Flower Image Classification

A. Introduction and Problem Statement

The classification of flowers is a significant topic due to their widespread existence and impact on the world. Flowers grow in a range of habitats and climates and play a crucial role in the food chain by feeding various insect species. They also have medicinal properties and can be used to produce drugs. Thus, it is crucial to have a proper understanding of different flower species to avoid any damage or misidentification, as well as to appreciate their true value. Knowledge in this field will also help in the cultivation of plants and support the growth of the pharmaceutical industry, especially for rare and endemic plant species [5].

The main challenge for flower classification is the high variability and complexity of flower images. There is a wide range of variations in flower color, shape, size, texture, and background. In addition, flowers can appear in different stages of growth and may be viewed from different angles or under varying lighting conditions.

By leveraging Deep Learning models, the primary goal is to achieve precise identification and classification of diverse flower species from images. The establishment of effective and dependable flower detection models can automate tasks that formerly depended on human intervention, potentially resulting in time and resource savings while enhancing precision and consistency. Expectation throughout the project is to analyze and compare different models and find the best fit for the flower classification which can later be used for different applications such as Pharmaceutical applications.

B. Dataset Selection:

Name	Total Images	Classes
Flowers Dataset I	5k	5
Flowers Dataset II	11.2k	7
Flowers Dataset III	15.7k	16

Dataset I: The dataset by utkarshsaxenadn on Kaggle contains flower images divided into 5 categories. It has 1000 images for testing and 5000 images for training with the same number of images(jpeg) with pixel size 225x225 in each category. [1].

Dataset II: A collection of 11200 images(jpg) of the pixel size 178x256 to 648x500 of flowers from 7 different classes —make up Nadyana's "Flowers" dataset on Kaggle which is a large enough number of images for training a deep learning model [2].

Dataset III: The dataset by "L3LLFF" on kaggle is the most versatile dataset of our selection which includes 15740 total images(jpg) of size 256x256 with 16 different classes. [3].

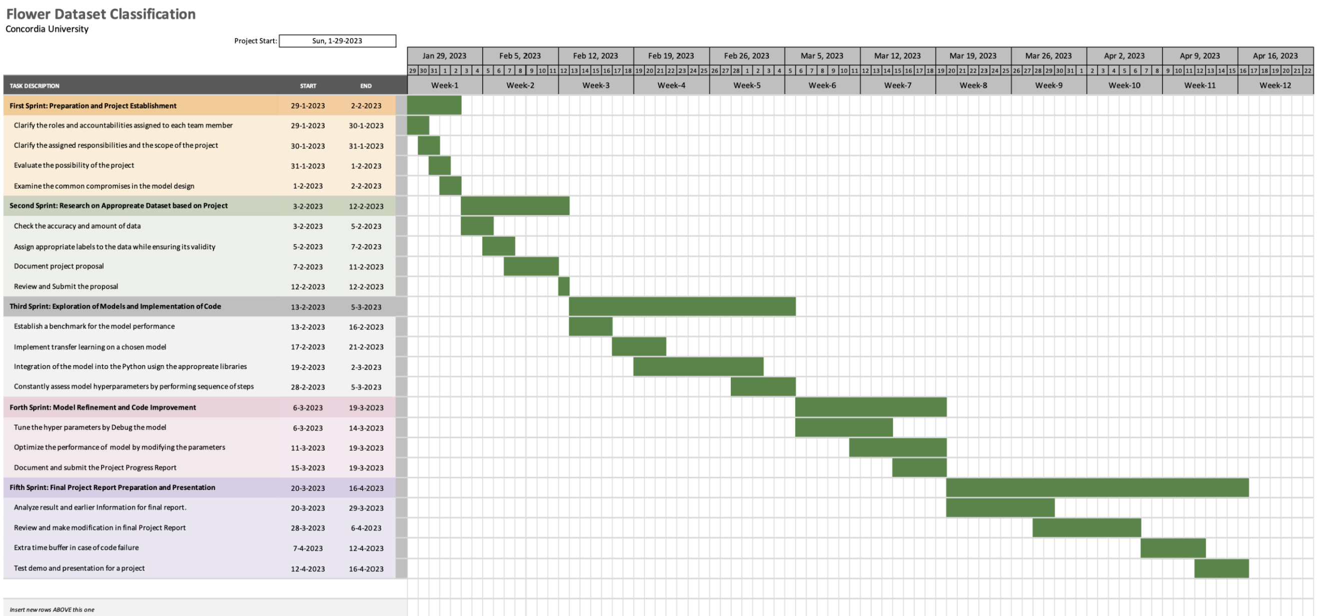
C. Possible Methodology

Steps for Pre-processing : Resize the images to the required input size for the model. Normalize the pixel values of the images. Apply data augmentation techniques such as random cropping, flipping, and rotation. Color space conversion: Images can be represented in different color spaces, such as RGB, HSV, or grayscale. Converting the image to a different color space can help improve the accuracy of the classifier, as some color spaces may be more suitable for certain types of images. One-hot encoding: The categorical labels of the flower images may need to be encoded as one-hot vectors, which can be used as targets for the model's output. Splitting the dataset: The dataset may be divided into training, validation, and testing sets, with each set serving a different purpose in model development and evaluation.

Model Development When it comes to flower image detection, ResNet18 [4] can be a good fit because it can effectively learn the complex features and patterns in the images while still being computationally efficient. MobileNetV2 [7] is well-suited for flower image detection because it balances accuracy with computational efficiency. It uses depthwise separable convolutions to reduce the number of parameters and computations, making it efficient to run on resource-constrained devices. At the same time, it has been shown to achieve good accuracy on various computer vision tasks, including image classification and object detection. VGGNet19 [6] uses a stack of multiple convolutional and max-pooling layers to extract increasingly complex features from the input image. 3 models will perform on 3 different datasets which means that a total of 9 different models will be generated. Moreover, we will use two of the trained models as base models on datasets except itself for transfer learning. Some of the parameters like Learning rate, Batch sizes, and the number of training epochs while model training decreases coverage loss. We will use the above hyperparameters with the help of Hypothesis testing to optimize.

Assessment of the Model Metrics including Accuracy, Precision, Recall, and F1 Score are employed in order to evaluate our classification model. The confusion matrix can also be shown to assess the model's effectiveness. The ROC curve evaluates the binary classifier and flower image classification performance. A labeled dataset should be divided into training and testing sets. TPR and FPR values are obtained from predictions and actual labels at different thresholds, plotted on the ROC curve, and the AUC is calculated to measure the model's performance. The closer the AUC is to 1, the better the model performs.

D. Gantt Chart



E. Project Milestone

1. Goal I (Completion of First Sprint: Preparation and Project Establishment.): In this sprint, we meticulously examine the complexity of the problem statement and identify all necessary components. Upon completion of this stage, we finalize our models, and datasets, and allocate tasks to the teams.

2. Goal II (Completion of Second Sprint: Research on Appropriate Dataset based on Project): This is a vital process where we carefully assess the datasets, evaluating their image quality and quantity. The result of this stage will be a detailed project proposal that will be improved based on the feedback received from the team and the professor.

3. Goal III (Third Sprint: Exploration of Models and Implementation of Code): In the Model Exploration Sprint, we will commence with the implementation process that involves scripting in Python for all the models. The final versions of these models will then be forwarded to the subsequent Sprint for further optimization. However, this Sprint will furnish us with functional models that have been put to the test, along with adjusted parameters to ensure accurate results.

4. Goal IV (Forth Sprint: Code improvement and model refinement): Currently, we have a functioning model and we will tweak the hyper-parameters to enhance its performance. We will keep a close eye on all evaluation metrics and make necessary adjustments to the parameters. Afterward, we will compile a progress report using this information and present it for examination.

5. Goal V (Fifth Sprint: Final Project Report Preparation and Presentation): At this stage, a comprehensive analysis and the final report and presentation for the project will be created and implemented.

F. Project Deliverables

1. First Sprint: Preparation and Project Establishment: Outline team member responsibilities, assess project viability and analyze model trade-offs.

2. Second Sprint: Research on Appropriate Dataset based on Project, submit a comprehensive project proposal.

3. Third Sprint: Working model

4. Forth Sprint: Code improvement and model refinement and submit the Project Progress Report.

5. Fifth Sprint: Final Project Report Preparation and Presentation

References

[1] Flower classification — 10 classes —. https://www.kaggle.com/datasets/utkarshsaxenadn/flower-classification-5-classes-roselilyetc.

[2] Flowers. https://www.kaggle.com/datasets/nadyana/flowers.

[3] Flowers: A collection of images for flower classification. https://www.kaggle.com/datasets/l3llff/flowers.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[5] Büşra Rümeysa Mete and Tolga Ensari. Flower classification with deep cnn and machine learning algorithms. In *2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, pages 1–5. IEEE, 2019.

[6] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.

[7] Debjyoti Sinha and Mohamed El-Sharkawy. Thin mobilenet: An enhanced mobilenet architecture. In *2019 IEEE 10th annual ubiquitous computing, electronics & mobile communication conference (UEMCON)*, pages 0280–0285. IEEE, 2019.