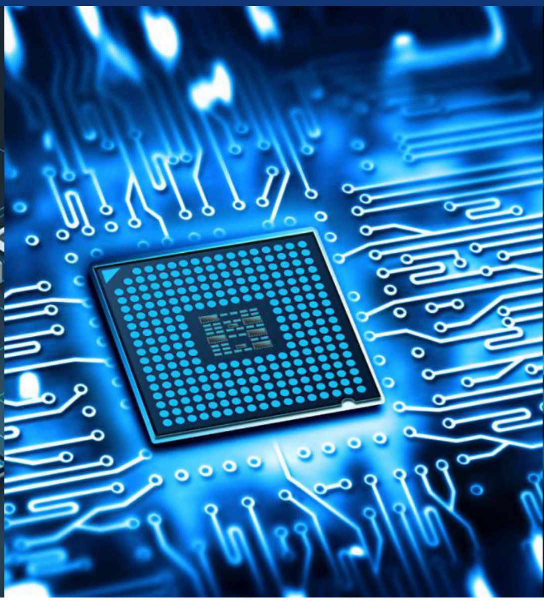
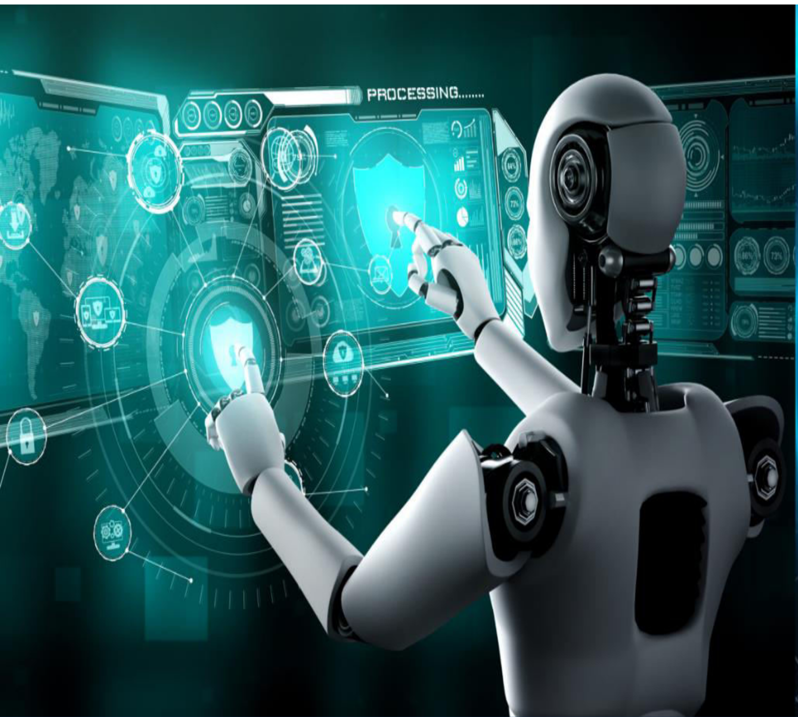
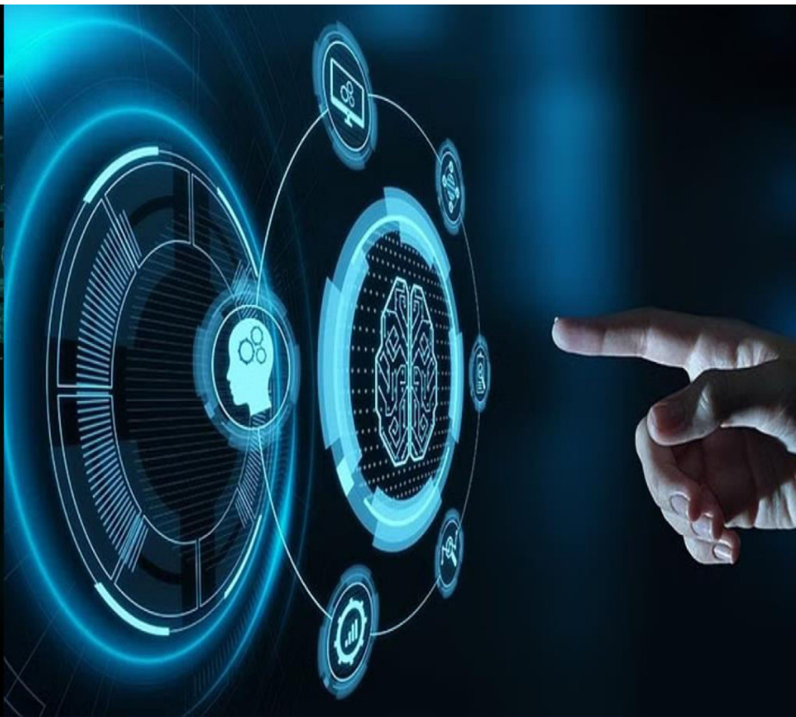
ISSN(O): 2320-9801
ISSN(P): 2320-9798


International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)





Impact Factor: 9.927

Volume 14, Issue 7, July 2026

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Flower Identification using Convolutional Neural Networks: An Intelligent Deep Learning Approach for Automated Flower Species Classification

A.Nagabhushana[1], T.Rajesh[2]

Asst. Professor, Dept. of MCA, NSRIT, Visakhapatnam, AP, India[1]

Dept. of MCA, NSRIT, Visakhapatnam, AP, India[2]

ABSTRACT: Flower identification is an important task in agriculture, botany, horticulture, and environmental research. Manual identification of flower species requires botanical knowledge and is often time-consuming. This paper presents an intelligent flower identification system using Convolutional Neural Networks (CNN) to automatically classify flower images. The proposed system recognizes seven flower species, namely Rose, Lotus, Sunflower, Tulip, Daisy, Lily, and Dandelion. Image preprocessing techniques such as resizing and normalization are applied before training the CNN model to improve prediction performance. The trained model is integrated with a Flask-based web application that enables users to upload flower images and obtain prediction results with confidence scores. The system also provides botanical information, prediction history, and PDF report generation for future reference. Experimental results demonstrate that the proposed system achieves reliable classification accuracy and provides a simple, efficient, and user-friendly solution for automated flower species identification.

KEYWORDS: Flower Identification, Convolutional Neural Network (CNN), Deep Learning, Image Classification, Flask, TensorFlow, Keras.

I. INTRODUCTION

Flower species identification plays a significant role in agriculture, horticulture, botanical research, and environmental conservation. Traditional flower identification methods depend on human expertise and require considerable time and effort. Since many flower species share similar visual characteristics such as color, shape, and texture, manual identification may lead to incorrect classification.

Recent advancements in Artificial Intelligence (AI) and Deep Learning have significantly improved image classification systems. Convolutional Neural Networks (CNN) have become one of the most effective deep learning techniques for image recognition because they automatically extract meaningful features and provide highly accurate classification results.

This paper presents an automated Flower Identification Using CNN system that classifies flower images into seven different flower species. The proposed system combines image preprocessing, CNN-based feature extraction, and classification within a Flask web application. In addition to prediction results, the application provides confidence scores, botanical information, prediction history, and PDF report generation, making the system suitable for educational, research, and practical applications.

II. LITERATURE REVIEW

Several researchers have proposed different approaches for automated flower classification using machine learning and deep learning techniques. Early flower recognition systems mainly relied on handcrafted features such as colour, texture, and shape, which often produced limited classification accuracy.

With the rapid advancement of deep learning, Convolutional Neural Networks (CNN) have become the preferred approach for image classification tasks due to their ability to automatically extract meaningful features from images. CNN-based models have significantly improved the accuracy of flower species identification by learning complex visual patterns directly from training datasets.

© 2026 IJIRCCE | Volume 14, Issue 7, July 2026 DOI: 10.15680/IJIRCCE.2026.1407033

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

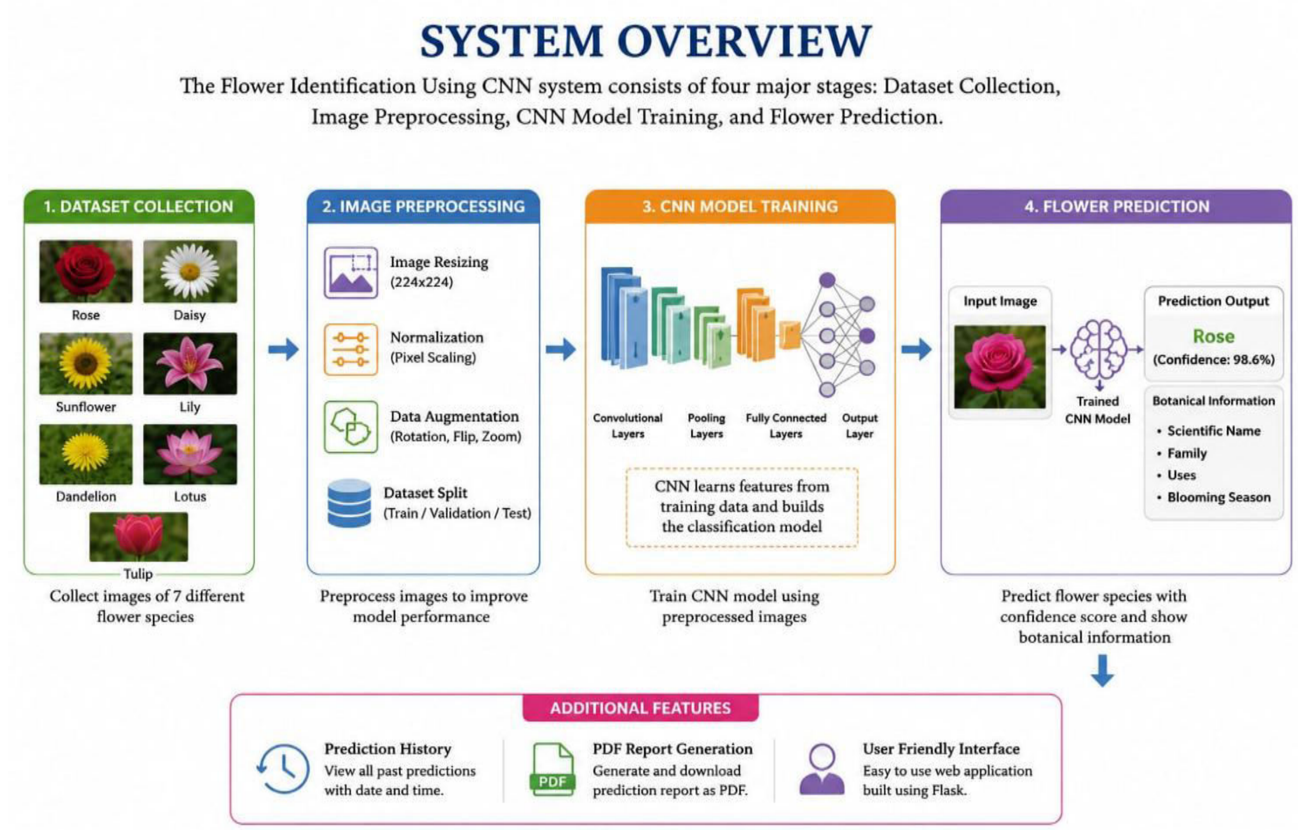
New studies have demonstrated that CNN architectures can effectively classify multiple flower species with high accuracy while reducing manual feature engineering. Researchers have also integrated deep learning models with web-based applications to provide user-friendly interfaces for real-time flower identification. However, many existing systems focus only on flower classification and provide limited information about the identified flower.
The proposed work extends the conventional flower classification approach by integrating a CNN model with a Flask-based web application. In addition to accurate flower prediction, the system provides confidence scores, botanical details, prediction history, and PDF report generation, making it more practical for educational and research purposes.

III. SYSTEM OVERVIEW

The proposed Flower Identification Using CNN system is developed to automatically identify flower species from digital images using deep learning techniques. The system consists of four major stages: dataset collection, image preprocessing, CNN model training, and flower prediction.

Initially, a flower image dataset containing seven flower species is collected and prepared for training. The images are pre-processed by resizing and normalizing them to improve the performance of the CNN model. The processed images are then used to train the CNN model, which learns important visual features required for accurate classification.
After successful training, the CNN model is integrated with a Flask-based web application. Users can upload a flower image through the web interface, and the trained model predicts the flower species with a confidence score. The application also displays botanical information such as the scientific name, family, symbolism, uses, and blooming season. Additionally, prediction history is maintained, and users can generate a downloadable PDF report for future reference.

The overall system provides an efficient, accurate, and user-friendly solution for automated flower species identification.



www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

IV. METHODOLOGY

The proposed methodology consists of five major stages, namely Dataset Collection, Image Preprocessing, CNN Model Training, Flower Prediction, and Performance Evaluation. Each stage plays an important role in improving the overall accuracy and efficiency of the flower identification system.

4.1 Dataset Collection

A flower image dataset containing seven different flower species was collected for training and testing the CNN model. The dataset includes Daisy, Dandelion, Lily, Lotus, Rose, Sunflower, and Tulip images. The collected images contain different lighting conditions, backgrounds, and flower orientations, making the model more robust for real-world prediction.

4.2 Image Preprocessing

Before training the CNN model, all flower images are pre-processed to improve classification performance. The preprocessing stage includes image resizing, normalization, and data augmentation. These techniques help reduce noise, improve image quality, and prevent model overfitting during training.

4.3 CNN Model Training

The pre-processed flower images are used to train the Convolutional Neural Network (CNN) model. During the training process, the CNN automatically learns important visual features such as color, shape, texture, and petal patterns from the flower images. The dataset is divided into training and validation sets to evaluate the learning performance of the model. Multiple training epochs are performed to improve the classification accuracy while minimizing the training loss. After successful training, the model is saved and integrated with the Flask web application for real-time prediction.

4.4 Flower Prediction and Classification

Once the CNN model is trained, users can upload a flower image through the web application. The uploaded image undergoes the same preprocessing steps before being passed to the trained CNN model. The model predicts the flower species by comparing the extracted image features with the learned patterns. The predicted flower name, confidence score, and botanical information such as scientific name, family, symbolism, uses, and blooming season are displayed to the user. The prediction result is also stored in the prediction history, and a PDF report can be generated for future reference.

4.5 Performance Evaluation

The performance of the proposed system is evaluated using training accuracy, validation accuracy, training loss, and validation loss. These evaluation metrics help measure the learning capability and prediction performance of the CNN model. Experimental results show that the trained model achieves high classification accuracy with low loss values, demonstrating its effectiveness in recognizing different flower species. The generated accuracy and loss graphs indicate that the model learns efficiently without significant overfitting and provides reliable prediction results for real-world flower images.

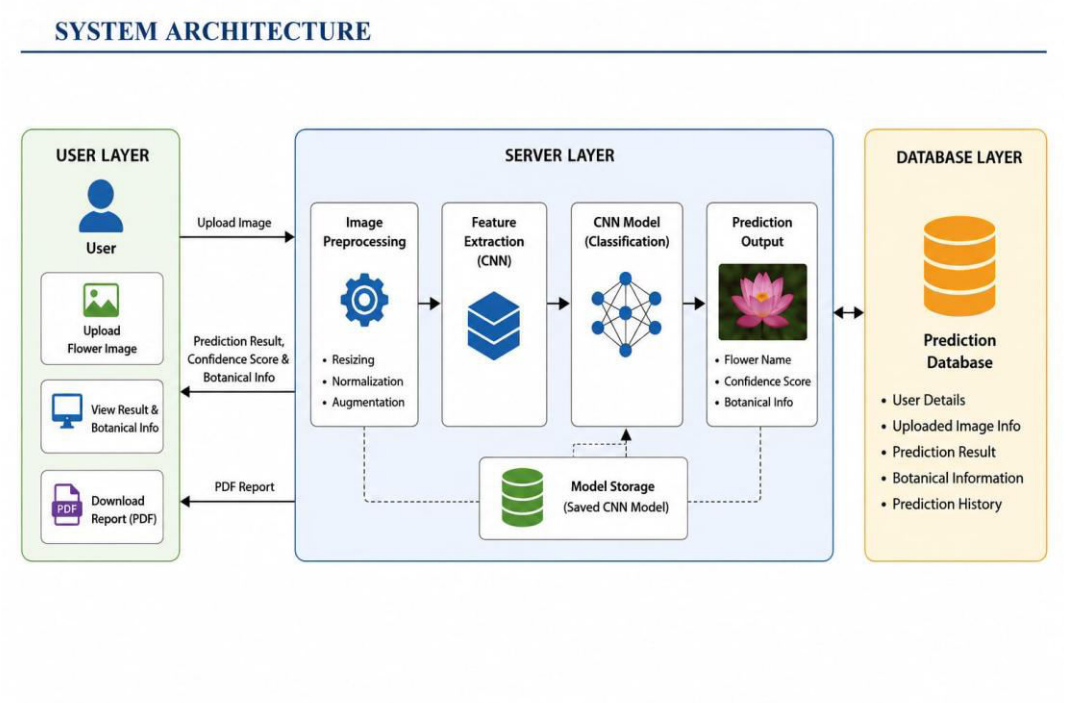
V. SYSTEM ARCHITECTURE

The proposed Flower Identification Using CNN system follows a client–server architecture that integrates deep learning with a web-based application for automated flower species identification. The architecture consists of three major components: the User Layer, Processing Layer, and Database Layer.

Initially, the user uploads a flower image through the Flask web interface. The uploaded image is pre-processed by resizing and normalizing it before passing it to the trained CNN model. The CNN model extracts important image features and classifies the flower into one of the seven predefined flower species.

The prediction result, confidence score, and botanical information are displayed to the user through the web application. All prediction details are stored in the database, allowing users to view prediction history and generate PDF reports whenever required. The overall architecture ensures efficient, accurate, and user-friendly flower species identification.

© 2026 IJIRCCE | Volume 14, Issue 7, July 2026 DOI: 10.15680/IJIRCCE.2026.1407033
www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|
International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)
(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



VI. MODULES

6.1 User Authentication Module

The User Authentication Module allows users to securely register and log in to the application. During registration, user details such as username, email address, and password are stored in the database. Only authenticated users can access the flower prediction system, ensuring secure and personalized access to the application.

6.2 Flower Prediction Module

The Flower Prediction Module is the core component of the application. Users upload a flower image through the web interface, and the image is pre-processed before being passed to the trained CNN model. The model predicts the flower species and displays the predicted flower name along with the confidence score.

6.3 Botanical Information Module

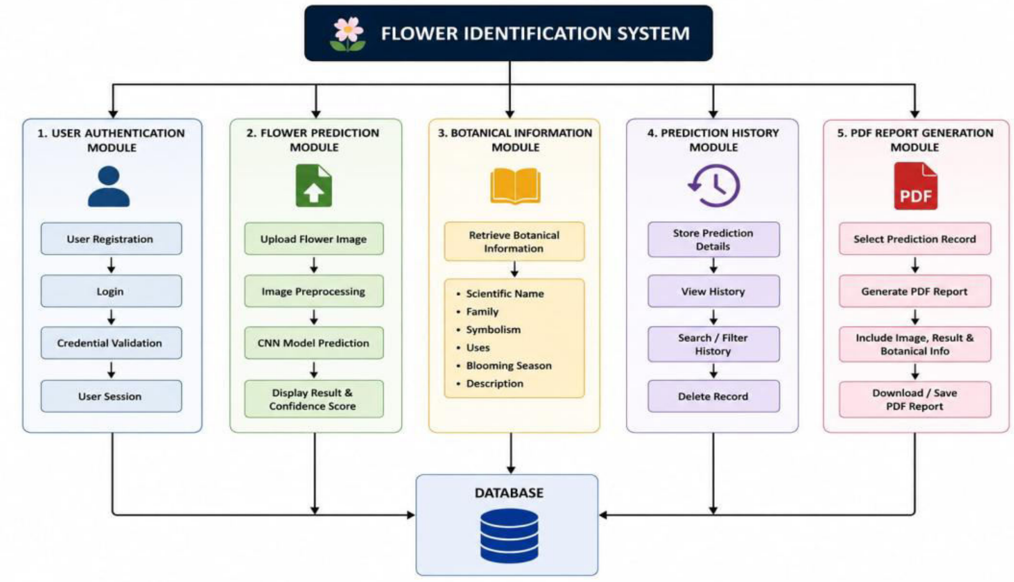
After predicting the flower species, the system displays detailed botanical information related to the identified flower. This includes the scientific name, family, symbolism, uses, blooming season, and a brief description. These details help users understand more about the identified flower.

6.4 Prediction History Module

The Prediction History Module stores every prediction made by the user in the database. Users can access their previous prediction records at any time. This feature helps maintain a history of classified flower images for future reference.

6.5 PDF Report Generation Module

The PDF Report Generation Module enables users to download a detailed prediction report in PDF format. The generated report includes the uploaded flower image, predicted flower name, confidence score, scientific name, family, symbolism, uses, blooming season, and description. This feature is useful for documentation, educational purposes, and future reference.



VII. IMPLEMENTATION DETAILS

7.1 Technology Stack

The proposed Flower Identification Using CNN system is developed using Python as the programming language and Flask as the web framework. The Convolutional Neural Network (CNN) model is implemented using TensorFlow and Keras, while image preprocessing is performed using OpenCV and NumPy. ReportLab is used for generating PDF reports containing prediction details.

7.2 Frontend Implementation

The user interface is developed using HTML, CSS, and JavaScript to provide an interactive and responsive web application. Users can upload flower images, view prediction results with confidence scores, access botanical information, check prediction history, and download PDF reports through the web interface.

7.3 Backend Implementation

The backend is developed using the Flask framework to handle image uploads, image preprocessing, CNN model prediction, database communication, prediction history management, and PDF report generation. It acts as the communication layer between the user interface and the trained CNN model.

7.4 Report Generation

The system automatically generates a PDF report containing the uploaded flower image, predicted flower name, confidence score, scientific name, family, symbolism, uses, blooming season, and description. The generated report can be downloaded and used for future reference.

7.5 Model Implementation

A Convolutional Neural Network (CNN) model is trained using categorized flower images including Rose, Lotus, Sunflower, Tulip, Daisy, Lily, and Dandelion. The trained model is integrated with the Flask application to perform accurate real-time flower species prediction.

7.6 Security Measures

The system implements user authentication, secure image upload validation, controlled file handling, and secure database management to ensure reliable prediction, data integrity, and safe storage of user information and prediction history.

© 2026 IJIRCCE | Volume 14, Issue 7, July 2026 DOI: 10.15680/IJIRCCE.2026.1407033

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

VIII. EXPERIMENTAL DETAILS AND RESULTS

8.1 Experimental Setup

The proposed Flower Identification Using CNN system was implemented and tested using Python, TensorFlow, Keras, Flask, OpenCV, and NumPy. The experiments were conducted on a system with an Intel Core i5 processor, 8 GB RAM, and the Windows operating system. The dataset consists of seven flower species: Rose, Lotus, Sunflower, Tulip, Daisy, Lily, and Dandelion. The CNN model was trained using the pre-processed dataset and evaluated using testing images.

8.2 Test Results

The developed system was tested under different scenarios to verify its functionality and prediction performance. The following table summarizes the test results.

Test Case	Description	Result	Status
User Image Upload	Upload valid flower image	Passed	✓
Image Preprocessing	Resize and normalize image	Passed	✓
CNN Model Execution	Load trained CNN model	Passed	✓
Flower Prediction	Predict flower Species	Passed	✓
Confidence Score	Display prediction confidence	Passed	✓
Botanical Information	Display disease description	Passed	✓
Prediction History	Recommend appropriate doctor	Passed	✓
PDF Report Generation	Generate downloadable PDF	Passed	✓
Response Time	Prediction generated within few seconds	Passed	✓

8.3 Performance Evaluation

The performance of the proposed CNN model was evaluated using Accuracy, Precision, Recall, and F1-Score. Experimental results indicate that the model accurately classifies different flower species and provides reliable prediction results with minimum execution time.

8.4 Observations

The proposed system successfully classified flower images into their respective flower categories. The integration of Convolutional Neural Networks (CNN) with image preprocessing techniques significantly improved prediction accuracy. The system also provides botanical information, prediction history, and PDF report generation, making it more useful for educational and research purposes.

IX. APPLICATIONS

- **Botanical Research:**
- The system assists researchers in identifying different flower species quickly and accurately.
- **Agriculture and Horticulture**:
- Farmers and horticulturists can identify flower species for cultivation and maintenance.
- **Educational Purpose:**
- Students can learn about flower species along with their botanical information through an interactive web application.

- **Plant Conservation:**
- The system helps in documenting and preserving flower species by maintaining digital prediction records.
- **Gardening and Plant Enthusiasts:**
- Users can identify unknown flowers and learn about their scientific names, family, uses, and blooming season.

X. LIMITATIONS

The proposed Flower Identification Using CNN system has a few limitations. The prediction accuracy mainly depends on the quality of the uploaded flower image. Images with poor lighting, blur, or complex backgrounds may affect the prediction result. The current system is trained to classify only seven flower species. It cannot accurately identify flower species that are not included in the training dataset. The system also requires a trained CNN model to achieve reliable prediction performance.

XI. FUTURE WORK

The proposed system can be further enhanced by increasing the number of flower species in the dataset to improve its usability. Advanced deep learning models such as ResNet, EfficientNet, or MobileNet can be integrated to achieve higher prediction accuracy. The system can also be extended to identify flowers from live camera input and mobile applications. Additional features such as multilingual support, cloud-based deployment, and automatic disease detection in flowers can further improve the functionality of the application..

XII. CONCLUSION

The proposed Flower Identification Using CNN system provides an efficient and accurate solution for automatic flower species classification. By utilizing image preprocessing techniques and a Convolutional Neural Network (CNN), the system successfully identifies flower species and provides confidence scores along with detailed botanical information. The integration of a Flask-based web application, prediction history, and PDF report generation makes the system user-friendly and suitable for educational, research, and horticultural applications. The experimental results demonstrate that the proposed system achieves reliable performance and can serve as an effective tool for automated flower identification.

REFERENCES

[1] Nurul Fatihah Sahidan, Ahmad Khairi Juha, Norasiah Mohammad, and Zaidah Ibrahim, "Flower and Leaf Recognition for Plant Identification Using Convolutional Neural Network," Indonesian Journal of Electrical Engineering and Computer Science, Vol. 16, No. 2, 2019.
[2] M. Cıbuk, U. Budak, Y. Guo, M. C. Ince, and A. Sengur, "Efficient Deep Features Selections and Classification for Flower Species Recognition," Measurement, Vol. 137, 2019.
[3] Mesut Toğaçar, Burhan Ergen, and Zafer Cömert, "Classification of Flower Species Using Deep Learning Features," Measurement, Vol. 158, 2020.
[4] Touqeer Abbas, Abdul Razzaq, Muhammad Azam Zia, and Shikali Casper, "Deep Neural Networks for Automatic Flower Species Localization and Recognition," Computational Intelligence and Neuroscience, 2022.
[5] Muhammed Yildirim, Ahmet Cinar, and Emine Cengil, "Flower Identification and Quality Assessment Using Machine Learning and CNN," Grenze International Journal of Engineering and Technology, Vol. 10, No. 2, 2024.
[6] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," Communications of the ACM, Vol. 60, No. 6, 2017.
[7] Karen Simonyan and Andrew Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," International Conference on Learning Representations (ICLR), 2015.
[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep Residual Learning for Image Recognition," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
[9] Christian Szegedy, Wei Liu, Yangqing Jia, et al., "Going Deeper with Convolutions," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.
[10] François Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

[11] Mingxing Tan and Quoc V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," International Conference on Machine Learning (ICML), 2019.
[12] M. Turkoglu, U. Budak, Y. Guo, and A. Sengur, "A Multi-Division Convolutional Neural Network-Based Plant Recognition System," Applied Sciences, Vol. 11, 2021.
[13] "Research on Plant Species Identification Based on Improved Convolutional Neural Network," Phyton International Journal of Experimental Botany, Vol. 92, 2023.
[14] Philipe A. Dias, Amy Tabb, and Henry Medeiros, "Multispecies Fruit Flower Detection Using a Refined Semantic Segmentation Network," 2018.
[15] Mustafa Yurdakul, Enes Ayan, Fahrettin Horasan, and Sakir Tasdemir, "A Mobile Application for Flower Recognition System Based on Convolutional Neural Networks," 2026.