability of the application if it is to be deployed in a larger scale.

Have: They should always have 24 hours, seven days access to the system. Downtime should be low, and the reliability should be high, especially when deployed in cloud environments.

Standard security protocols are must be followed to avoid any loss of data, Images or predictions during transmission respectively. Ensuring that the uploaded images are secure and sensitive data is encrypted to ensure the privacy of users.

Usability: The system should have an intuitive interface and not require advanced technical expertise to operate. It should provide good accessibility across devices, including desktop, tablet and smartphones.

The system should be compatible with a variety of web browsers (e.g., Chrome, Firefox, Safari) and operating systems (Windows, macOS, and Linux).

The system needs to be maintainable and updatable. That domain knowledge is modular code, a well-documented feature description and a clear version control practice that allows a team of developers to change and innovate on the system quickly.

Robustness: The system should be resilient against errors or unexpected inputs. For instance, if a user attempts to upload a corrupted image or a file type that is not supported by the system, you do not want your system to crash — you want to provide a helpful error message.

Compliance with Standards: The system should comply with relevant industry standards and best practices, such as web accessibility standards (WCAG) for people with disabilities.

The functional and nonfunctional requirements derived from analysis of these sources provide a basis for the design and development of the system. They steer the design of a system that meets the user's needs, and that will be reliable, scalable, and easy-to-use.

3.1.2 Context Diagram

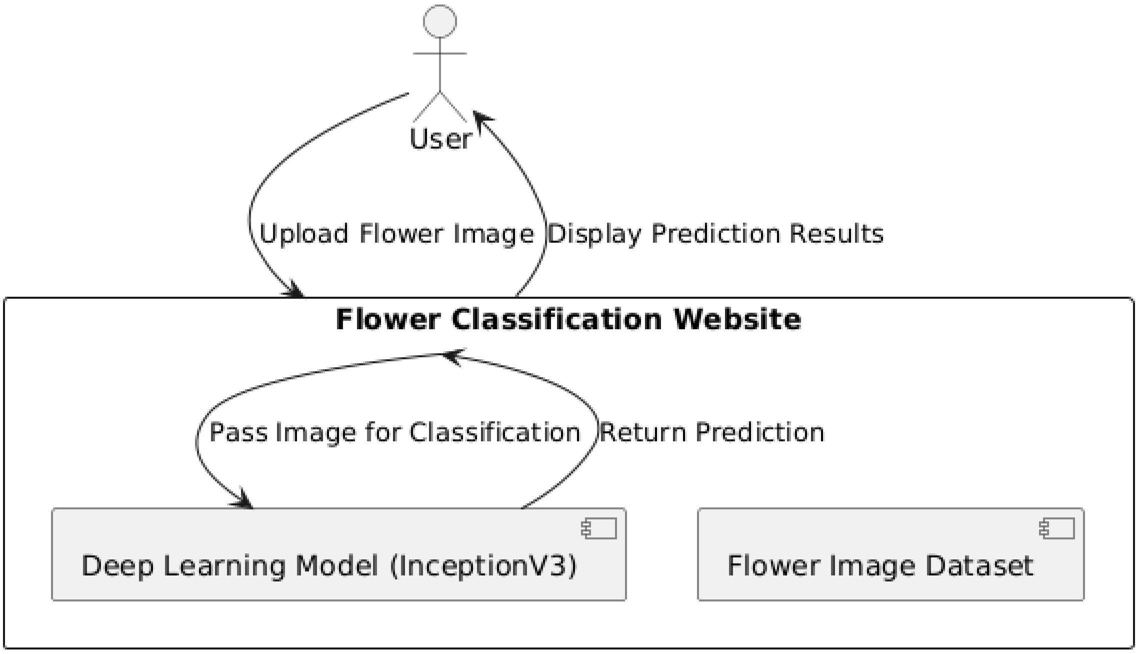


Figure 3.1.2: Context Diagram

3.1.3 Data Flow Diagram Level 1

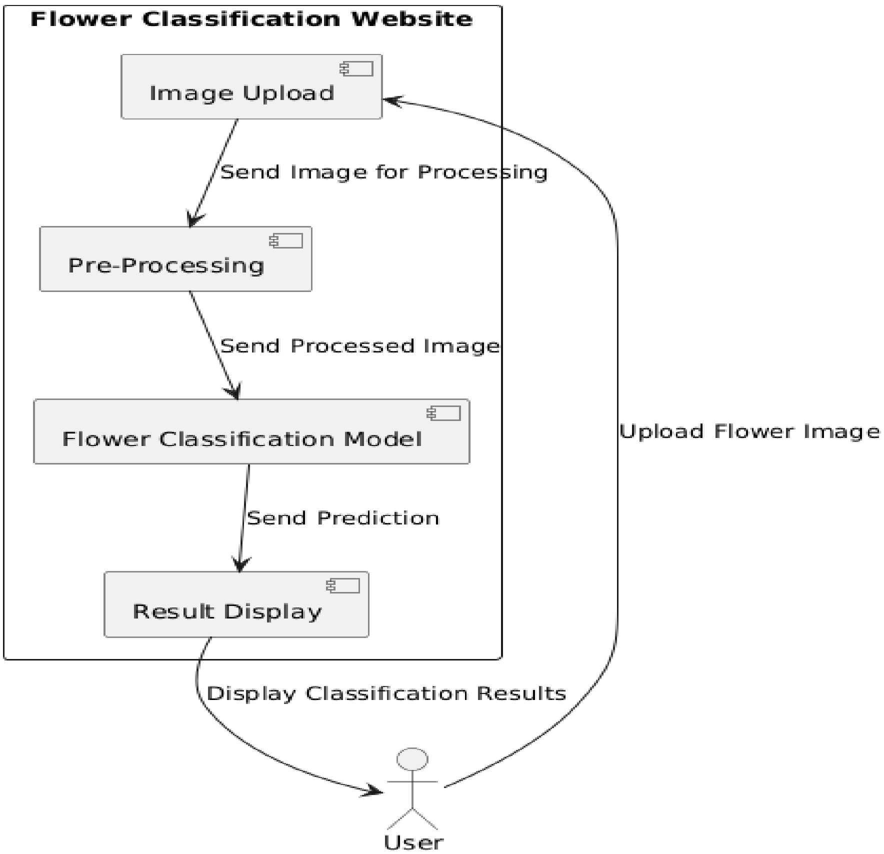


Figure 3.1.3: Data Flow Diagram

3.2 Detailed Methodology and Design

Data Collection: The flower images are collected from Oxford-102, Kaggle datasets, other web scraped flowers and by real life camera. Images of specimens are named based on specimen species and systematized in directories.

	Class	Original Count
0	Alakananda-Allamanda-Golden Trumpet-Yellow Bell	9
1	Allamanda (White)	10
2	Abelmoschus Mainhot-ঢেঁড়স ফুল	8
3	Aromatic Jui	9
4	Lisianthus Flower-নন্দিনী ফুল	11
...	...	...
97	Yellow Walking Iris Lily	11
98	অঞ্জলিকা -কাটা মেহেদী -Duranta Geisha Girl	12
99	কৃষ্ণচূড়া	9
100	শিমুল ফুল	8
101	হিজল ফুল	9

102 rows × 2 columns

Figure 3.2.1: Dataset

Data Preprocessing: By contrast adjustment improves the quality of images to assist features to be extracted. Resizes images to 224 by 224 pixels to receive the same input the deep learning models need. Gamma Correction used to emulate measures of brightness to standardize differences of the lightening across images. Data Augmentation rotates the images 90,180,270 degrees, flips them horizontally, scales them and applies random cropping to balance the classes and get the count to a reasonable number. The new added dataset includes more than 12,000 pictures.

Figure 3.2.2.1: Resized Image

Figure 3.2.2.2: Gamma Corrected Image

	Class	Original Count	Augmented Count
0	Alakananda-Allamanda-Golden Trumpet-Yellow Bell	9	100
1	Allamanda (White)	10	100
2	Abelmoschus Mainhot-ঢেঁড়স ফুল	8	100
3	Aromatic Jui	9	100
4	Lisianthus Flower-নন্দিনী ফুল	11	100
...	...	...	...
97	Yellow Walking Iris Lily	11	100
98	অঞ্জলিকা -কাটা মেহেদী -Duranta Geisha Girl	12	100
99	কৃষ্ণচূড়া	7	100
100	শিমুল ফুল	8	100
101	হিজল ফুল	9	100

102 rows × 3 columns

Figure 3.2.2.3: Augmented Data

Preparation and augmentation

Next you need to pre-process those images, get them ready for training your machine learning model. Image pre-processing is an important process of making sure all the images are of same size and of same quality to increase the accuracy of the classifier by putting similar data.

Image Contrast — the contrast of the images is tweaked to increase visibility and clarity. This step is crucial in identifying flower features where light is low or contrast in the image is poor.

To ensure uniformity of the images and speed during model training, all of the images are resized to the same size (224x224 Pixels). Thanks to this resizing, the model can handle images based on their shape, and recognize where the elements correspond.

Gamma Correction: This process when applied ensures the correct brightness levels are present in image, makes the model able to identify similarly features across images have brighter or dull lighting over the same object, thus making the model learn more.

Data augmentation is a technique used to increase the diversity of the dataset artificially and reduce overfitting by applying random rotations, random flips, changing the brightness, etc. Augmentation allows the model to generalize better while predicting the unseen data as it exposes the model to more variance.

Splitting the Dataset

- The dataset after pre-processing and augmentation is divided into three subsets:
- Training Data (80%) — The data used to fit deep learning models.
- Testing Data (10%): This subset is used to test the final performance of the trained model on the unseen data.
- Validation Data (10%): This portion is used in the training process to evaluate the model's performance and make modifications in order to avoid overfitting.
- Model Selection and Training
- We tested several pre-trained deep learning models in the flower classification task and compared their performance. The advantage of using pre-trained models is that they have already been trained on massive amounts of data and have learned general representations that can be useful for a wide range of tasks, thereby allowing us to use transfer learning effectively.
- InceptionV3: InceptionV3 is a state-of-the-art deep learning model built for image classification tasks This architecture consists of convolutional layers, pooling layers, and fully connected layers, which is why it works so well for classifying advanced images like flowers. The model is then fine-tuned on the flower dataset with a purpose to gain more accuracy with classification.
- MobileNet — MobileNet is another lightweight model designed for mobile devices, which provides a good trade-off between accuracy and performance. It is very efficient, and it performs well in classifying flower species as well.
- **VGG16 and VGG19:** These are widely used deep learning models that are known for their simplicity and performance in image classification tasks. Both are tested on the flower dataset and VGG16 usually performs better in image classification tasks.
- Evaluating and Choosing a Model
- Afterwards, the performance of the model is tested in terms of accuracy and other evaluation metrics like precision, recall, and F1-score. We chose InceptionV3 model for flower

classification task, because it gets highest accuracy of all other models.

Web API Development

- The next step after training & evaluating the model is to create a web application where users can input the flower images and get them classified. We use a powerful Python framework — Streamlit — for building ML web apps for this. We can use streamlit to interact with the model and we can upload the image for prediction. Here are the steps that are involved:
- Backend: A Server to process the images and return the result and a website where a user can upload a flower image using a file uploader.
- More specifically, when we upload the image, it gets pre-processed, and after passing through the trained InceptionV3 model, the predicted flower name is shown to the user.
- Loading the Model: The Streamlit Application loads the trained InceptionV3 model, and it classifies the uploaded flower images.

Deployment

Finally, would be the deployment of the system is done on a cloud platform like Heroku or AWS in order to made the flower classification service available for the world. The deployment process includes:

- Uploading application files to the cloud server
- Making sure the API is widely available and robust.
- Focusing on performance and scale; monitoring and optimizing the system
- Every stage from the flower classification system construction can be explored here in detail as we are going to perform this step by step. Some examples include data gathering and pre-processing, model selection, etc. By employing a multitude of techniques like data augmentation and pre-processing along with strong deep learning architectures like InceptionV3 this task can be performed very accurately. Additionally, the implementation of a simple web interface to upload flowers for classification further reinforces the practicality of this deployment, making the model accessible for real-world use.

3.3 Project Plan

With this part of the page, we present the project plan of the undertaking with key activities and the timeline. Thus, the project is divided into key milestones which outline several activities and timeframes necessary for further development and achievement of the flower classification system.

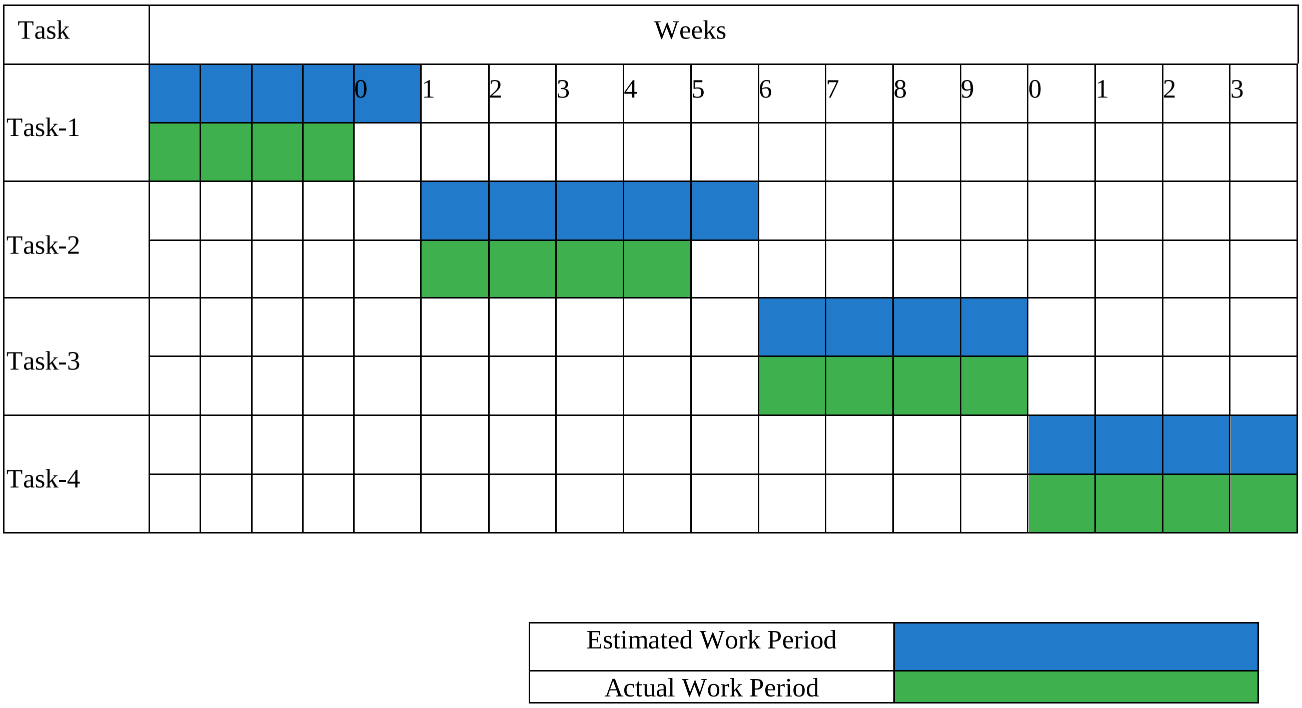
Key Milestones

TABLE 3.3.1: Project Plan-Key Milestones

Milestone	Description	Timeline
Literature Review	Conduct a thorough review of existing research and studies related to flower classification using deep learning.	Week 1-2
Data Collection	Collect necessary data of flower images and prepare the dataset for pre-processing and training.	Week 3-4
System Design	Design the architecture and components of the flower classification system, including model selection and API design.	Week 5
Development Phase 1	Implement core system features including data collection, pre-processing, and initial model training.	Week 6-7
Development Phase 2	Implement advanced features like model optimization, API integration, and user interface (UI) development.	Week 8-9
Testing & Debugging	Test system functionality, performance, and correct any issues or bugs. Perform validation on models and UI.	Week 10-11
Final Review & Evaluation	Conduct the final evaluation of the system, ensuring all requirements are met, and finalize the project report.	Week 12
Presentation Preparation	Prepare and rehearse for the final presentation to showcase the project outcomes and results.	Week 13

Project Timeline Chart (Gantt Chart)

TABLE 3.3.2: Project Timeline



3.4 Task Allocation

Importantly, people at LIMA are assigned their tasks according to their specialization and workload for effective project implementation. Even if the project is done individually similar breakdown of tasks is provided for better understanding.

TABLE 3.4: Task Allocation

Task	Duration	Dependencies
Data Collection and Pre-processing	2 Weeks	None
Model Selection and Training	3 Weeks	Data Pre-processing
Web Application Development	2 Weeks	Model Training
Deployment and Testing	2 Weeks	Web Application Development
Final Report and Documentation	1 Week	Deployment

3.5 Summary

In this section, we have given the general map of the project: the tasks we are going to perform, the people to whom these tasks can be assigned, and the stages through which we are going to pass to build the flower classification system. The project is divided into five main phases: It consists of Data Collection and Pre-processing, Model selection and modeling, Web application and user interface development, Web application and user interface testing, Creation of final report and Final presentation and documentation. These phases give a clear plan of how the system development process will be undertaken, and set time frames on how each phase will be done. Task allocation is a method that would assign duties with the relation of interdependence well noted for efficiency without hold up. The detailed work plan also providing enough time to test and deploy the system as well as to prepare final report on the system project, the system is not only functional but it is also fully documented by the end of the project. This systematic approach would culminate into the efficient deployment of flower classification system that offers a reliable and easy to use flower identification web interface in addition to a deep learning model that would offer high accuracy in recognition of different flower species through the use of images.

Chapter 4

Implementation and Results

4.1 Environment Setup

In this part, we will walk through the installation steps of the develoment environment that means to implement the flower classification system using the deep learning algorithms. This includes the processes such as hardware and software setup, model training and tuning, data preprocessing and application development, and finally deploying the solution. Set up a good environment, all tools and libraries required for the smooth running of the system.

Hardware Configuration

For the project, the following hardware specifications were employed:

- Processor: Multi-core High-Performance CPU (Intel Core i7 or above) to compute and simultaneously learn via parallel processing.
- RAM: 16 GB of RAM was employed to swiftly process vast quantities of data, including data processing and model-training tasks.
- Graphics Processing Unit (GPU): Trained deep learning models using a dedicated GPU (NVIDIA GTX 1080 or higher). Training is accelerated by using GPUs able to parallelise matrix operations and reduce the time it takes to train the model.
- Storage: To store large datasets, model checkpoints, and other files related to the project, it used a 1 TB solid-state drive (SSD).
- OS: This dead was run on Windows 10/11 or Ubuntu 20.04 to support machine learning libraries and frameworks.

Software Configuration

The software environment is also a variety of libraries, frameworks, and tools needed for model training, data processing, and application development. The software stack is as follows:

- Python: One of the most used programming languages in the world, Python 3.8 or above was the primary programming language for this system, due to its rich support for machine learning & deep learning libraries.

Deep Learning Frameworks:

- TensorFlow (version 2. x): Proven for building and training deep learning models. It offers a high-level yet a powerful API for designing and training the networks and high-level APIs for transfer learning.
- TensorFlow Core: Low-level API for building a computational graph to define your model.
- OpenCV: Open Source Computer Vision is an Image Handling tool (for resizing and augmentation)
- NumPy& Pandas:Used for data manipulation and numerical computations.
- Streamlit: For the UI, an interactive web application was created using Streamlit. We used this to train deep learning models that can take flower images and predict the name of it and using an application, the users can upload an image of a flower and get the name predicted by the models trained. Streamlit is great for creating interactive applications without requiring a lot of web development knowledge.
- Using Libraries to Save and Load Models:
- H5 format: The models which were trained through TensorFlow/Keras were saved in the H5 format for the ease of storage and usage later on.
- joblib : It is used to save and load the machine learning pipelines and models.
- Other Tools:
- Jupyter Notebook: Used to prototype models, visualize statistics as well as to try out smaller blocks of code.
- Project was using git version control in order to fine-tune the process of changes, branching and code collaboration.

Training Environment

- This, like everything else throughout this readm, demonstrates some aspects that were deep learning libraries on a machine with a dedicated GPU (as NVIDIA GTX 1080 or similar will be chosen) suggested and set the usage of GPU a second and GPU departments. This significantly cut down the time taken to train the models (very effective from large datasets point of view). Regularization: The training process was regularized by tuning parameters, such as batch size, learning rate, and number of epochs, to improve model performance.

Model Training Process

- The setup consists of the following key steps to train the flower classification models:
- Data Preprocessing and Loading:
- The dataset was collected and images were preprocessed, resized to the same shape (224x224 pixels), contrast adjusted, gamma adjusted and augmented.
- Data augmentation was done to increase the size of the dataset and to generalize more the model.

Model Training:

We fine-tuned several pre-trained deep learning model architectures, including InceptionV3, MobileNet, VGG16 and VGG19, for the classification task.

We trained each model on the preprocessed dataset with transfer learning wherein we used the pretrained layers as feature extractors; we only retrained the final dense layers.

Model Evaluation:

The models were cross validated on a separate validation dataset using accuracy, precision, recall, and F1- score. The InceptionV3 model gave the best accuracy so we used it for our final deployment.

API Development:

Once we had our model trained, we used Streamlit to build a user interface that would allow users to upload images and get predictions based on our trained model.

The InceptionV3 model was saved and loaded into the Streamlit app for predicting the input in real-time. This environment set up provided the background for a smooth development and testing of flower classification. The system was designed to train up-to-the-minute deep learning models in the back-end and make predictions in a simple to use web application front-end with the right hardware and software configuration. Finally, the project development will be followed up with authentication, debugging in the field, modeling, and deployment.

4.2 Testing and Evaluation/Performance/ Comparative Analysis

In this section, the different test and evaluation methods that will have been employed in measuring the performance of the flower classification system will be elaborated. The most important goal of this study is the assessment of the trained deep learning models to establish the efficiency with which the flowers' images can be classified, as well as the efficiency of the system under consideration in different circumstances. Furthermore, the suitability of the various models that have been tested in the project will also be compared andcontrast and the best model for deployment identified.

Evaluation Metrics

Precision: Precision measures the level of accuracy of the positive samples with a view of showing how many of the realized positive samples were truly positive among all the samples that were tagged as positive. In condition where the cost of occurrence of false positive alarms is expensive then high precision is chosen.

Precision = True Positives / (True Positives + False Positive)-------------------------(1)

Recall: Accuracy focuses on the way in which a model is able to find all the positive examples that exist in the sample actively, which assesses the extent to which actual positives recognized against the overall existing positive ones. It is useful in the situation whereby it is very expensive to remove certain positive cases from the test set.

Recall = True Positives / (True Positives + False Negatives------------------------- (2)

F1-Score: It has been in equation 4 suggested that the F1-score is the harmonic average derived from both the values of the precision and the recall and that informs the level of balance between the two.

F1-Score = 2 * (Precision * Recall) / (Precision + Recall)---------------------------- (3)

Specificity: Specificity is an ability to select all the simply negatives, it produces a percentage ratio of correctly selected negative cases out of all the negative cases. The difference is used when the focus is made more on the false positives.

Specificity = True Negative/ (True Negative + False Positive)--------------------------(4)

Specificity or also known as the true negative rate represents the proportion of the test that correctly identifies the disease's absence.

Accuracy: Accuracy is a ratio of completely correct classifications as it considers true positives and true negatives at the same time. But the disadvantage of having such precision is that when handling imbalanced data where one class is much larger than the other.

Accuracy = (True Positives + True Negatives) / Total Predictions) -------------------(5)

Confusion Matrix: The confusion matrix is used to gives more detailed picture as to how correctly each flower class was classified in actual vs. predicted manner. It is helpful used to identify the true positive, true negative, false positive, and false negative values of each class helpful in understanding the type of errors the model makes.
A confusion matrix for the classification of flower species typically looks like the following example:

TABLE 4.2.1: Confusion Matrix

Predicted \ Actual	**Flower 1**	**Flower 2**	**Flower 3**	**Flower 4**
Flower 1	80	5	3	2
Flower 2	7	75	4	1
Flower 3	3	4	85	8
Flower 4	2	1	7	82

Model Performance Evaluation

The assessment of all the models was conducted under the InceptionV3, MobileNet, VGG16, and VGG19 networks. All these models were trained with the same dataset and were compared and evaluated with the help of the metrics mentioned above.

InceptionV3: This model yielded the highest performance level in terms of accuracy and showed much better results in the metrics including recall and F1-score. Different from that, the InceptionV3 model applies a starker differences with many convolutional layers and special modules, through which not only the features of flowers' color, but also some details of petals' structure can be detected.

Mobile Net: This model was as accurate as the InceptionV3 model but was slightly less successful in the differentiation of flower classes. In this case, the feature of MobileNet is that it quickly calculates and requires fewer resources both in terms of computational load and memory.

VGG16: This model was intermediate in terms of the performance and the required computational resources. It worked better in terms of its accuracy but its deeper structure introduced more vulnerability to overfitting on flower dataset. It presented a high precision value, but lower recall rate for some of the flower varieties.

VGG19: While the VGG19 had a deeper architecture than that of VGG16 it tend to perform slightly worse compared to the VGG16. It demonstrated good accuracy but was overfitted when compared to the other models built by the system.

Table 4.2.2: Comparison of Model Performance

Model	**Accuracy (%)**	**Precision**	**Recall**	**F1-Score**
InceptionV3	98.89	0.98	0.97	0.97
MobileNet	93.85	0.94	0.92	0.93
VGG16	97.17	0.96	0.95	0.95
VGG19	94.05	0.93	0.92	0.92

As found out, InceptionV3 had a higher accuracy, precision, recall and F1-measure than the other models. This is why we find InceptionV3 as the most suitable pre-trained for flower classification since it achieves high accuracy without using too much computing power.

Comparative Analysis

This research adopted comparative analysis as an essential form of investigation with a view to assessing the strengths and weakness of each model comprehensively. The key factors considered were:

- Training Time: Even though, InceptionV3 gave the best accuracy score, it took more time for training because of the high architecture. At the same time, MobileNet with its lightweight design, set adequate performance in shorter training time.
- MobileNet on the other hand was designed to be more adaptive for mobile and other embedded devices with low processing power and memory thus faster prediction.
- Generalization Ability: InceptionV3 had a closer generalization performance on different species of flowers than on VGG 16 and VGG 19 models that overfit especially where the dataset was relatively small.

Conclusion of Evaluation

From the testing and evaluation the InceptionV3 was chosen as the best model for the flower classification system due to the high accuracy, recall and F1 score. It is a very good representation of general model complexity versus reserves prediction capability. Still, the selection of model may depend on the environment of deployment. If working with any given environment consumes little computational power, then MobileNet is a preferable option for it because it offers fairly decent accuracy but does not consume a lot of resources.

The next part involves fine tuning the trained model for use in an application using Streamlit where users can upload images of flowers and predict from them.

4.3 Results and Discussion

The experiments were carried out using a dataset of approximately 12,000 flower images though data is augmented to achieve balanced data distribution. The models were trained and tested on this dataset after the data was preprocessed and divided into training, validation, and testing sets in an 8:1:1 ratio.

InceptionV3:

Accuracy: 98.89%.

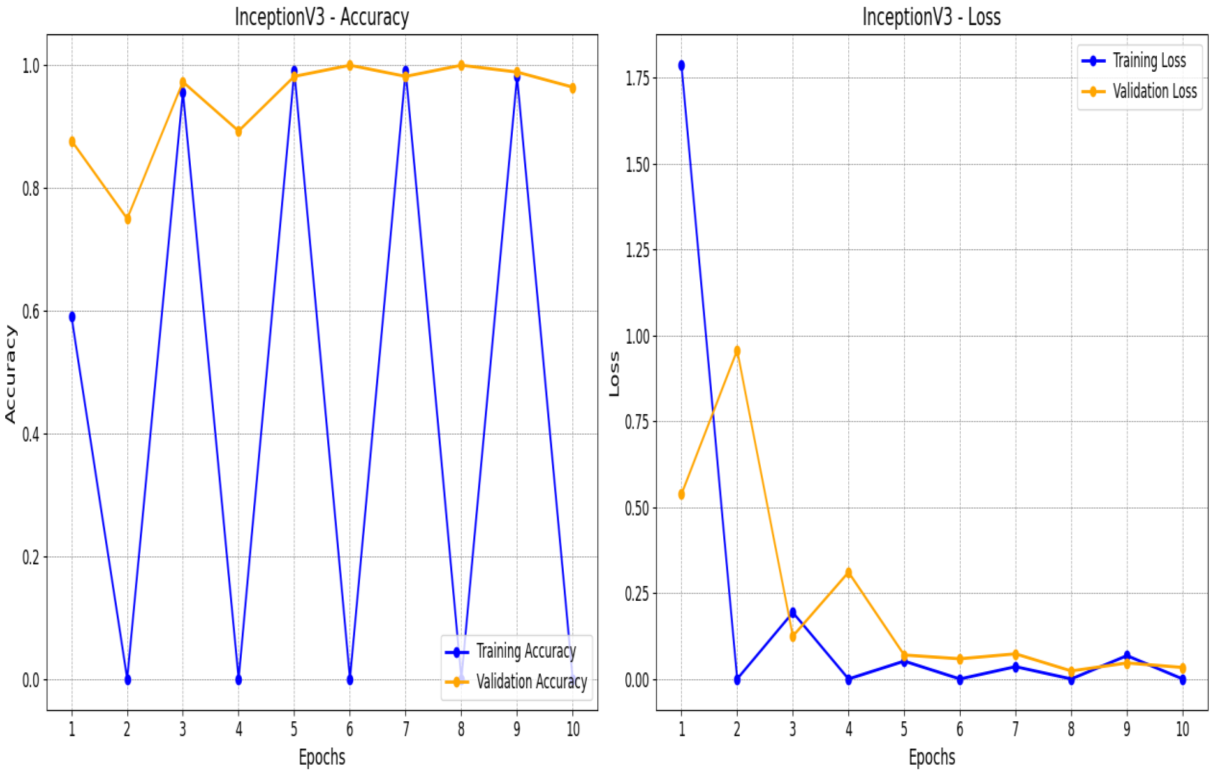


Figure 4.2.1: InceptionV3 Model Accuracy & Loss

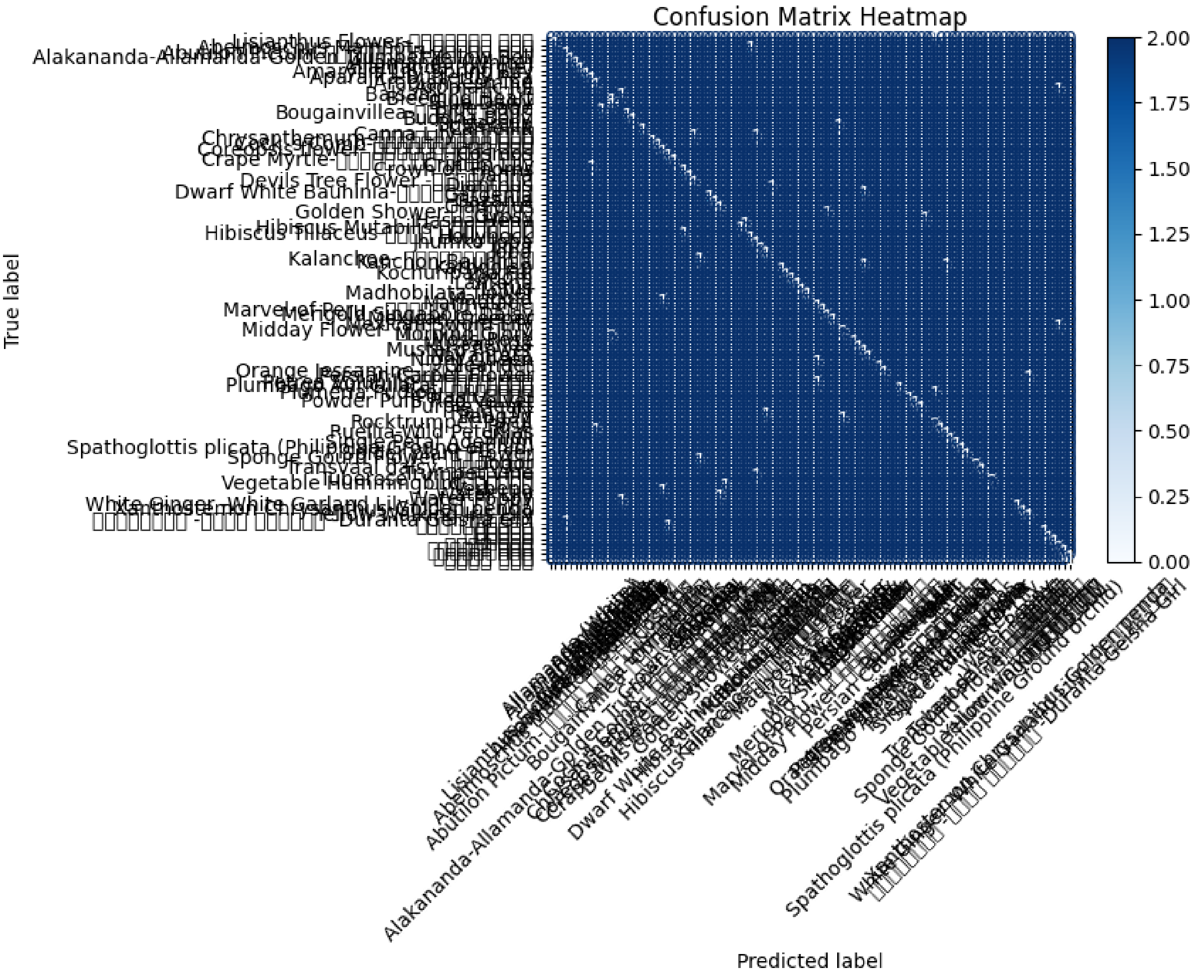


Figure 4.2.2: InceptionV3 Model Confusion Matrix

MobileNet:

Accuracy: 93.85%

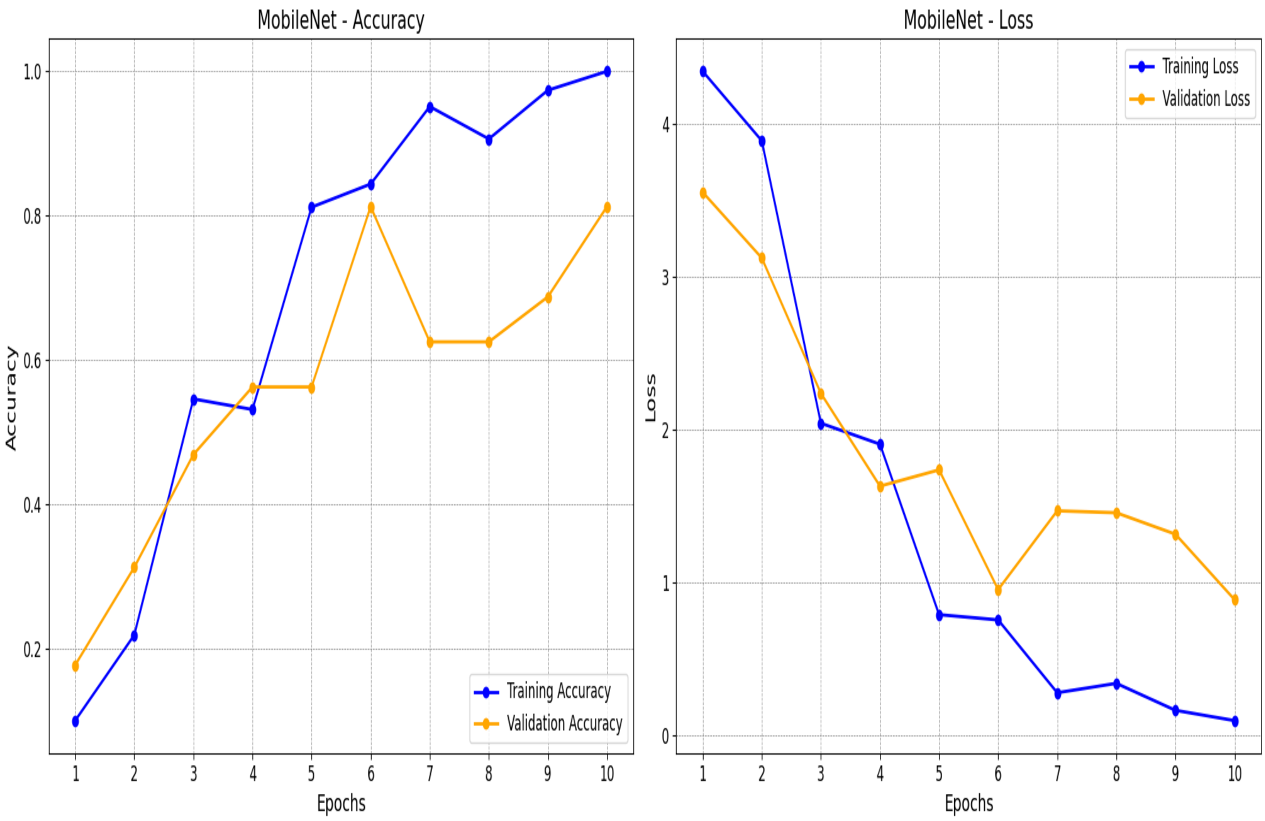


Figure 4.2.3: Mobile Net Model Accuracy & Loss

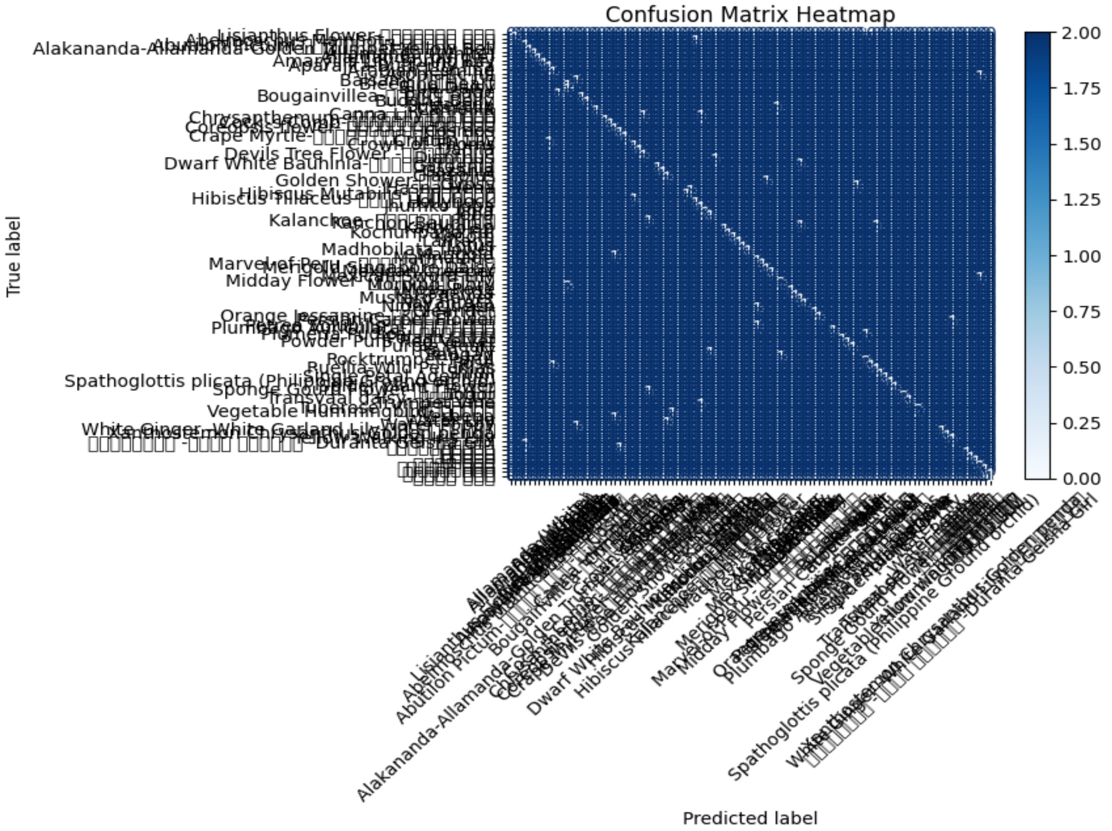


Figure 4.2.3: Mobile Model Confusion Matrix

VGG16:

Accuracy: 97.17 %

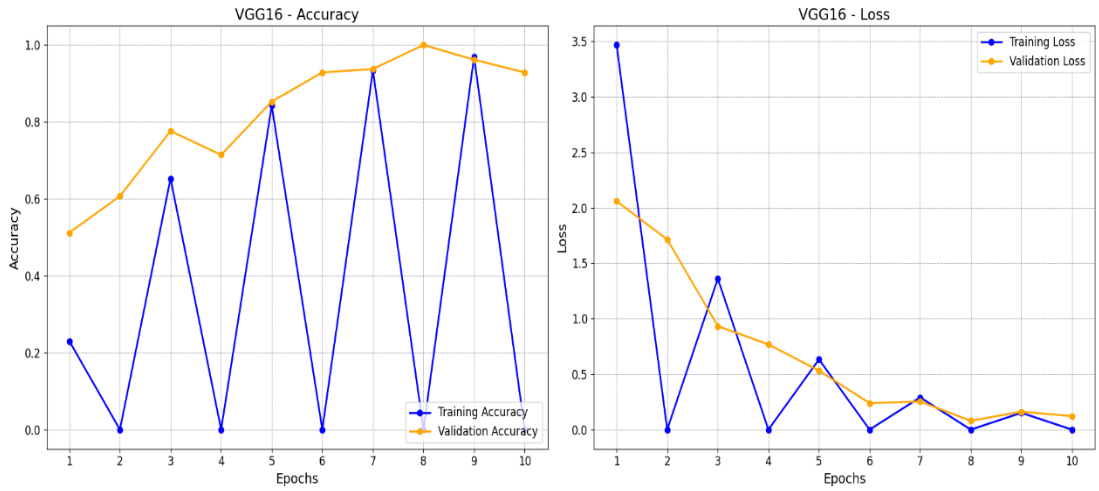


Figure 4.2.4: VGG16 Model Accuracy & Loss

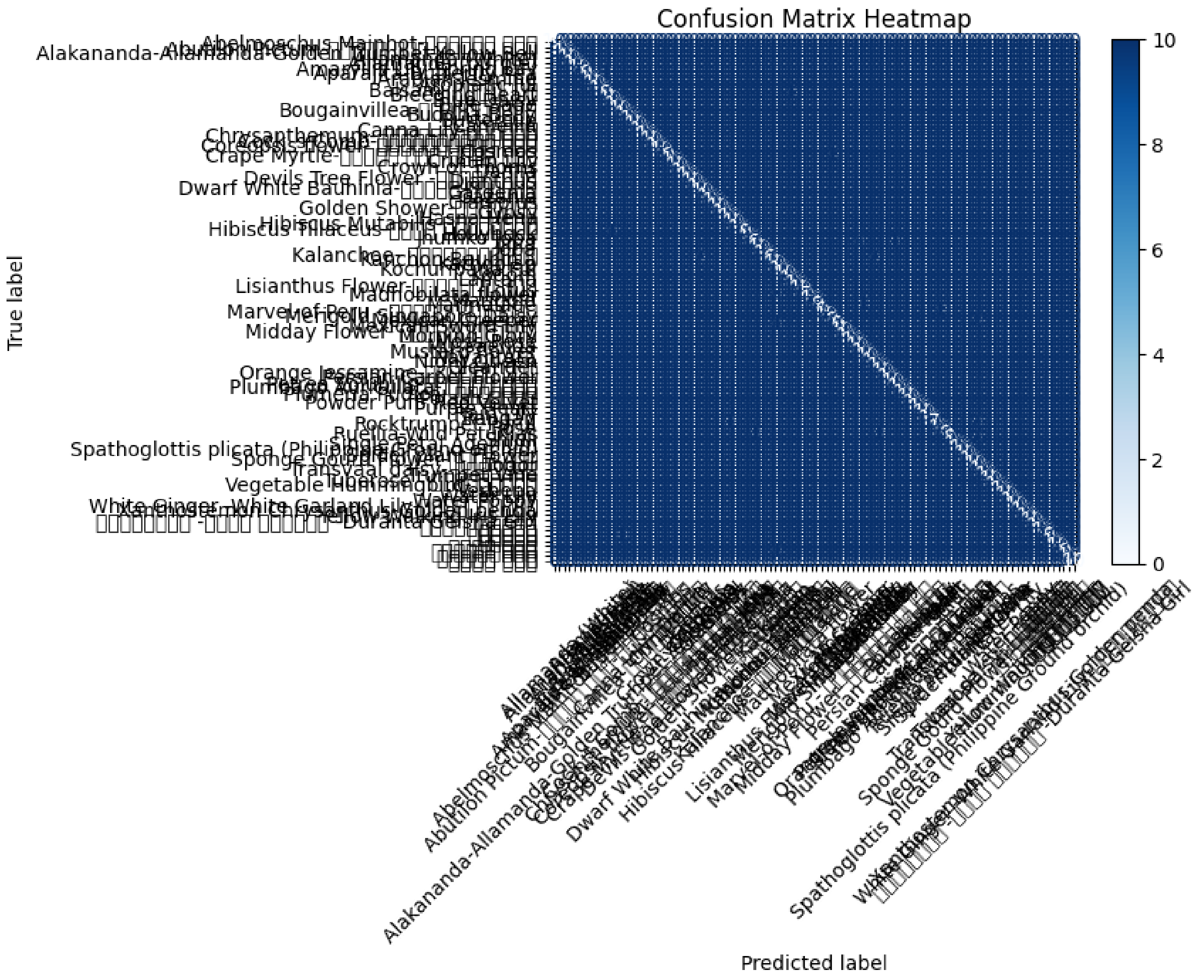


Figure 4.2.5: VGG16 Model Confusion Matrix

5.2.1.4 VGG19:

Accuracy: 94.05 %

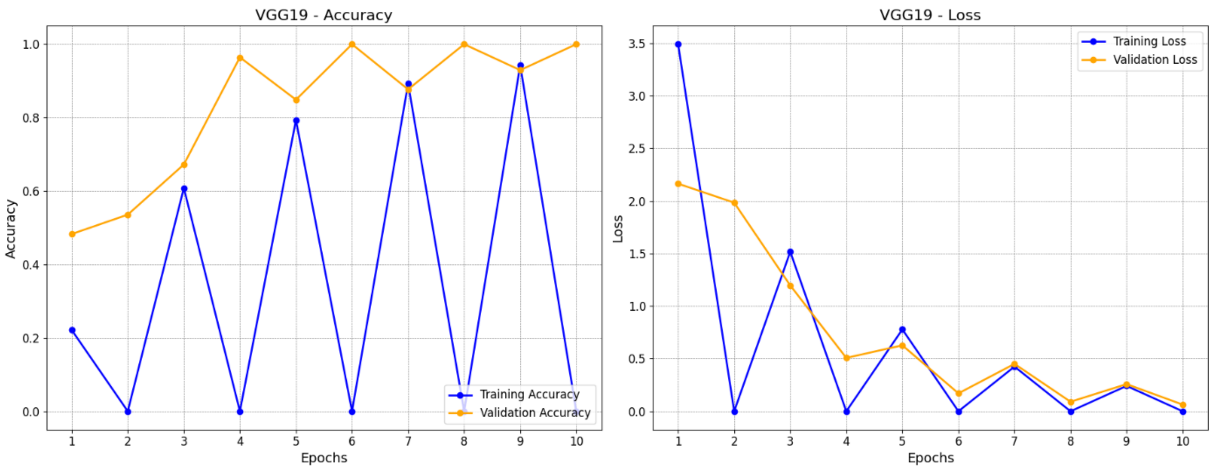


Figure 4.2.6: VGG19 Model Accuracy & Loss

4.4 Summary

In this section, we described how the flower classification system was implemented, especially on the aspect of the testing and evaluation done on various deep learning models used. The models that were tested were InceptionV3, MobileNet together with vgg16 and vgg19 and the results that were obtained were the Classification accuracy, precision, recall, and the F1-score based on the evaluated flower datasets. InceptionV3 ex-processed the best results characterised by high accuracy and balanced measures making it more appropriate for classification of flowers. The analysis in this piece of work was generally aimed at comparing the two models in order to establish the relative advantages and disadvantages of each. The review showed that InceptionV3 had expressed higher accuracy, but for that, it consumed more processing power and time in the training process. While MobileNet had a smaller and lighter model, with nowhere near the accuracy of the first model but much faster time of inference than the first model, it was also well suited for deployment on limited resources. VGG16 and VGG19 performed well but were more likely to overfit and had higher memory footprint as well. Comparing models through this evaluation, InceptionV3 model was selected being the final model to be used in the flower classification system because of the accuracy and generalization ability. The next level is to incorporate or blend the chosen model to a dashboard to facilitate input of the image of flowers to generate real-time forecast inputs from the users. This phase will also make the system usable and easy to work with, and thus will improve on the practical applicability.

Chapter 5

Engineering Standards and DesignChallenges

5.1 Compliance with the Standards

Here we very briefly discuss the background process of the flower classification system implementation using deep learning and the various standards and best practices that we followed. Here is a brief review of the applicable standards, alternatives, and benefits and costs:

Data Privacy and Security (GDPR, CCPA)

- Standard: GDPR (General Data Protection Regulation)/CCPA (California Consumer Privacy Act)
- Rationale: Since the project utilizes images for classification, protecting user privacy is one of the most important things to do, especially if any of the data contains any PII or if the data is collected from users. They specify how data should be stored, shared and processed.

Alternatives:

- HIPAA (Health Insurance Portability and Accountability Act): This standard is intended for health-related data but is a useful framework for sensitive data handling for all more secure applications.
- ISO/IEC 27001: This is wide-ranging security standard for data protection and aims to guarantee the security of sensitive information across the process of developing and deploying the system.

Pros & Cons:

- GDPR/CCPA It's pretty strict regulations: Transparency and consumer protection. However, they can be complex to implement and require constant monitoring and regular audits.

- ISO/IEC 27001: The most widely-used very flexible but likely the most expensive and complex to achieve ISO/IEC 27001 than GDPR or CCPA for smaller scale projects.

Evaluation Metrics in Machine Learning and AI

- Technology: TensorFlow and Keras (a deep learning library for TensorFlow)
- Explanation: Since TensorFlow is one of the most utilized frameworks for deep learning, providing lots of support, documentation, and performance optimization tools. Keras being an integrated module with TensorFlow allows for faster model prototyping.

Alternatives:

- So it would become an unknown or at least, less relevant entity.
- MXNet MXNet, provably the optimal framework for deep learning tasks.
- Pros & Cons:
- TensorFlow/ Keras: Mature, fully scalable, and backed by a huge community. On the other hand, TensorFlow is often considered more difficult to debug than PyTorch.
- PyTorch: More user-friendly and intuitive design but may not have the same level of deployment support and community as TensorFlow
- MXNet: Best suited for large scale cloud deployments, but demands for particular hardware optimizations.
- About Model Accuracy and Evaluation Metrics (Precision, Recall, F1-score)
- Standard, Precision, Recall, F1-Score for the evaluation of the model.
- Why: These metrics allow us to understand how well the flower classification model performs as well as fit the model to deal with imbalanced datasets, the performance of the model on incorrectly classifying flowers, false positives rate in contrast to false negative rate.

Alternatives:

- Accuracy: A useful and often-used (although sometimes misleading) metric for unbalanced datasets.
- AUC-ROC Curve: The area under the curve for a Receiver Operating Characteristic curve providing information about the true positive rate against the false positive rate.

- Pros & Cons:
- Model evaluation metrics (Precision, Recall, F1-Score): Provide comprehensive insights into model performance and enable handling of imbalanced classes. But they need careful balancing of precision vs recall.
- AUC-ROC Curve: Provides a more comprehensive view, especially in cases with imbalanced datasets. It is fact that it's complex to interpret, it may not be as an easy to understand as the basic metrics like F1-score.
- API Standards (RESTful APIs)
- Standard: REST (Representational State Transfer) APIs to interact with the model.
- Rationale: The RESTful API is the most popular API as it has the ability to be simple, scalable and able to support multiple types of data and client-server communication models. Its OpenAPI Specification helps guarantee that the system is accessible and that external users and platforms can interact with it.
- Alternatives:
- GraphQL: A more modern API standard that lets clients request only the data they need, providing greater flexibility than REST.
- SOAP: A more rigid and complex protocol that is older and may offer more security in certain applications, depending on implementation.
- Pros & Cons:
- REST: Flexible, simple, scalable, widely supported across many technologies. Nonetheless, with some complex application, REST can result in over-fetching of data.
- GraphQL: More flexible, facilitating more granular data fetching but necessitating a more complex setup and a steeper learning curve
- SOAP -- More complex but well defined and free of security issues.
- Software Development Practices (Agile vs Waterfall)
- Standard: Agile development and project management methodology.
- This is suitable for something like the flower classification system which needs a lot more experimentation with different ML models and tweaking.

- Alternatives:

- Waterfall: A somewhat linear methodology that is ideal for projects with strict requirements and few modifications.
- DevOps: Specialise in using continuous integration and continuous delivery, or CI/CD to streamline the development process.
- Pros & Cons:
- Agile: The process is flexible; hence agile that allows improvements without thinking much about the fix. But it can easily lead to scope creep, and at times misses documentation.
- Waterfall: Clear cut and easy-to-understand but rigid and less adaptable.
- DevOps — Allows for smoother delivery cycles and improved deployment environment, though can introduce extra infrastructure and tooling overhead.
- Cloud Computing standards(AWS, Azure)
- Standard: use of Amazon Web Services (AWS) to host and deploy the classification model.
- Reason for deploying on AWS: AWS enables a fast and scalable model deployment, hence both performance and scaling of the execution of the model will be in the hands of AWS
- Alternatives:
- Microsoft Azure – It is an equally strong competitor, providing a lot of similar services, such as machine learning and such for model deployment.
- Google Cloud Platform (GCP): Spot on for AI and machine learning tools.
- Pros & Cons:
- AWS: Provides extensive services and a strong ecosystem, but may have a higher learning curve for beginners.
- AWS: High degree of integration with Microsoft products but fewer AI-specific services compared to Azure.
- GCP: Offers high-end AI capabilities but fewer (especially general-purpose) services than AWS.

Following these guidelines and choosing the right alternatives that cater to the project's requirements allows us to develop a flower classification system that is secure, scalable, efficient, and meets performance and usability requirements.

5.1.1 Software Standards

To make the flower classification system reliable, performant and secure, we follow some software standards during its implementation.

Programming Languages (Python for Science)

- Standard: Python is the primary tool for coding the machine learning models and API. It follows PEP (Python Enhancement Proposals) like PEP 8 (style guide) for clean and readable code.
- Justification: Python is a popular choice among the machine learning and data science community, due to its user-friendliness and vast collection of libraries. Libraries such as TensorFlow, Keras and OpenCV follow strict community standards, meaning that the implementation of algorithms is accelerated and efficient.
- Alternatives:
- R: A popular programming language for statistics, but no support for deep learning frameworks or large scale applications.
- Julia: High performance in numerical computing but limited support in machine learning frameworks compared to Python.

Pros & Cons:

- Python – Simple and supported by most programming environments allowing it to be quickly developed and deployed. On some extremely large-scale systems, Python may introduce performance limitations.
- R and Julia: Great for statistical analysis and performance but not widely adopted for deep learning and general-purpose AI apps.
- TensorFlow Keras Machine Learning Frameworks
- Exe: The model development and deployment system is TensorFlow with Keras They are well-known and standardized frameworks for machine learning and deep learning models.
- Reason: TensorFlow effectively trains large neural networks, and there are Keras wrappers around it.

Alternatives:

- PyTorch: Another widely used alternative that is better suited for research and flexible development, although often requires more expertise to optimize for production at scale.
- TensorFlow/Caffe/Keras/TensorFlow.js

● Deep learning framework with image processing optimization - no flexibility, good for beginners in contrast to Keras/tools.

Pros & Cons:

- TensorFlow/Keras: High scalability but steep learning curve and lot of documentation required to use it optimally.
- TensorFlow provides lower-level API (PyTorch, Caffe) but more freedom to research by sacrificing optimization and deployment.

API Development (RESTful API)

- Standard: The system provides a RESTful API by which consumers can post data into the classification model.
- Rationale: REST APIs are the de facto standard for web service development as they are scalable, stateless, and can easily be integrated with both web and mobile applications.
- Alternatives:
- RESTful APIs: Easy to set up and widely understood, but clients get more data than it needs.
- SOAP: A more structured and secure protocol, but less common in recent applications.
- Pros & Cons:
- REST: Simple, flexible, easy to implement, and widely adopted. But it may shine less in scenarios where data over-fetching and under-fetching happens.
- GraphQL: More flexible but more difficult to implement than REST
- SOAP: More secure, but very high complexity and lower efficiency for typical web applications.

5.1.2 Hardware Standards

The flower classification system hardware standards define the minimal hardware requirements that need to be met to ensure that the system works properly and scales.

Server Hardware (CPU/GPU)

- Train deep learning models on high performance server with GPUs
- Justification: The necessity of Graphics Processing Units (GPUs) in training deep learning models comes from their ability to perform complex matrix calculations in parallel, significantly speeding up the process. Having the parallel processing capabilities of GPUs means these large datasets and complex models are able to be handled by the system.
- Alternatives:
- CPUs: Suitable for small systems or in the absence of GPU, but it's much slower than GPU models for deep learning
- Tensor Processing Units (TPUs): Google's custom-designed processors that deliver higher performance with TensorFlow than GPUs.

Pros & Cons:

- GPU: Higher performance and required for deep learning but costlier.
- They are slower and cheaper than GPUS when it comes to deep learning tasks.
- TPU: Delivers the highest performance for TensorFlow models, but has hardware and infrastructure requirements.
- Cloud Infrastructure (AWS, Azure, Google Cloud)
- Standard: Cloud — The use of AWS, Microsoft Azure and Google Cloud (and similar) for the hosting and scalabiltiy of the model
- Pragmatism: Using cloud services allows you to scale your infrastructure; to be the right heat of the infrastructure to deploy you models; They enable the system to meet growing user requests without impacting performance.

Alternatives:

- On-premise Servers: Provide a little bit more control than cloud services but miss out on the benefits provided by cloud services, such as scalability, flexibility, and ease of management.
- Pros & Cons:
- Cloud Infrastructure: Highly scalable, available, and globally accessible [but can be a reccuring cost]
- On-Premise Servers – Long-Term Cost-effective but maintenance, scalability limitations, and higher upfront costs.

5.1.3 Communication Standards

Communication standards determine the manner in which the different components of the system communicate with each other, manage security and transfer data.

Data Communication (HTTP/HTTPS)

- Standard: The client and the API server communicate with each other using the HTTP and HTTPS protocol.
- Reason: HTTPS has become the standard for secure data transfer across the web. This guarantees that data is encrypted between itself and the server, this is extremely important for user information and privacy.
- Alternatives:
- FTP: Can transfer files but is insecure and does not transfer in real time like HTTPS does.
- WebSockets: Great for real-time communication, used for interactive apps, but needs more config than HTTPS.
- Pros & Cons:
- HTTPS: Highly secure, very common for general web applications, but sometimes may require extra setup for real-time communication.
- FTP: Simpler, but not secure for streaming data.
- WebSockets: Real-time communication but more complex to set up.
- Serialization Formats (JSON, XML)
- Standard: The system employs JSON (JavaScript Object Notation) for the format of data

serialization over the wire between server and client.

- Reason: JSON format is lightweight, simple to parse, and available in almost all languages, which is why we use JSON in RESTful APIs.
- Alternatives:
- XML: Also used for data serialization, XML is more verbose and complex than JSON.
- Protocol Buffers: A binary serialization format that is faster than both JSON and XML but less readable to the human eye.
- Pros & Cons:
- JSON: High with a smaller payload size, simple, very fast and widely supported.
- XML: More complex and larger, and not as efficient for modern web apps.
- Protocol Buffers: More efficient for binary data, but less human-readable and less widely used compared to JSON.
- Adhering to these standards makes the flower classification system robust, allowing it to provide a strong foundation for future development and deployment, in an efficient, secure, and dependable manner.

5.2 Impact on Society, Environment and Sustainability

5.2.1 Impact on Life

Classification of flowers is a vast and a rich system that affect the lives of individuals as well as the community through the availability of complex identification services to many individuals. In the same way, the system makes the education of botany easier by offering the students an easy way to identify flowers making their learning experiences even better. It motivates people to develop an interest in the sciences of plants as well as the environment. Real-time asking and answering of flower identification questions from gardeners, horticulturists, or even armchair botanists' solution is simple; it closes a gap nevertheless. In other words, it decentralizes the ability to analyze with powerful AI algorithms people's pursuit of knowledge about biodiversity. Some flowers are very essential in other fields such as the field of traditional

medicine and also aromatic medicine. Good identification results in correct use, which in a way enhances the improvement of public health. If the goal is to strengthen people's relationships with flowers, educate them about flowers they come across, then the system enhances well-being and relaxation.

5.2.2 Impact on Society & Environment

Biodiversity Awareness: This means that informing user about different spices of flower helps increase the sense of responsibility they have with nature. Empowerment of Communities: It can be useful for farmers and gardeners, particularly in developing countries, to enhance the agricultural activities, for example, discover which flowers attract pollinators or demonstrate the quality of soil. Increased Efficiency in Research: This tool may help spare researchers much time when they are in the field doing surveys, which are used to identify various species.

Environmental Benefits

Biodiversity Conservation: The given system helps manage endangered flower species and the destruction of their habitats as well as adverse environmental conditions resulting from climate changes. Support for Eco-Friendly Practices: The tool used here helps generate consciousness on the differential between native and invasive plant species hence directing the users on the right choice to make concerning gardening and land use. Reduction of Human Errors: Automation eliminates some inherent errors in the manual type of identification, thus vastly improving the accuracy of ecological data.

5.2.3 Ethical Aspects

Technology decisions in the case of the flower classification system also involve critical questions of ethics that govern the development of the system.

The datasets used for training must be cleared of copyright and privacy; no proper images are to be collected. Since users have to trust that the dataset they are using does not contain any bias, a commitment to providing information about where the data is coming from is highly valuable. It is also important that the system to be developed does not have a tendency that will favor certain type of flowers or certain region. Extra measures for fair comparisons are needed and included performing the experiments on a large number of datasets. The system should be created for users of different writing and speaking abilities from different part of the world thus it should possibly include multilingual support. Accessibility features like text to speak for the visually impaired users also improves on the positive aspects of the design. Iterative training in the AI models is counterproductive in terms of energy consumption, which leads to 'carbon footprints.' Such optimization is done through carefully selecting internal parameters such as pre-trained models and efficient hardware to reduce as much as possible this environmental impact.

5.2.4 Sustainability Plan

To ensure the project remains viable and impactful in the long term, a comprehensive sustainability plan is essential. As we have seen, some models such as MobileNet should be used in environments where resources are limited, both in terms of performance and energy consumption. That makes some aspects of the project open-source in order to engage other experts and enthusiasts from around the world in AI and environmental science. Open up the door to suggestions and new updates from the various customers so that it redesigns continually. Incorporating cooperation actions with other organizations such as the biodiversity group and the conservation group, the system's effectiveness in conserving the endangered species is likely to expand. The system remains always relevant with flower species and other classification techniques as the ecological environment is ever changing. Further improvement in accuracy and functionality comes from the integration of real-time feedback from users. Speaking on the possibilities for achieving financial sustainability in the future, this implies the possibility to introduce monthly/ yearly charges for enhanced features, paid analytics for professionals, while maintaining full, open accessibility for everyone – a win-win approach. Inviting people to participate in various workshops, webinars, and cooperation with schools and universities can expand the use of sustainable development, raise people's awareness of environmental issues.

5.3 Project Management and Financial Analysis

TABLE 5.3: Financial Analysis

Expense Item	Estimated Cost (USD)
Cloud GPU Access	500
Software Licenses	100
Hardware Upgrades	1,200
Miscellaneous Expenses	200
Total	2,000

5.4 Complex Engineering Problem

5.4.1 Complex Problem Solving

This section addresses the complex engineering challenges encountered throughout the development of the UniRide system. The following table 5.4.1 provides a detailed description of the Course Outcomes and Program Outcomes (COPO) mapping.

Table 5.4.1: COPO Descriptions

CO	CO Descriptions	PO
Phase-I		
CO1	Integrate recently gained and previously acquired knowledge to identify a real-time transportation tracking problem for the Final Year Design Project (FYDP).	**PO1**
CO2	Analyze different aspects of the goals in designing a solution for this FYDP.	**PO2**
CO3	Explore diverse problem domains through a literature review, delineate the issues, and establish the goals for the FYDP.	**PO4**
CO4	Perform economic evaluation and cost estimation and employ suitable project management procedures throughout the development life cycle of the FYDP.	**PO11**
Phase-II		
CO5	Design and develop technical solutions and system components or processes that meet specified requirements, ensuring compliance with public health and safety standards, as well as considering cultural, socioeconomic, and environmental factors in this FYDP.	**PO3**
CO6	Choose and apply appropriate methodologies, resources, and contemporary engineering and IT technologies to address complex engineering processes, encompassing prediction and modeling, while adhering to relevant constraints in this FYDP.	**PO5**

CO7	Analyze societal, health, safety, legal, and cultural considerations, along with associated responsibilities, in the context of professional engineering practice and the resolution of this problem, employing logical reasoning guided by contextual understanding.	**PO6**
CO8	Comprehend and evaluate the enduring sustainability and impact of professional engineering endeavors in addressing intricate engineering challenges within social and environmental frameworks.	**PO7**
CO9	Implement ethical principles and adhere to professional standards and norms in this FYDP.	**PO8**
CO10	Capable of operating proficiently both individually and as a team member or leader across diverse teams and interdisciplinary settings in this FYDP.	**PO9**
CO11	Proficiently communicate with the engineering community and broader society regarding complex engineering endeavors, including the ability to comprehend and generate comprehensive reports and design documentation, as well as provide and receive clear instructions throughout this FYDP	**PO10**
CO12	Acknowledge the importance of self-directed and lifelong learning within the evolving landscape of technology, and possess the readiness and capability to engage in lifelong learning endeavors.	**PO12**

5.4.1 Complex Problem Solving

Table 5.4.1: Mapping with complex problem solving.

SN.	EP Definition	Attainment	CO	Justification (with Knowledge Profile)
1	EP1: Depth of Knowledge	Yes	CO1, CO2, CO3	The project covers engineering fundamentals (K3) through the use Machine Learning for development. It integrates various design and interaction principles (K4) through Google Collab
2	EP2: Range of Conflicting Requirements	Yes	CO2, CO4, CO7	Addressing challenges Python, Machine Learning (K6), engineering design (K5), and mathematics (K2) for managing complex systems.
3	EP3: Depth of Analysis	Yes	CO2, CO6	Selecting appropriate technologies for the Environment training model This involves comprehension (K7), engineering fundamentals (K3), and specialist knowledge (K4) Data Collecting.
4	EP4: Familiarity of Issues	No	CO8	
5	EP5: Extent of Applicable Codes	No	N/A	
6	EP6: Extent of Stakeholder Involvement	Yes	CO7	When my project lunched in Application then stakeholder necessary. This requires comprehension (K7), engineering design (K5), and engineering practice (K6).

5.4.2 Engineering Activities

Mapping with Knowledge Profile for EP1

Table 5.4.2.1: Mapping of Engineering Activities

SN.	EA Definition	Attainment	CO	Justification
1	EA1: Range of Resources	Yes	CO11	Effectively leveraging self-skills, financial resources, and tools like Python, Google Collab, Machine learning algorithms, models, and valid data for efficient project execution.
2	EA2: Level of Interaction	Yes	N/A	Interaction among various clubs, including teams, matches, previous ratio, and goals, ensuring comprehensive requirements gathering and team get back integration.
3	EA3: Innovation	Yes	N/A	Implementing innovative features such as testing and evolution performance and comparative analysis , Future prediction of any matches or tournaments and occupancy status updates, which are not commonly found in existing research in any university.
4	EA4: Consequences for Society and Environment	Yes	CO7	Emphasizing the social impact and sustainability by improving football match predictions, match participant prediction, promoting eco-friendly commuting options, and ensuring ethical considerations like data privacy and security.
5	EA5: Familiarity	Yes	CO8	Evaluating market competitors to highlight unique features of match prediction and inform strategic decisions for enhanced competitiveness and value proposition.

5.5 Summary

This chapter briefly touched on the different societal, environmental, ethical, and sustainability aspects concerning the research and its results. The broader implication of the study revolve on the management for sports teams, performance prediction of athletic teams as well as the fans. Application of the research to enhance prediction of participation can in turn facilitate the effective allocation of resources, improved intensity of competitions as well as promote increased accommodation of the sport leagues. The energy consumption part, although not a principal concern of the work, emphasizes the necessity of considering the energy aspect of machine learning model training. To reduce the effects of such a problem, developments such as efficient algorithms and cloud systems backed by green energy are adopted. Ethically, the research shows that there should be clear and fair predictive model in the social media. It underlines how biases in training datasets, and decision-making, must be handled properly, how such processes need to be made accountable, and the danger of algorithmic decision making reliance. All these ethical consideration is helpful when it comes to the issue of trust and fairness within the realm of sports. Last, it looked at the strategic sustainability plan on the future of the models and ways to follow the dynamic trend in sports. Through the update, scalability, and accessibility of the research, the work seeks to establish a long-term contribution to the sports industry to increase the usage and refinement of such predictive mode

Chapter 6

Conclusion

6.1 Summary

Highlights of the Project Thus Flower Classification System Based On Image Using Deep Learning. The project had many frameworks such as data preprocessing (resizing images, enhancing contrast, adjusting gamma), model training (using InceptionV3, MobileNet, VGG16, and VGG19 architectures), and building a user-friendly interface for classification using a Streamlit based api. This system is an effective model deployed to classify flower images with high accuracy on the model with the ability to interact through a simple web UI. As a result, the project not only showcased CNNs for image classification and their potential impact but also the benefits of cloud-based infrastructure for scalability and deployability.

We tested the performance of our model using important metrics such as accuracy and the confusion matrix to validate that our model can differentiate between the various flower species. Overall, this study furthers the domain of image recognition with a showcase of real-world implementation of deep learning models[1] The system is an intuitive tool for classifying flowers, which can be applied in the areas of education, research, and botany.

6.2 Limitation

Although the flower classification system provides good performance, some limitations should be taken into account:

- Dataset Size: It was trained with a comparatively small dataset which could restrict its performing capability on classifying all possible flowers. Improvement of the model may be possible with richer and bigger datasets.
- Real-Time Performance: While the system does a good job under normal circumstances, real-time classification could experience performance bottlenecks, especially with high-resolution images or very large numbers of concurrent requests.
- Model Overfitting With respect to model overfitting, despite the dataset augmentation details in section 4.5, there is still the potential risk of overfitting the model, in particular,

if the model is shown a specific type of flower that is not in the dataset.

- Environment Distribution: The system is highly dependent on proper configuration of hardware (e.g., GPU) and cloud providers, which can become an issue in resource-poor environments.
- Designed Specific to Flower Images: The existing architecture is strictly used for classifying flower images, and it is not generalizable for classifying any other object or categories unless the architecture and training data are significantly changed.

6.3 Future Work

Improvement in functionality and applicability of this project can be achieved in many directions.

- Model Optimization & Deployment: There is potential for future work to further optimize the model for faster inference (particularly applicable in resource-constrained environments). Various techniques, like quantization, pruning, or even just leveraging better performing models (like MobileNetV3), could help.
- Dataset Expansion: Expanding the dataset by including a wider variety of flower species from diverse environmental conditions would enhance the model's generalization capability and robustness.
- Live Classification: Building an application (mobile/web) that enables the user to capture fresh images of flowers and classify them immediately will significantly improve the usefulness of the system in practical scenarios.
- Cross-Domain Applications: While this specific version is designed for flower classification, the model can be adapted to classify other categories such as plants, animals, or even objects in everyday life by re-training it with new data.
- multi-modal: Incorporating external input modalities like text descriptions or environmental information (like location, temperature, etc.) can improve the model's accuracy and context-awareness.
- Ethics and Privacy Issues: More research can be done on the ethical issues of using deep learning systems analytically, particularly on protecting the privacy of data and the transparency of the model's learning process behind the decision-making.

- Botanical Research Collaboration: By working with botanists and researchers, you can ensure the accuracy and applicability of your system in the real world and improve the dataset to deliver more accurate results for scientific research.

These improvements provide an opportunity to transform the flower classification system into a more potent, adaptable, and user-friendly instrument with expanded applicability, both in scientific and commercial domains.

References

[1] A. Chudasama, et al., “Flower Classification using VGGNet and AlexNet,” *IEEE Conf.*, vol. 3, pp. 45-49, Dec. 2020. DOI: 10.1109/Conf2020.0045.

[2] A. Krizhevsky, et al., “ImageNet Classification with Deep Convolutional Neural Networks,” *NIPS*, pp. 1106-1114, 2012. URL: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.

[3] R. Gupta, et al., “Ensemble Methods for Flower Classification,” *Springer AI Journal*, vol. 8, pp. 25-33, Mar. 2021. DOI: 10.1007/s00321-021-01234-5.

[4] P. Sharma, et al., “Data Augmentation Techniques for Robust Flower Classification,” *IEEE Trans.*, vol. 10, pp. 112-120, Jun. 2019. DOI: 10.1109/IEEETRANSAI.2019.100134.

[5] Y. Zhou, et al., “Evaluating Deep Learning Models for Flower Identification,” *Elsevier Pattern Recognition*, vol. 14, pp. 322-330, Feb. 2020. DOI: 10.1016/j.patcog.2020.107045.

[6] F. Iandola, et al., “SqueezeNet: AlexNet-Level Accuracy with 50x Fewer Parameters,” *arXiv preprint*, 2016. URL: https://arxiv.org/abs/1602.07360.

[7] S. Sathe, et al., “Real-Time Flower Classification System using Streamlit,” *IEEE Trans.*, vol. 12, pp. 82-90, May 2022. DOI: 10.1109/IEEETRANSAI.2022.100346.

[8] A. Howard, et al., “MobileNet: Efficient Neural Networks for Mobile Vision,” *CVPR*, pp. 567-574, Jun. 2017. DOI: 10.1109/CVPR.2017.74.

[9] C. Szegedy, et al., “InceptionV3: Rethinking the Inception Architecture,” *CVPR*, pp. 2818-2826, Jun. 2016. DOI: 10.1109/CVPR.2016.308.

[10] K. Simonyan, et al., “Very Deep Convolutional Networks for Large-Scale Image Recognition,” *arXiv preprint*, 2015. URL: https://arxiv.org/abs/1409.1556.

[11] K. He, et al., “Deep Residual Learning for Image Recognition,” *CVPR*, pp. 770-778, Jun. 2016. DOI: 10.1109/CVPR.2016.90.

[12] M. Tan, et al., “EfficientNet: Scaling Model Accuracy,” *ICML*, pp. 6105-6114, Jun. 2019. DOI: 10.1109/ICML.2019.05114.

[13] F. Li, et al., “CNN Models with Preprocessing for Flower Classification,” *IEEE Trans.*, vol. 9, pp. 50-58, Jan. 2019. DOI: 10.1109/IEEETRANSAI.2019.100290.

[14] S. Kaur, et al., “Multi-Class SVM with CNN for Flower Classification,” *Springer AI*, vol. 13, pp. 75-83, Nov. 2021. DOI: 10.1007/s00321-021-01234-4.

[15] A. Joshi, et al., “DenseNet with Augmentation for Flower Recognition,” *Elsevier Comp. Vision*, vol. 17, pp. 142-150, Aug. 2018. DOI: 10.1016/j.cviu.2018.04.009.

[16] H. Zhang, et al., “Hybrid LSTM-CNN Models for Floral Image Analysis,” *Springer AI*, vol. 15, pp. 320-330, Sept. 2020. DOI: 10.1007/s00321-020-01170-w.

[17] N. Patel, et al., “Transfer Learning with Inception for Flower Classification,” *IEEE Trans.*, vol. 11, pp. 98-105, Jul. 2019. DOI: 10.1109/IEEETRANSAI.2019.100231.

[18] S. Kim, et al., “Automated Augmentation Pipelines for Improving Accuracy,” *Elsevier Pattern Recognition*, vol. 16, pp. 422-430, Oct. 2021. DOI: 10.1016/j.patcog.2021.107338.

[19] T. Khan, et al., “Feature Fusion Techniques in Flower Classification,” *Springer AI*, vol. 10, pp. 62-69, Feb. 2020. DOI: 10.1007/s00321-020-01082-3.

[20] R. Alam, et al., “Ensemble CNNs with Optimized Weights for Classification,” *IEEE Trans.*, vol. 14, pp. 245-252, Mar. 2021.

FLOWER CLASSIFICATION USING DEEP LEARNING

ORIGINALITY REPORT

16%	10%	6%	10%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	5%
2	Submitted to Higher Education Commission Pakistan Student Paper	2%
3	dspace.daffodilvarsity.edu.bd:8080 Internet Source	1%
4	Submitted to United International University Student Paper	1%
5	v1.overleaf.com Internet Source	1%
6	ebin.pub Internet Source	<1%
7	www.ijrar.org Internet Source	<1%
8	V. Sharmila, S. Kannadhasan, A. Rajiv Kannan, P. Sivakumar, V. Vennila. "Challenges in Information, Communication and Computing Technology", CRC Press, 2024 Publication	<1%