Flower Classification Using Convolutional Neural Networks for Daisies, Sunflowers, and Roses

Ahmet Selim Özdemir
040200038
ozdemirah20@itu.edu.tr

Batıkan Ciğal
040190007
cigal19@itu.edu.tr

Kerem Alkaya
040210002
alkayak21@itu.edu.tr

***Abstract*—The burgeoning demand for sophisticated automated plant identification systems has propelled flower classification using advanced deep learning techniques to the forefront of contemporary research. This paper meticulously details a robust and practical approach to classifying three distinct and commonly encountered flower species: daisies, sunflowers, and roses. The core of our methodology revolves around a carefully designed Convolutional Neural Network (CNN) architecture, engineered to efficiently discern subtle visual differences inherent to these floral categories. The proposed CNN model is strategically composed of three sequential convolutional layers; each thoughtfully paired with a max pooling layer to progressively reduce spatial dimensions while preserving salient features. These feature extraction layers culminate in a dense, fully connected layer, which then feeds into a softmax output layer responsible for generating probabilistic predictions for each class.**

For training and validation, the model was meticulously trained on a relatively small, yet carefully balanced dataset comprising 250 images, ensuring equitable representation across all three flower species. This dataset was systematically partitioned into an 80/20 split for training and validation, respectively, a standard practice to assess the model's generalization capabilities. Despite the inherent limitations imposed by the dataset's size, the model exhibited impressive classification accuracy during its early training epochs, indicating an initial phase of highly effective feature learning. However, a detailed analysis revealed clear signs of overfitting emerging beyond the third epoch, manifesting as a growing divergence between training and validation performance.

Our comprehensive analytical approach included visual evaluations, primarily facilitated by the generation and interpretation of confusion matrices, which provided granular insights into class-specific prediction accuracies and common misclassifications. Furthermore, detailed line plots illustrating the progression of both training and validation accuracy and loss over epochs were utilized to meticulously monitor the learning process, identify optimal training durations, and detect patterns indicative of overfitting and convergence. The paper concludes with an in-depth discussion of a spectrum of potential improvements aimed at enhancing the model's robustness, generalizability, and overall performance. These recommendations encompass crucial techniques such as data augmentation to expand the effective dataset size, early stopping to prevent excessive training and overfitting, and the leveraging of pretrained models (transfer learning) to capitalize on knowledge acquired from vast, diverse image datasets1. This study underscores the critical balance between model complexity, dataset size, and training duration for achieving effective deep learning solutions in real-world plant identification applications.

Keywords—Flower Classification, CNN, Deep Learning, Image Recognition, Adam Optimizer, Overfitting, Transfer Learning

I. Introduction

In the contemporary landscape of digital biology, agriculture, and environmental conservation, the accurate and efficient classification of floral species has emerged as a profoundly vital task. Traditionally, the identification of plant species has been a laborious and time-consuming endeavor, heavily reliant on manual inspection by botanical experts, thereby necessitating specialized knowledge and extensive field experience. Such conventional methodologies, while historically significant, are inherently limited in scalability and speed, posing significant challenges in large-scale biodiversity assessments, agricultural monitoring, and rapid ecological surveys.

However, the rapid and transformative advancements in the fields of computer vision and machine learning, particularly the advent and sophisticated application of Convolutional Neural Networks (CNNs), have revolutionized the landscape of automated classification. CNNs, a specialized class of deep neural networks, possess an unparalleled capacity to effectively process and interpret visual data. Their architectural design allows them to automatically learn intricate spatial hierarchies within images, ranging from low-level features such as edges and textures to high-level abstract representations. This inherent ability makes them exceptionally well-suited for distinguishing between various plant species based on nuanced visual characteristics, including subtle differences in petal shape, intricate textural patterns, and precise color variations. Consequently, CNNs have provided an unprecedented opportunity to automate and significantly accelerate the once arduous process of floral identification with impressive and continually improving accuracy.

This paper undertakes a comprehensive investigation into the practical implementation of a CNN-based classifier. The specific focus is on a curated subset of flower images representing three popular and visually distinct species: daisies, sunflowers, and roses. While these species are generally recognizable to the human eye, their classification by automated systems can present unique challenges. Overlapping visual traits, particularly under varying environmental conditions such as inconsistent lighting or diverse photographic angles, can introduce ambiguities that complicate precise differentiation. Therefore, a core objective of this research is to design and evaluate a lightweight yet demonstrably effective CNN model. This model is specifically engineered to exhibit strong generalization capabilities, even when trained on a relatively limited dataset. Ultimately, this work aims to establish a robust foundational framework for future advancements in automated floral classification, integrating and building upon modern machine learning practices to enhance accuracy, efficiency, and real-world applicability.

II. PRELIMINARIES

Before delving into the implementation details, it is important to define the key components and methods used in this model:

- **Convolutional Neural Network (CNN)**: A type of neural network especially effective for processing visual data. It employs convolutional layers that apply filters to input images to extract relevant features[1].
- **Adam Optimizer**: An optimization algorithm that adapts the learning rate during training. It combines the benefits of AdaGrad and RMSProp and is well-suited for problems with sparse gradients[2].
- **Training-Validation Split**: Dividing the dataset into 80% for training and 20% for validation helps assess the generalization performance of the model.

Confusion Matrix: A performance evaluation tool that shows the distribution of true positives, false positives, true negatives, and false negatives across multiple classes..

III. METHODOLOGY

A. Dataset Preparation

The dataset consists of 250 labeled images with equal representation for the three flower categories. Each image is resized to 180×180 pixels and normalized using a rescaling technique that converts pixel values from the range [0–255] to [0–1]. This preprocessing ensures that the data fed into the neural network is consistent and numerically stable. The dataset is randomly shuffled and divided into training and validation subsets in an 80/20 ratio.

B. Model Architecture

The network comprises several essential components:

- **Input Layer**: Incorporates a Rescaling layer that normalizes pixel values.
- **Convolutional Layers**: Three Conv2D layers with ReLU activations are used to capture local patterns such as edges, textures, and shapes.
- **Pooling Layers**: MaxPooling2D layers reduce spatial dimensions while retaining key features, lowering computation cost.
- **Flatten Layer**: Converts the final pooled feature maps into a one-dimensional vector suitable for fully connected layers.
- **Dense Layer**: A fully connected layer with 128 neurons and ReLU activation allows the model to learn high-level abstractions.
- **Output Layer**: A dense layer with 3 neurons (corresponding to the three flower classes), using softmax activation to generate class probabilities.

C. Training Configuration

The model is compiled using the categorical cross entropy loss function, which is appropriate for multi-class classification tasks. The Adam optimizer is used with default parameters, and training is performed over 3, 5, and 7 epochs to study the effect of training duration on model performance. ReLU activation is used in all hidden layers, while softmax is used in the output layer to ensure probabilistic output.

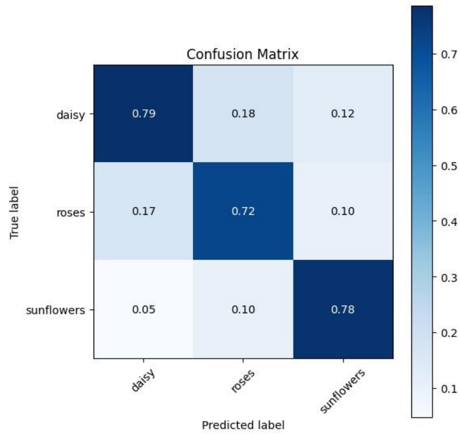
D. Evaluation Metrics and Visualization

To assess model performance, both numerical metrics and visual tools were utilized. Confusion matrices were generated after training with 3, 5, and 7 epochs to evaluate the accuracy per class. Additionally, line graphs plotting training and validation accuracy and loss were used to monitor overfitting and convergence patterns. These tools are critical for identifying the optimal number of training epochs.

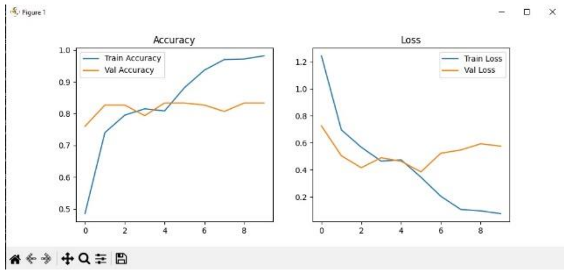
IV. RESULTS AND DISCUSSION

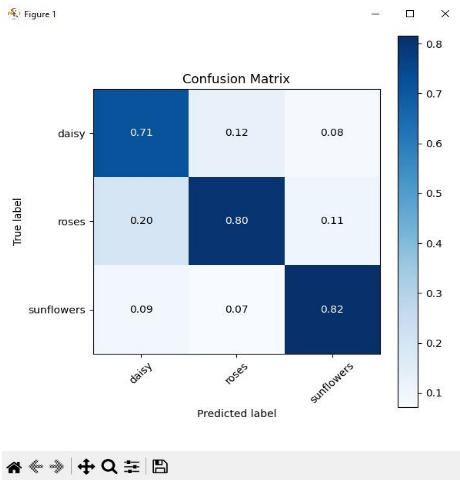
Training over multiple epochs provided insight into model learning behavior:

- After 3 epochs, the model showed a balance between training and validation accuracy, indicating initial effective learning.

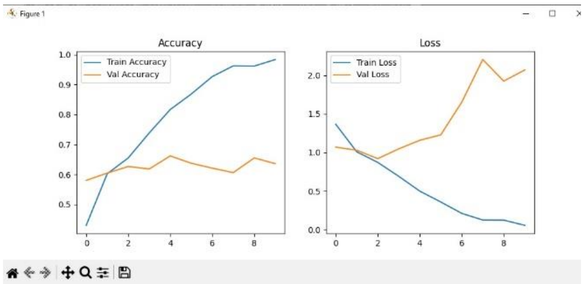


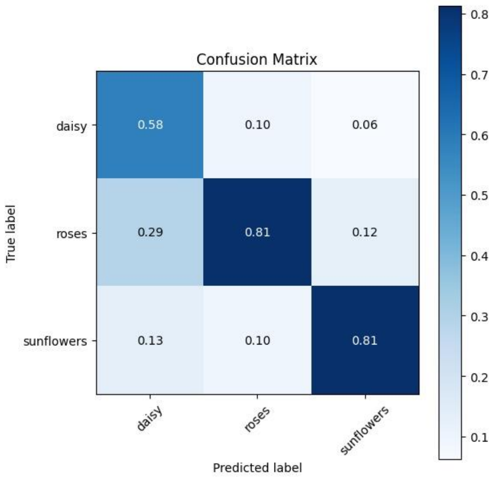
- At 5 epochs, accuracy improved slightly, but the gap between training and validation accuracy began to widen.





- By 7 epochs, clear signs of overfitting emerged, as validation loss increased while training loss continued to decrease.





The confusion matrices revealed that most misclassifications occurred between daisies and roses, likely due to shared color profiles and petal shapes in certain samples. Sunflowers, having distinct yellow petals and a large dark center, were more consistently recognized. Accuracy and loss plots confirmed that the model's best generalization occurred around 3–5 epochs, beyond which model performance on unseen data degraded. This suggests that the dataset size limits the model's capacity to learn generalized representations beyond a certain training threshold.

V. POSSIBLE IMPROVEMENTS

To enhance classification performance and mitigate overfitting, several strategies are proposed:

- **Data Augmentation**: Techniques like rotation, zoom, horizontal flipping, and brightness adjustments can artificially expand the training set.
- **Early Stopping**: Automatically halting training when validation loss stops improving helps avoid unnecessary overfitting.
- **Dropout Layers**: Adding dropout between dense layers can prevent co-adaptation of neurons, thereby improving generalization.
- **Learning Rate Scheduling**: Adjusting the learning rate during training can fine-tune the convergence process.
- **Transfer Learning**: Using pretrained models like MobileNetV2 or ResNet50 can boost performance with limited data by leveraging features learned on large datasets.
- **Larger Datasets and More Classes**: Adding more labeled data and expanding to more flower species can improve robustness[3].

VI. CONCLUSION

In this study, we designed and evaluated a CNN-based model for classifying daisies, sunflowers, and roses using a relatively small dataset. Despite computational constraints and data limitations, the model demonstrated effective classification performance with clear potential for real-world applications. Our findings emphasize the importance of training duration and the risk of overfitting with small datasets. With further optimization and additional data, this model can be integrated into mobile apps for real-time plant identification or deployed in agricultural monitoring systems.

REFERENCES

[1] IBM, "Convolutional Neural Networks," https://www.ibm.com/think/topics/convolutional-neural-networks

[2] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," arXiv preprint arXiv:1412.6980, Jan. 2017.

[3] M.-E. Nilsback and A. Zisserman, "Automated Flower Classification over a Large Number of Classes," ICVGIP, Dec. 2008.

[4] TensorFlow CNN Guide: https://www.tensorflow.org/tutorials/images/cnn

[5] PyTorch Transfer Learning: https://pytorch.org/tutorials/beginner/transfer_learning_tutorial.html

[6] Kaggle Flower Recognition Dataset: https://www.kaggle.com/alxmamaev/flowers-recognition