LOCAL FLOWER DETECTION THROUGH DEEP LEARNING APPROACH

BY

Tanu Sarkar

ID: 0242310004213001

This Report Presented in Partial Fulfillment of the Requirements for the Degree of MS in Management Information System (MIS).

Supervised By

Dr. Sheak Rashed Haider Noori
Professor & Head
Department of CSE
Daffodil International University

Co-Supervised By

Arif Mahmud
Associate Professor
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

6 JULY 2024

APPROVAL

This Thesis titled **"Local flowers detection through deep learning approach"**, submitted by Tanu Sarkar, ID No: 0242310004213001, 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 MS. in Management information system and approved as to its style and contents. The presentation has been held on 6 July 2024.

BOARD OF EXAMINERS

Narayan Ranjan Chakraborty
Associate Professor and Associate Head
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

Board Chairman

Md. Sadekur Rahman
Assistant Professor
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner 1

Dr. Ohidujjaman Tuhin
Assistant Professor
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner 2

Sadat Hossain
Data Scientist
Risk Management Division,
BRAC Bank Limited

External Examiner

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Dr. Sheak Rashed Haider Noori, Professor & head, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:

Dr. Sheak Rashed Haider Noori
Professor & head
Department of CSE
Daffodil International University

Co-Supervised by:

Arif Mahmud
Associate Professor
Department of CSE
Daffodil International University

Submitted by:

Tanu Sarkar

Tanu sarkar
ID: 0242310004213001
Department of MIS
Daffodil International University

ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project/internship successfully. We really grateful and wish our profound our indebtedness to **Dr. Sheak Rashed Haider Noori**, **Professor & Head**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of "Deep learning" to carry out this project. Her endless patience ,scholarly guidance ,continual encouragement , constant and energetic supervision, constructive criticism , valuable advice ,reading many inferior draft and correcting them at all stage have made it possible to complete this project. We would like to express our heartiest gratitude to **Mr. Dr. Sheak Rashed Haider Noori, Professor and Head** Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work. Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

Flowers are some of the most beautiful creations of the Almighty. There are numerous seasonal flowers in Bangladesh. Each flower has its unique color, shape, and name. In our daily lives, we can see a variety of flowers when walking along highways, rail lines, and even in our gardens. However, the majority of our population is unaware of the significance of this beautiful creation. The identification and classification of local flower species using deep learning techniques has important implications for botanical research, biodiversity protection, and ecological monitoring. In this paper, we present an in-depth analysis of multiple deep learning models for detecting and classifying flowers from a dataset of 2400 photos of various flower species the tested models consist of VGG-16, MobileNet, M-1, and ResNet-50. Each model was selected based on its distinct architecture and established efficacy in image recognition tasks. The results of our study indicate that although all models work well, there are significant variations in their abilities. Specifically, the VGG-16 model demonstrates great potential for real-time applications and achieving high accuracy of this dataset, with a rate of 97.73%.Six different kinds of flowers have been collected for our research. Our dataset consists of 2450 photos overall and 410 images per flower. We allocated 60% of the images to the training set in order to train the dataset. The validation set is assigned 20% of the total data, while another 20% is allocated to the test set for the purpose of testing. To improve the models' robustness and generalizability, we apply preprocessing techniques like resizing, normalization, and data augmentation. Six different kinds of flowers are used in this system. However, we plan to add more flowers in the future to make our system better.

Keywords: Deep Learning, Resnet-50, VGG-16, Augmentation, MobileNet.

TABLE OF CONTENTS

CONTENTS **PAGE**

LIST OF FIGURES

List of Tables

CHAPTER 1

Introduction

1.1 Introduction

Imagine the wonder of nature without flowers. A bloom is a good way to identify a plant. There are already a great variety of flowers in the globe, and more and more varieties are being found every day. In order to find the exact flowers they are hoping to get, individuals will occasionally use search engines like Google. On sometimes, as we stroll along park or roadside paths, we come upon a flower that is completely new to us. Due to the great diversity of flower species, as well as variances in color, shape, size, and environmental circumstances, flower detection and classification is a difficult task. Flower identification techniques are time-consuming and often require expert knowledge. But now that deep learning has arrived, this procedure can be automated, which speeds it up and makes it easier for non-experts to understand. A branch of machine learning called deep learning has demonstrated impressive performance on image identification challenges. Deep learning approach, in particular, are very good at processing and interpreting visual input because they can recognize hierarchical structures in pictures. The effectiveness of various deep learning models, including VGG-16, MobileNet, M-1, and ResNet-50, in identifying and categorizing native flowers is examined in this study. These models present a easiest way to overcome the difficulties associated with manual identification in the context of flower detection.

1.2 Motivation

We have made an effort to develop a system that can identify and categories a few native flowers. Because the majority of people in our nation, especially those living in cities, are unaware of the variety of local flowers. Many of our university's students are keen to work with locally cultivated flowers, most of them are unaware of the specific varieties. Our younger generation, who are somewhat knowledgeable with local flowers, will also find it useful. Because of this, since we work with flowers, we want to make a method that will help everyone classifying and identifying flowers.

1.3 Objective

- Our goal is to create a system that allows individuals to recognise and identify unique native flowers found in Bangladesh.
- Learn which method produces the best results for our dataset; this is one of our system's secondary goals.
- We use local flowers, yet many people in our city are unaware of them. So that our built technology can assist them.

1.4 Expected outcome

1. Model Performance Comparison: With regard to local flower detection, we hope to ascertain how well VGG-16, MobileNet, M-1, and ResNet-50 perform in comparison. This will involve a thorough examination of each model's accuracy, precision, recall, and computing efficiency.
2. The goal of our thorough evaluation process is to determine which deep learning model provides the optimum combination of computing efficiency and accuracy for flower detection. This model will be the suggested method for assignments involving similar botanical classification.
3. Practical Applications: Botanical research, biodiversity protection, and environmental monitoring are anticipated to benefit from the findings of this study. The results will aid in the creation of automated plant identification systems and aid in the preservation of the local flower.

CHAPTER 2

Background Study

2.1 Introduction

In this part we will discuss several relevant studies that has been conducted to our project. We employed some deep learning models in our research. Deep learning models are widely utilized picture classification technology that is primarily utilized for classifying flowers. There are 2450 photos in our dataset overall, with 405 images for each flower. We use six different kinds of flowers. Our dataset was trained using a machine learning method.

2.2 Related works

Deep learning is a highly popular technique for automatically classifying and recognizing photos. There are several transfer learning models that are particularly effective in deep learning. Frequently used for tasks involving images, such as segmentation, classification, and recognition. Details of some of the transfer learning models that researchers use most frequently, along with an overview of the projects they work on, is provided below: Hazem Hiary[1] A novel two-step deep learning classifier that can recognize flowers from a range of species is offered by the authors. The flower area is first automatically divided into segments so that the smallest bounding box may be located around it. Second, they build a powerful convolutional neural network classifier to distinguish between the various flower varieties. Their classification accuracy is higher than 97%. Y Liu [2] create a large dataset of 79 kinds of flower photos and propose a framework based on CNN to address this problem and obtained 76.54% accuracy. T Ensari [3] presents a method for classifying flower photos that uses data augmentation and deep neural networks. Initially, they suggested a classification model that would enhance the accuracy of flower image classification. This model would employ Deep CNN to extract features and a variety of machine learning algorithms for classification purposes. Secondly, they used data augmentation process to get better accuracy and they achieved 99% accuracy. MVD Prasad [4] proposed using convolutional neural networks (CNN), a powerful AI tool, to classify flower images. They designed various CNN architecture and test their dataset to get accuracy. They got 97.78% accuracy. A Lodh [5] to achieve the goal of distinguishing flower images from their natural background, they designed and implemented a segmentation algorithm based on average color and color distribution variance. Then they tested their dataset and got 85.93% accuracy. Busra Rumeysa Mete [6] proposed some machine learning classifier like multilayer perception, KNN, random forest, SMV. They processed their dataset and test their dataset to get accuracy. At last they got 99% of accuracy. U C. Dey [7] suggested a system that can classify photos of fruits using a transfer learning method. They employed a few transfer learning models, including MobileNet, Inception-v3, and VGG-19. Then they trained and tested their dataset and achieved 99.53% of accuracy.

2.3 Research Summary

In this research first I create a dataset of local flower that contain 2450 images. I collect six types of different flower's images. I used some deep learning models like VGG-16, MobileNet, M1 and ResNet-50 for image classification. Then I trained my dataset by those algorithms. For train my dataset I declared the path of my dataset and the models trained respectively. Before that I preprocessed my dataset because in raw data there was always some noise in the dataset. So, to get better accuracy data preprocessing is must. After that I tested my dataset and got the validation accuracy.

2.4 The advantage of our research

Our dataset can be used by any developer to build an Android application. A developer has opportunities to construct a software system that can recognize different types of flowers and offer a concise explanation about each of them by utilizing Python or any other programming language. The CNN architecture may be enhanced, and there is a lot of room for dataset improvement. Our work can be helpful to anyone who wishes to work with CNN and Android development.

2.5 Challenges

Gathering the dataset for our study was an extremely difficult undertaking. The majority of the images in our dataset were not gathered online. We were able to acquire around 90% of our dataset. We therefore faced a significant hurdle in gathering the dataset. Our task was to identify duplicate blooms. We are aware that many flowers have the same size and shape. In this instance, achieving a high degree of precision is never easy for us. Color recognition was difficult. Certain flowers, like the red, pink, and white roses, come in numerous colors. It was difficult to determine color using CNN.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Introduction

This part will cover some of the steps we followed to complete our research. There are numerous steps that we must take. Data gathering, preprocessing, and noise reduction are some of the steps we took. Tensorflow is what we utilize to train our dataset. Subsequently, we look at testing our entire dataset. Convolutional neural networks were the primary tool we used in our research to carry out our project. We employed several models, including MobileNet, VGG16, RseNet50, and a customized model, to obtain accuracy for our dataset.

3.2 Procedure for collecting data

We took the majority of the project's images with our phones. Another image that we gathered from Google. We have gathered six varieties of native flowers found in Bangladesh, including joba, noyontara, rose, gadha, dalia, and jarbera. For our study, we gathered 2450 images in total. Every bloom has 410 pictures. We used 481 images in the testing set, 482 images in the validation set across six classes, and training set contains 1450 photos to train our dataset. Dataset for the gadha flower is displayed in the following figure.

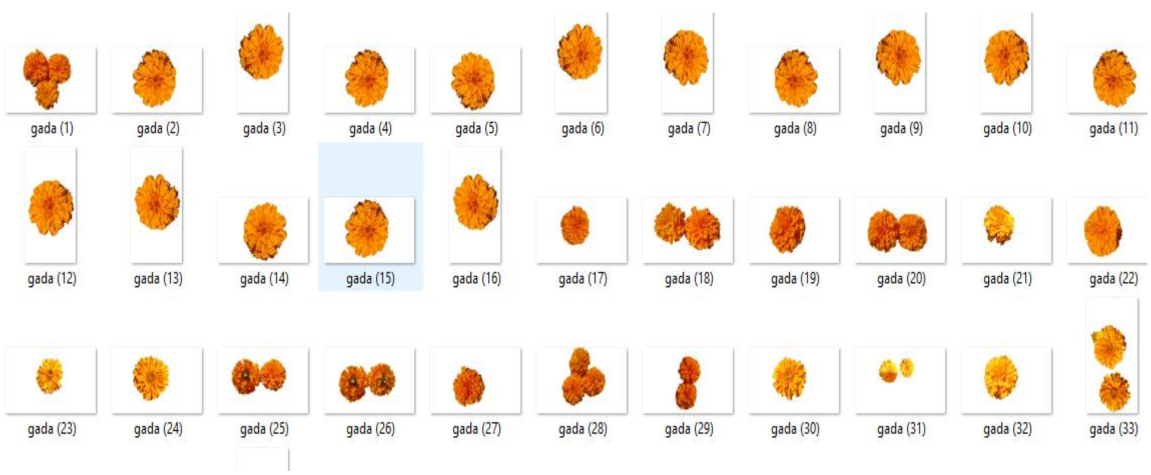

Fig. 3.2.1: Sample dataset of gadha

3.3 Statistics for data summaries

TABLE 3.3.1: TRAIN DATA

Flower's Name	Quantity
Joba Flower	241
Gadha Flower	241
Nayontara Flower	241
Jarbera Flower	241
Dalia Flower	241
Rose Flower	241

TABLE 3.3.2: VALIDATION DATA

Flower's Name	Quantity
Joba Flower	81
Gadha Flower	81
Nayontara Flower	81
Jarbera Flower	81
Dalia Flower	81
Rose Flower	81

TABLE 3.3.3: TEST DATA

Flower's Name	Quantity
Joba Flower	81
Gadha Flower	81
Nayontara Flower	81
Jarbera Flower	81
Dalia Flower	81
Rose Flower	81

3.4 Image preprocessing

The majority of the time, the raw dataset produced low accuracy results in addition to unexpected results. Because some irrelevant or noisy data can be found in raw data. Therefore, picture pre-processing is necessary to provide the best accuracy and appropriate results.

3.5 Training model

We used four different convolutional neural network models to train our dataset.

Models are listed below:

- M-1
- MobileNet
- ResNet-50
- VGG-16

CHAPTER 4

EXPERIMENTAL RESULTS AND DISCUSSION

4.1 Introduction

In this section, we shall discuss two types of outcomes: the confusion matrix and the model's accuracy analysis. Additionally, we display plot diagrams for every model. We employed four different deep learning models MobileNet, M1 model, ResNet-50, and VGG-16 to get the classification results. We have evaluated our dataset independently with each model to determine which one works best with our dataset.

Confusion matrix

When the actual values of the test data are known, a classification model may be evaluated using a confusion matrix, a table that displays the results. The results of an algorithm's execution can be seen graphically.

4.2 Results of experiments and analysis for the MobileNet model

Epoch-wise accuracy levels

In epoch number 13, we achieved a remarkable accuracy of 89%.

```
Epoch 9/50
88/89 [============================>.] - ETA: 0s - loss: 0.2326 - accuracy: 0.9680
Epoch 00009: val_accuracy improved from 0.86598 to 0.87423, saving model to MobileNetV2model.h5
89/89 [==============================] - 12s 139ms/step - loss: 0.2311 - accuracy: 0.9677 - val_loss: 0.6530 - val_accuracy: 0.8742
Epoch 10/50
87/89 [============================>.] - ETA: 0s - loss: 0.1791 - accuracy: 0.9784
Epoch 00010: val_accuracy did not improve from 0.87423
89/89 [==============================] - 12s 140ms/step - loss: 0.1800 - accuracy: 0.9782 - val_loss: 0.5551 - val_accuracy: 0.8412
Epoch 11/50
88/89 [============================>.] - ETA: 0s - loss: 0.1461 - accuracy: 0.9751
Epoch 00011: val_accuracy did not improve from 0.87423
89/89 [==============================] - 12s 138ms/step - loss: 0.1445 - accuracy: 0.9754 - val_loss: 0.6587 - val_accuracy: 0.8680
Epoch 12/50
87/89 [============================>.] - ETA: 0s - loss: 0.0945 - accuracy: 0.9842
Epoch 00012: val_accuracy did not improve from 0.87423
89/89 [==============================] - 13s 141ms/step - loss: 0.0924 - accuracy: 0.9845 - val_loss: 0.7384 - val_accuracy: 0.8577
Epoch 13/50
87/89 [============================>.] - ETA: 0s - loss: 0.0802 - accuracy: 0.9835
Epoch 00013: val_accuracy improved from 0.87423 to 0.89485, saving model to MobileNetV2model.h5
89/89 [==============================] - 13s 143ms/step - loss: 0.0792 - accuracy: 0.9831 - val_loss: 0.7141 - val_accuracy: 0.8948
Epoch 14/50
88/89 [============================>.] - ETA: 0s - loss: 0.1593 - accuracy: 0.9780
Epoch 00014: val_accuracy did not improve from 0.89485
89/89 [==============================] - 13s 146ms/step - loss: 0.1575 - accuracy: 0.9782 - val_loss: 2.9254 - val_accuracy: 0.7567
Epoch 15/50
87/89 [============================>.] - ETA: 0s - loss: 0.2183 - accuracy: 0.9705
```

Fig. 4.2.1: MobileNet model’s accuracy

Confusion matrix:

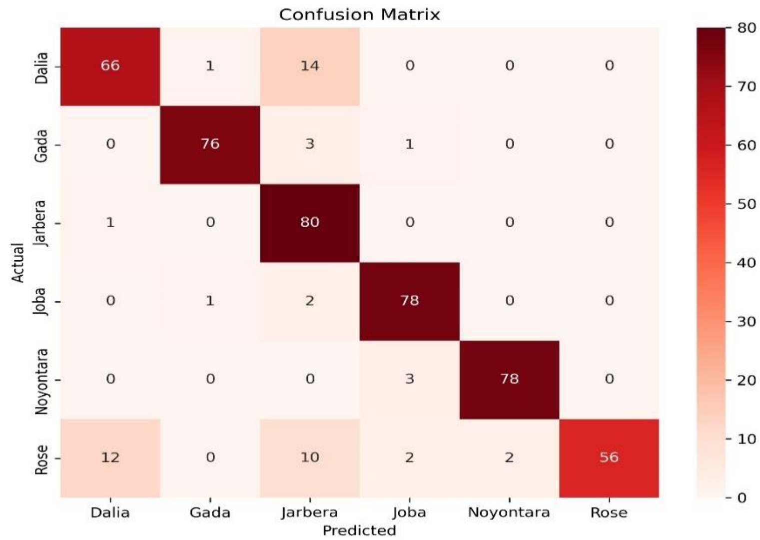

Fig. 4.2.2: MobileNet model confusion matrix

Plot Diagram:

Accuracy of training and validation

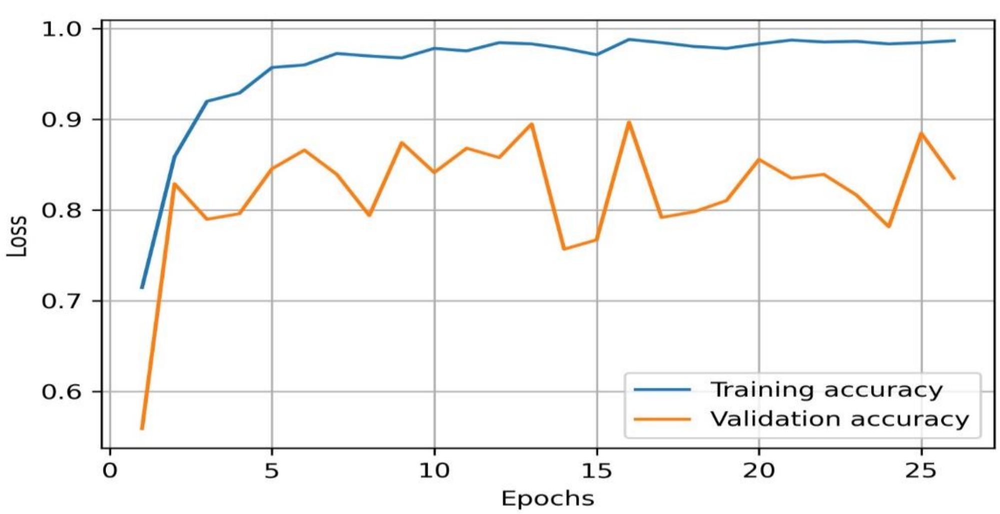

Fig. 4.2.3: Accuracy of training and validation for mobileNet model

Loss graph for training and validation

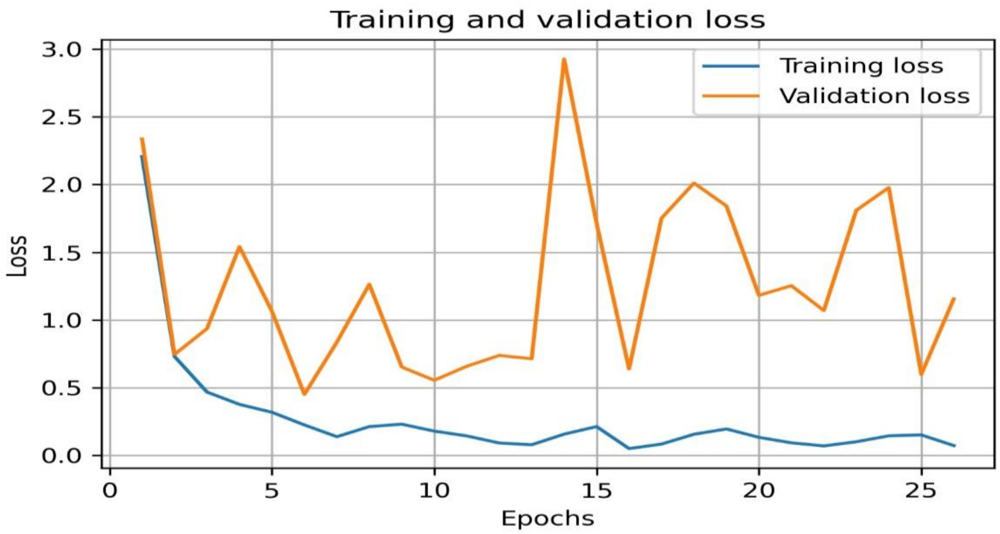

Fig. 4.2.4: Train and validation losses for mobileNet model

4.3 Results of experiments and analysis for the ResNet-50 model

Epoch-wise accuracy levels

In this case we got same level of accuracy in every epoch.

```
Epoch 00003: val_accuracy did not improve from 0.16495
Epoch 4/50
45/45 [==============================] - 16s 349ms/step - loss: 0.8923 - accuracy: 0.8017 - val_loss: 1.8275 - val_accuracy: 0.1649

Epoch 00004: val_accuracy did not improve from 0.16495
Epoch 5/50
45/45 [==============================] - 16s 350ms/step - loss: 0.6883 - accuracy: 0.8235 - val_loss: 2.2314 - val_accuracy: 0.1649

Epoch 00005: val_accuracy did not improve from 0.16495
Epoch 6/50
45/45 [==============================] - 16s 352ms/step - loss: 0.6705 - accuracy: 0.8129 - val_loss: 4.2697 - val_accuracy: 0.1649

Epoch 00006: val_accuracy did not improve from 0.16495
Epoch 7/50
45/45 [==============================] - 16s 357ms/step - loss: 0.6610 - accuracy: 0.8495 - val_loss: 2.1939 - val_accuracy: 0.1649

Epoch 00007: val_accuracy did not improve from 0.16495
Epoch 8/50
45/45 [==============================] - 17s 374ms/step - loss: 0.6025 - accuracy: 0.8523 - val_loss: 2.1824 - val_accuracy: 0.1649

Epoch 00008: val_accuracy did not improve from 0.16495
Epoch 9/50
45/45 [==============================] - 17s 385ms/step - loss: 0.4322 - accuracy: 0.8488 - val_loss: 2.1562 - val_accuracy: 0.1649
```

Fig. 4.3.1: ResNet-50 model accuracy

Confusion matrix

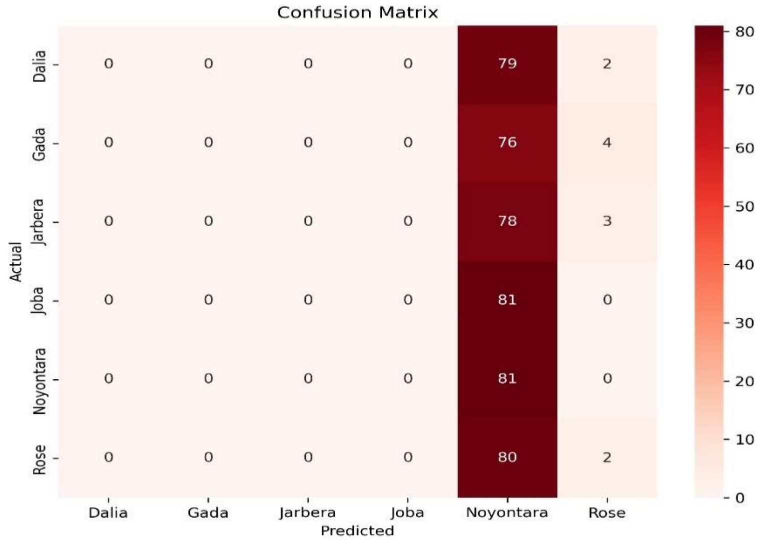

Fig. 4.3.2: ResNet-50 model confusion matrix

Plot diagram

Accuracy of training and validation

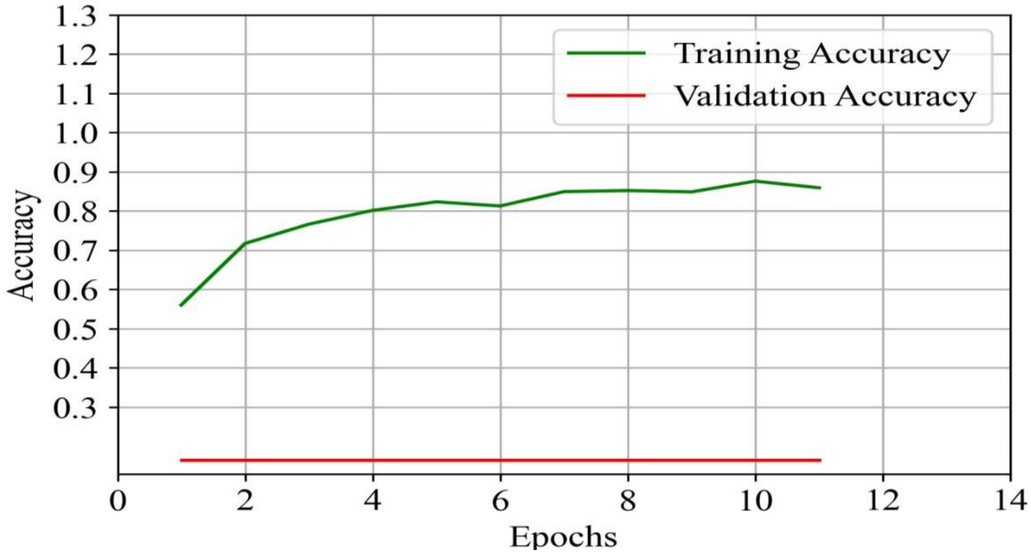

Fig. 4.3.3: Accuracy of training and validation for ResNet-50 model

Loss graph for training and validation

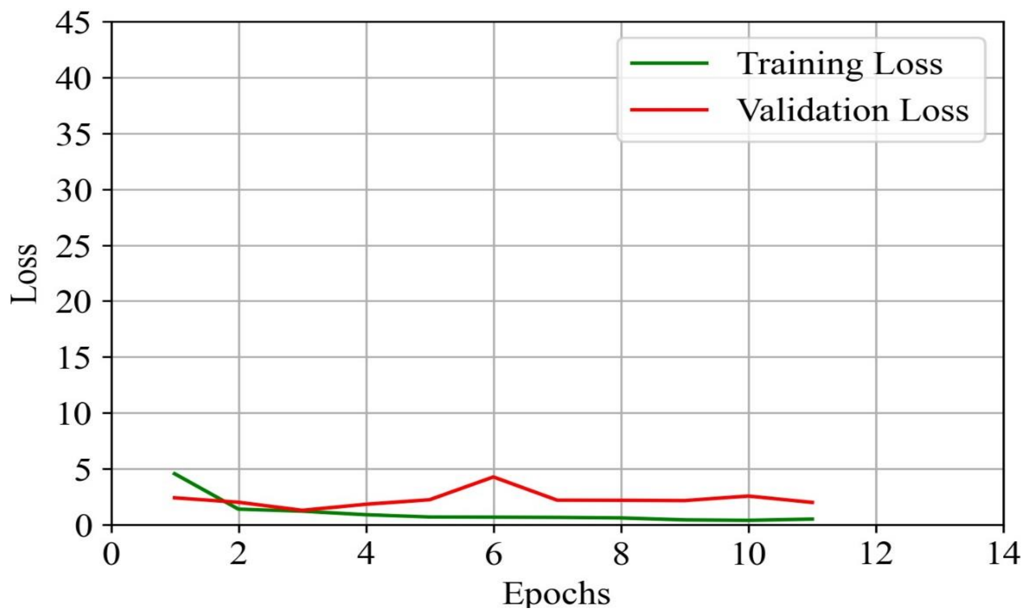

Fig. 4.3.4: Train and validation losses for ResNet-50 model

4.4 Results of Experiments and Analysis for the M-1 Model

Accuracy levels measured for each epoch

In the epoch number 41 I got best accuracy of 86%.

```
Epoch 00035: val_accuracy did not improve from 0.84948
Epoch 36/50
45/45 [==============================] - 13s 294ms/step - loss: 0.0458 - accuracy: 0.9866 - val_loss: 0.0095 - val_accuracy: 0.8392

Epoch 00036: val_accuracy did not improve from 0.84948
Epoch 37/50
45/45 [==============================] - 13s 295ms/step - loss: 0.0527 - accuracy: 0.9880 - val_loss: 0.0270 - val_accuracy: 0.8351

Epoch 00037: val_accuracy did not improve from 0.84948
Epoch 38/50
45/45 [==============================] - 13s 294ms/step - loss: 0.0467 - accuracy: 0.9873 - val_loss: 0.5095 - val_accuracy: 0.8536

Epoch 00038: val_accuracy improved from 0.84948 to 0.85361, saving model to model_1.h5
Epoch 39/50
45/45 [==============================] - 13s 296ms/step - loss: 0.0335 - accuracy: 0.9930 - val_loss: 0.5389 - val_accuracy: 0.8330

Epoch 00039: val_accuracy did not improve from 0.85361
Epoch 40/50
45/45 [==============================] - 13s 295ms/step - loss: 0.0313 - accuracy: 0.9923 - val_loss: 0.0057 - val_accuracy: 0.8454

Epoch 00040: val_accuracy did not improve from 0.85361
Epoch 41/50
45/45 [==============================] - 13s 295ms/step - loss: 0.0474 - accuracy: 0.9866 - val_loss: 0.0026 - val_accuracy: 0.8619
```

Fig. 4.4.1: M1 model accuracy

Confusion matrix

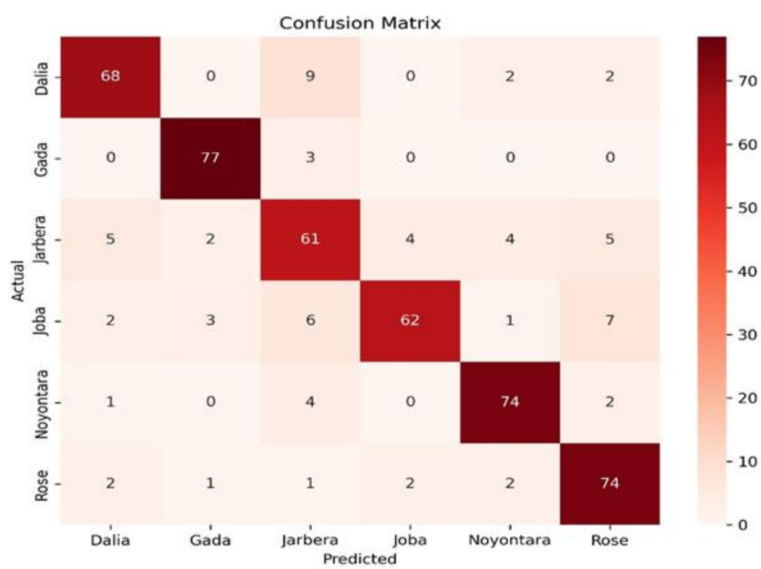

Fig. 4.4.2: M1 model's confusion matrix

Plot Diagram

Train and validation accuracy

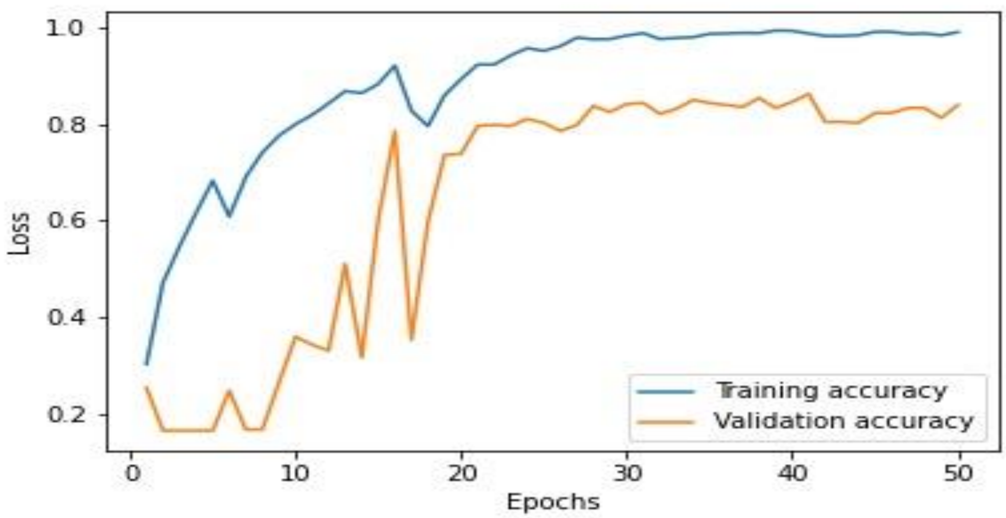

Fig. 4.4.3: M-1 model's training and validation accuracy.

Loss graph for training and validation

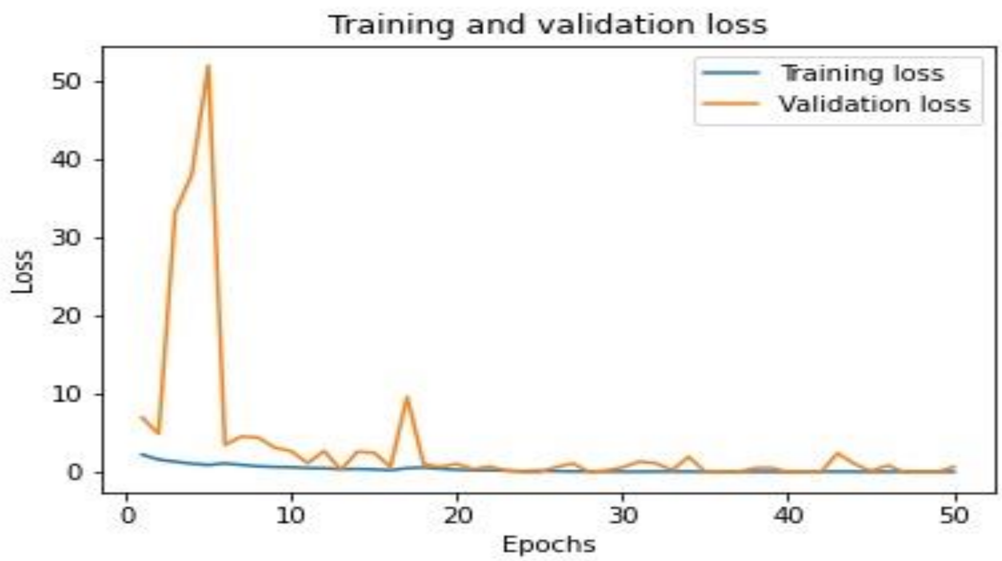

Fig. 4.4.4: Train and validation losses for M-1 model

4.5 Results of experiments and analysis for the VGG-16 model

Epoch-wise accuracy levels

In the epoch number 27 I got best accuracy of 97.53%.

```
45/45 [==============================] - 23s 509ms/step - loss: 0.0528 - accuracy: 0.9831 - val_loss: 0.1318 - val_accuracy: 0.9691
Epoch 24/50
44/45 [============================>.] - ETA: 0s - loss: 0.0309 - accuracy: 0.9885
Epoch 00024: val_accuracy did not improve from 0.97732
45/45 [==============================] - 23s 506ms/step - loss: 0.0310 - accuracy: 0.9887 - val_loss: 0.2241 - val_accuracy: 0.9546
Epoch 25/50
44/45 [============================>.] - ETA: 0s - loss: 0.0164 - accuracy: 0.9950
Epoch 00025: val_accuracy did not improve from 0.97732
45/45 [==============================] - 23s 501ms/step - loss: 0.0167 - accuracy: 0.9951 - val_loss: 0.2443 - val_accuracy: 0.9649
Epoch 26/50
44/45 [============================>.] - ETA: 0s - loss: 0.0155 - accuracy: 0.9957
Epoch 00026: val_accuracy did not improve from 0.97732
45/45 [==============================] - 23s 502ms/step - loss: 0.0179 - accuracy: 0.9951 - val_loss: 0.2780 - val_accuracy: 0.9670
Epoch 27/50
44/45 [============================>.] - ETA: 0s - loss: 0.0242 - accuracy: 0.9935
Epoch 00027: val_accuracy did not improve from 0.97732
45/45 [==============================] - 22s 498ms/step - loss: 0.0236 - accuracy: 0.9937 - val_loss: 0.1985 - val_accuracy: 0.9753
Epoch 28/50
44/45 [============================>.] - ETA: 0s - loss: 0.0245 - accuracy: 0.9914
Epoch 00028: val_accuracy did not improve from 0.97732
45/45 [==============================] - 23s 501ms/step - loss: 0.0256 - accuracy: 0.9909 - val_loss: 0.2379 - val_accuracy: 0.9691
Epoch 29/50
44/45 [============================>.] - ETA: 0s - loss: 0.0342 - accuracy: 0.9892
```

Fig. 4.5.1: VGG-16 model accuracy

Confusion matrix

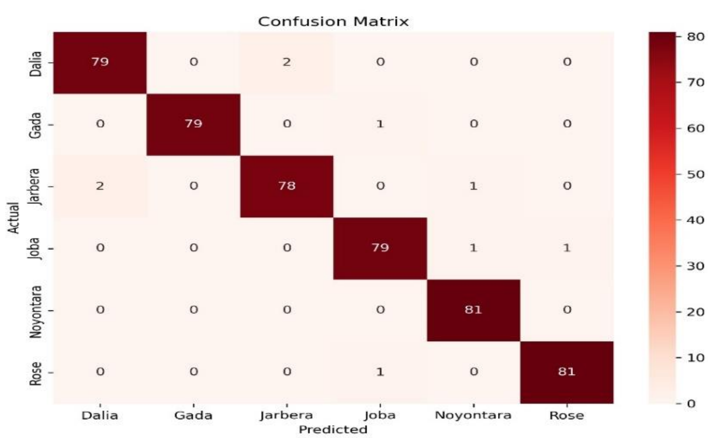

Fig. 4.5.2: VGG-16 model's confusion matrix

Plot diagram

Accuracy of training and validation

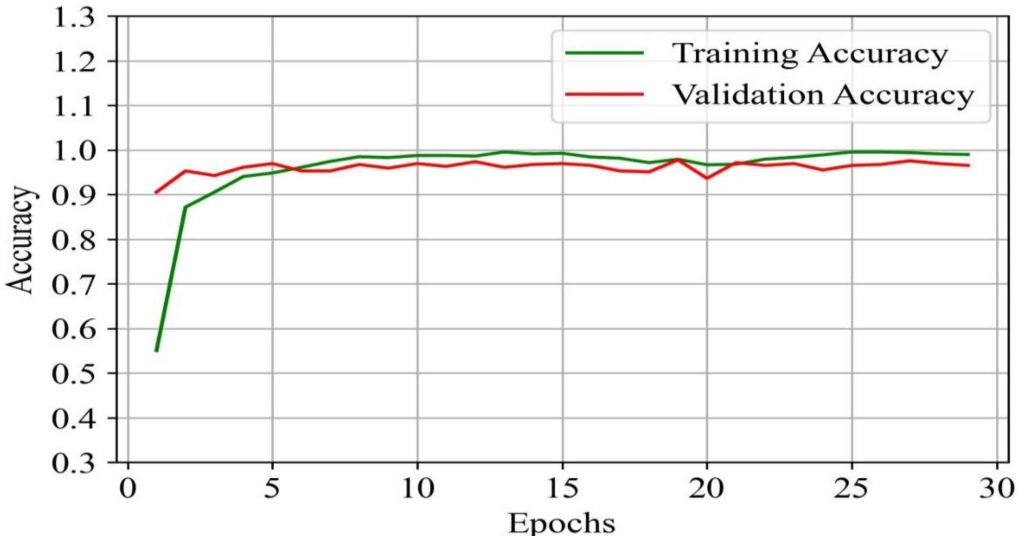

Fig 4.5.3: VGG-16 model's training and validation accuracy.

Loss graph for training and validation

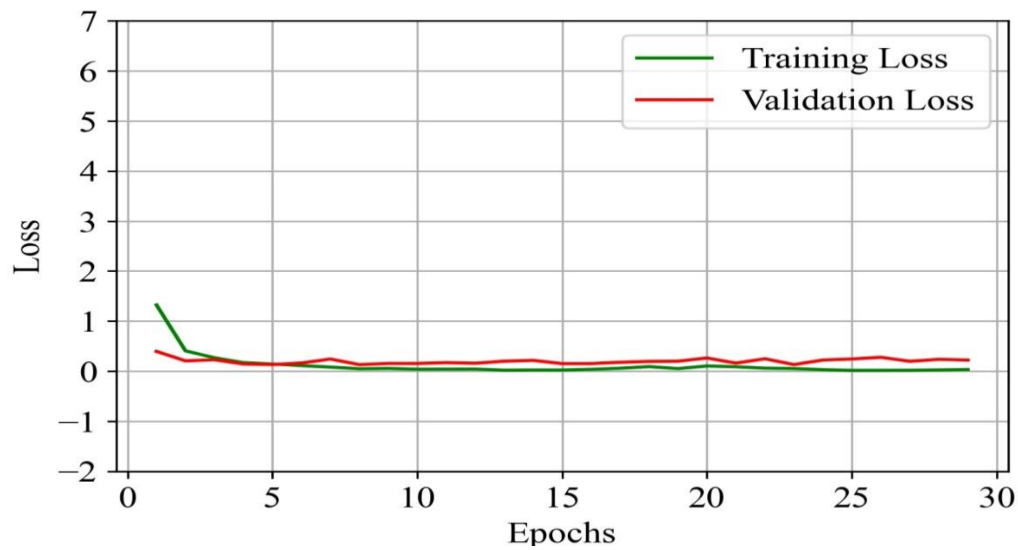

Fig. 4.5.4: Train and validation losses for VGG-16 model

4.6 Experiments

I have categorized several local flowers from Bangladesh in this project. I concentrated on the degree of accuracy. The quantity of data affects the accuracy rate. For each flower, I used 405 images, and my entire collection included 2450 images. I employed four different Deep learning model to determine accuracy. My best test accuracy with the VGG-16 model was 97.73%. Using the ResNet-50 model, my test accuracy dropped to 16%. As far as I'm aware, the quantity of data determines accuracy. My accuracy level would be higher if I collected more data and carefully preprocessed it beforehand.

CHAPTER 5

CONCLUSION AND FUTURE RESEARCH

5.1 Conclusion

The objective of this study was to evaluate the effectiveness of various deep learning models in the identification of local flower. The models tested were VGG-16, MobileNet, M-1, and ResNet-50. Using a large dataset of 2450 photos of several flower species, we sought to establish which model delivers the best accurate and efficient detection. Our results showed that the VGG-16 model had the highest accuracy, at 97%. This demonstrates the model's stability and applicability for precise image classification tasks like flower recognition, despite its increased processing requirements. Finally, our findings show that deep learning models are highly successful for local flower detection, with VGG-16 leading in accuracy.

5.2 Future Research

There is a lot of room for future development with this project. We don't develop any software that can identify flowers; instead, our suggested method only sorts flowers into groups. With the camera on a phone, it would be possible to make an Android app that can spot flowers in real time. In the future, additional attributes such as size and texture of the flowers will be taken into account for classification. It may be worth of exploring alternative classification techniques to enhance precision. More information should be stored in the database by expanding it. We only use six flowers in our system. In order to improve our dataset, more flower-related items can be included.

References:

[1] Hiary, Hazem, Heba Saadeh, Maha Saadeh, and Mohammad Yaqub. "Flower classification using deep convolutional neural networks." IET computer vision 12, no. 6 (2018): 855-862.

[2] Liu, Y., Tang, F., Zhou, D., Meng, Y. and Dong, W., 2016, November. Flower classification via convolutional neural network. In *2016 IEEE International Conference on Functional-Structural Plant Growth Modeling, Simulation, Visualization and Applications (FSPMA)* (pp. 110-116). IEEE.

[3] Mete, B.R. and Ensari, T., 2019, October. Flower Classification with Deep CNN and Machine Learning Algorithms. In *2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)* (pp. 1-5). IEEE.

[4] Prasad, M.V.D., Lakshmamma, B.J., Chandana, A.H., Komali, K., Manoja, M.V.N., Kumar, P.R. and Kiran, P.S., 2018. An efficient classification of flower images with convolutional neural networks. *International Journal of Engineering & Technology*, *7*(1.1), pp.384-391.

[5] Avishiktha Lodh, Ranjan Parekh,"Flower Recognition System based on Color and Gist Features", IEEE 2017.

[6] Busra Rumeysa Mete, Tolga Ensari,"Flower Classification with Deep CNN and Machine Learning Algorithms, IEEE 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies, 2019.

[7] U. C. Dey, F. Hasan, M. H. R. Papel, T. H. Turza, M. M. Hasan and M. R. Kabir, "Local Fruit Classification and Recognition using Transfer Learning Models," 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), Delhi, India, 2023, pp. 1-5.

[8] Shone, Nathan, Tran Nguyen Ngoc, Vu Dinh Phai, and Qi Shi. "A deep learning approach to network intrusion detection." IEEE transactions on emerging topics in computational intelligence 2, no. 1 (2018): 41-50.

[9] Stokes, Jonathan M., Kevin Yang, Kyle Swanson, Wengong Jin, Andres Cubillos-Ruiz, Nina M. Donghia, Craig R. MacNair et al. "A deep learning approach to antibiotic discovery." Cell 180, no. 4 (2020): 688-702.

[10] Javaid, Ahmad, Quamar Niyaz, Weiqing Sun, and Mansoor Alam. "A deep learning approach for network intrusion detection system." In Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS), pp. 21-26. 2016.

Local Flower Detection through Deep Learning Approach

ORIGINALITY REPORT

19%	18%	1%	10%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	9%
2	Submitted to Daffodil International University Student Paper	2%
3	123dok.com Internet Source	1%
4	Submitted to University of Wales Institute, Cardiff Student Paper	1%
5	soar.wichita.edu Internet Source	<1%
6	etheses.lib.ntust.edu.tw Internet Source	<1%
7	Submitted to National College of Ireland Student Paper	<1%