BIRZEIT UNIVERSITY

Faculty of Engineering and Technology Electrical and Computer Engineering Department

Artificial Intelligence

ENEE 4113

Project No. 2

Image classification

Prepared by:

Sawsana Atari **1221320**

Sara Al-Souqi **1221618**

Instructor: **Dr. Samah Alaydi**

Section: 2

Date: . June. 2025

Abstract

We present a comparative study of three machine-learning classifiers for flower image recognition, implemented end-to-end on Google Colab. First, reducing the original Flowers Recognition dataset from 7,000 to 1,220 images and then re-sizing each to 224 × 224 pixels using the ImageMagick tool. Handcrafted feature extraction was then applied: Gaussian Naïve Bayes relies on basic color-channel statistics (mean, standard deviation, and selected percentiles), while the Decision Tree model uses raw pixel-intensity values to learn hierarchical splits. Finally, the Multilayer Perceptron (MLP) employs two hidden layers with tunable hyperparameters (activation function, learning rate, and neurons per layer) to maximize predictive performance.

We evaluate all models on an 80/20 train–test split and report accuracy, precision, recall, and F1-score. Our results demonstrate that the MLP achieves the highest overall accuracy, whereas Gaussian Naïve Bayes provides a fast, low-complexity baseline. The Decision Tree classifier offers a middle ground—delivering moderate accuracy and full interpretability with reasonable computational cost. This work highlights the trade-offs between preprocessing choices, model complexity, and computational cost in resource-constrained image-classification settings.

Table of Contents

Table of Figures

1. Introduction and dataset description

This project measures the performance of three machine learning algorithms—Naive Bayes, Decision Tree, and Feedforward Neural Networks—for classifying images into multiple categories. The motivation is to understand how well each model acts when given simple pixel-based features, and to evaluate results we end with by measures such as accuracy and confusion matrix.

The dataset consists of 1,220 images of flowers, distributed among four categories: *daisy*(300 images), *rose*(300 images), *sunflower*(235 images), and *tulip*(385 images). Each image was resized to 224×224 pixels and stored in class-named folders. All image files share the same extension (*.jpg*). The file paths and labels were automatically extracted using Python scripts, and the data was preprocessed to ensure compatibility with machine learning classifiers.

2. Theory

2.1 Naive Bayes Classifier

The Naive Bayes classifier is a one of probabilistic model. This model build based on Bayes' Theorem, where the features used for classification are considered to be independent given the class label. In the context of image classification, this means that each feature (such as statistical summary) is treated as contributing independently to the probability of an image belonging to a specific class.

In this project, we used the Gaussian Naive Bayes variant, which models the distribution of each feature as a Gaussian (normal) distribution. For each sample (image), we extracted its simple statistical features: the mean, standard deviation, and median of the red, green, and blue channels. These features provide a compact summary of the color distribution of the image, making them suitable for this model.

Naive Bayes shows efficient and performs well on high-dimensional data. However, its independence assumption would limit its performance when features are correlated, as is often the case in image data. Despite its simplicity, it provides a good baseline for comparing with more complex models.

2.2 Decision Tree Classifier

The Decision Tree classifier in this project was trained using the same feature extraction method as the Feedforward Neural Network. Each image was resized and processed to extract 16-bin normalized histograms for the red, green, and blue channels, along with the mean and standard deviation of each channel. This combination of color distribution and statistical features provides a compact yet informative representation of the image.

The model was implemented using `DecisionTreeClassifier ` from Scikit-learn with default parameters. After extracting features, the dataset was split into training and testing sets, and the model was trained to classify images into flower categories.

Decision Trees are fast and interpretable, making them a good starting point for evaluating model performance. However, they are prone to overfitting, especially when the tree grows deep without pruning. Despite this limitation, the Decision Tree provides a useful baseline for comparison with more advanced models like MLP.

2.3 Feedforward Neural Network (MLP)

The Feedforward Neural Network, implemented in this project using the `MLPClassifier` from Scikit-learn, is a type of artificial neural network that maps input features to output classes through one or more hidden layers. MLPs is a complex ANN which is used to solve ,in addition to simple problem, the complex one.

In our setup, the network architecture included two hidden layers with 100 and 50 neurons, respectively. The activation function used was the logistic sigmoid (`activation='logistic'`), which was found to outperform ReLU and tanh in our case. This may be attributed to the nature of our input features—statistical and histogram-based summaries of image color channels—that were bounded and relatively low in dimensionality.

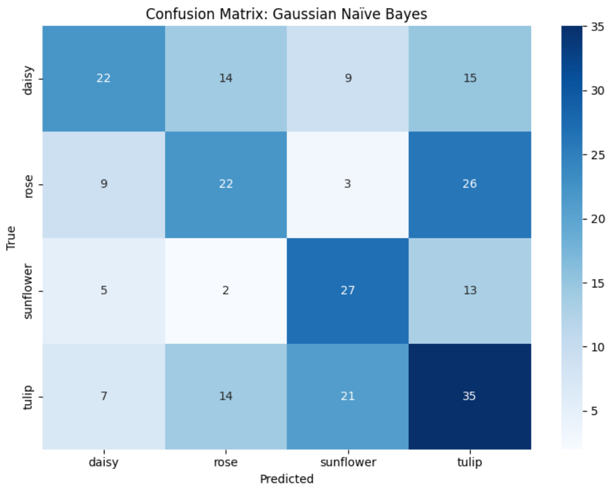
The input to the MLP consisted of advanced features extracted from each image: 16-bin normalized histograms for each of the R, G, and B channels, as well as the mean and standard deviation per channel. These features produce a richer representation than simple pixel values, enabling the MLP to learn fine distinctions between flower classes.

The model was trained using a learning rate of 0.001, which yielded the highest accuracy (≈53.7%) after experimenting with multiple values. Optimization was performed through backpropagation, as handled internally by the MLPClassifier. The output layer applies the softmax function to manage the multiclass classification task, and predictions are made by selecting the class with the highest probability score (argmax of the softmax output).

While training required more time compared to Naive Bayes and Decision Tree, the MLP achieved the highest classification accuracy among the three models, demonstrating its strength in capturing complex feature interactions.

3. Procedure

3.1 Naive Bayes Classifier



	precision	recall	f1-score	support
daisy	0.51	0.37	0.43	60
rose	0.42	0.37	0.39	60
sunflower	0.45	0.57	0.50	47
tulip	0.39	0.45	0.42	77
accuracy			0.43	244

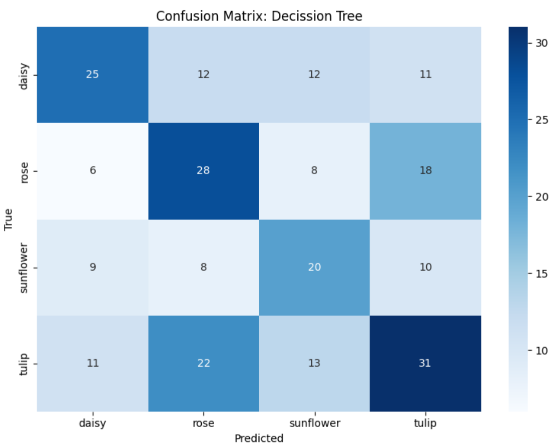
Figure 1: Confusion matrix and accuracy metrics for the Naive Bayes classifier

As shown in the classification report and confusion matrix above, the Naive Bayes classifier achieved an overall accuracy of 43%. The model struggled especially with the daisy and rose classes, which had low recall values of 0.37.

From the confusion matrix, we can observe that:

- ✧ Many daisy images were misclassified as rose or tulip.
- ✧ The sunflower class performed the best, with 27 out of 47 correctly classified (≈57% recall).
- ✧ There was a significant confusion between rose and tulip, which explains the model's poor performance on those categories.
- ✧ These results suggest that the Naive Bayes model, while simple and fast, is limited by its strong independence assumption — especially in image data where pixel/color features are often correlated.

3.2 Decision Tree Classifier



Decission Tree Performance:

	precision	recall	f1-score	support
daisy	0.49	0.42	0.45	60
rose	0.40	0.47	0.43	60
sunflower	0.38	0.43	0.40	47
tulip	0.44	0.40	0.42	77
accuracy			0.43	244

Figure 2: Confusion matrix and accuracy metrics for the Decision Tree classifier

- ✧ As shown in the classification report and confusion matrix above, the Decision Tree classifier achieved an overall accuracy of 43%, which is comparable to the Naive Bayes model. The performance was moderately balanced across the different flower classes.

- ✧ The daisy class had a precision of 0.49 and recall of 0.42, with 25 out of 60 images correctly classified. However, some were misclassified as rose and sunflower.

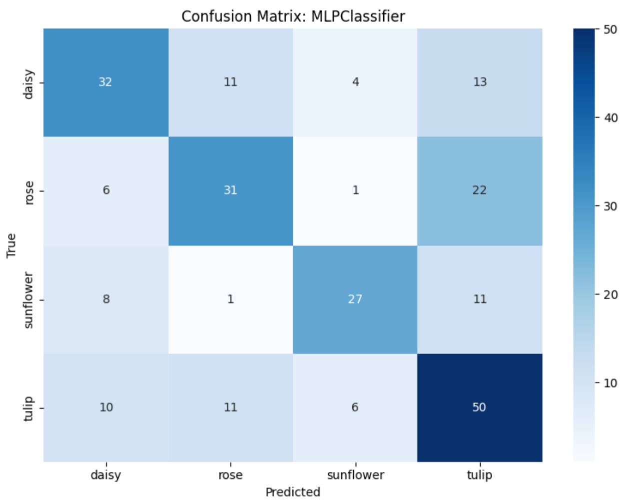
- ✧ The rose class showed slightly better recall at 0.47, with 28 images correctly identified, although 18 images were confused with tulips.

- ✧ The sunflower class, despite being smaller in size (47 images), had 20 correct predictions, but was also confused with other categories.

- ✧ The tulip class achieved the highest number of correct classifications (31 out of 77), though it was frequently misclassified as rose and sunflower.

- ✧ The confusion matrix indicates notable overlaps between tulip and rose, and between daisy and sunflower. Unlike the MLP model, the Decision Tree struggles with complex class boundaries, which affects its ability to separate similar classes like rose and tulip effectively.

3.3 Feedforward Neural Network (MLP)



	precision	recall	f1-score	support
daisy	0.57	0.53	0.55	60
rose	0.57	0.52	0.54	60
sunflower	0.71	0.57	0.64	47
tulip	0.52	0.65	0.58	77
accuracy			0.57	244

Figure 3: Confusion matrix and accuracy metrics for the MLP classifier

- ✧ As shown in the classification report and confusion matrix above, the MLPClassifier achieved an overall accuracy of 57%, outperforming the Naive Bayes classifier. The performance was relatively balanced across the different flower classes.

- ✧ The model achieved the highest precision (0.71) for the sunflower class, with 27 out of 47 images correctly classified — despite sunflower having the smallest number of samples in the test set.

- ✧ The tulip class also showed strong performance, with a recall of 0.65, indicating that most tulip images were correctly identified.

- ✧ The daisy and rose classes saw moderate improvements compared to the Naive Bayes model, both achieving over 50% recall. These two categories were particularly challenging in the previous model, where Naive Bayes frequently confused them with each other. The MLP model was better able to distinguish between them, since to its capacity to learn more complex feature interactions.

- ✧ The confusion matrix shows fewer severe misclassifications than in the previous models. For example, the model correctly classified 50 tulip images and 31 rose images. However, some confusion still exists between daisy and tulip, and between rose and tulip.

Overall, the MLP model demonstrated better generalization and class separation, benefiting from the richer feature set and its ability to model nonlinear relationships.

Conclusion

Based on our observations during training and evaluation, the following conclusions can be drawn:

- If the data/images were more equally distributed across all classes, the evaluation results might have been better.

- A larger dataset would also lead to improved results, since outliers are more likely to appear in small datasets. With more balanced and cleaner data, both the accuracy and the confusion matrix would look better.

- The dataset contains high similarities between certain categories—especially between tulip and rose. This caused the Naive Bayes and Decision Tree classifiers to struggle with distinguishing between them, while the MLP model handled these cases better. This highlights the limitations of simpler models when dealing with visually similar classes.

- The performance of the MLP model could be improved further by fine-tuning parameters like the number of epochs and batch size, whereas improvements for Naive Bayes and Decision Tree would require different approaches such as feature engineering, pruning, or ensemble methods.

References

DATA + colab on drive : https://drive.google.com/drive/folders/1vgb2I8IVW6Xp4Mwi505qMi7vufZTtjgP?usp=drive_link