**Iris Flower Species Recognition and Visualization**

**Problem Statement:**

Developing a system for recognizing and visualizing iris flower species based on input images.

**Proposed System/Solution:**

Utilize machine learning algorithms to classify iris flowers into different species based on features extracted from images. Develop a user-friendly interface for inputting images and displaying classification results.

**System Development Approach:**

**Data Collection:** Gather a dataset of iris flower images labeled with their corresponding species.

**Data Preprocessing:** Clean and preprocess the images to enhance feature extraction.

**Feature Extraction:** Extract relevant features from the images using techniques like histogram of oriented gradients (HOG) or convolutional neural networks (CNNs).

**Model Training:** Train a machine learning model (e.g., support vector machine, random forest, or neural network) on the extracted features to classify the iris flowers.

**Interface Development:** Design and develop a user interface for uploading images and displaying classification results.

**Testing and Validation:** Evaluate the system’s performance using a separate test dataset to ensure accuracy and reliability.

**Algorithm:**

**One possible algorithm is as follows:**

**Preprocess the input image to enhance features.**

Extract features using HOG or a pre-trained CNN.

Train a classifier (e.g., SVM) on the extracted features.

Classify the input image into one of the iris flower species.

**Deployment:**

Deploy the system as a web or mobile application, allowing users to upload images and receive classification results in real-time.

**Result:**

Present the accuracy and performance metrics of the developed system, including confusion matrices, precision, recall, and F1-score.

**Conclusion:**

The developed system effectively recognizes and visualizes iris flower species with high accuracy, demonstrating the feasibility of using machine learning for botanical classification tasks.

**Reference:**

Cite relevant research papers, datasets, libraries, and frameworks used in the development of the system.

**Program**:

# Import necessary libraries

Import pandas as pd

Import seaborn as sns

Import matplotlib.pyplot as plt

From sklearn.datasets import load\_iris

From sklearn.model\_selection import train\_test\_split

From sklearn.ensemble import RandomForestClassifier

From sklearn.metrics import classification\_report, accuracy\_score

# Load the Iris dataset

Iris = load\_iris()

Iris\_df = pd.DataFrame(data=iris.data, columns=iris.feature\_names)

Iris\_df[‘target’] = iris.target

# Data Exploration and Visualization

Print(“Summary Statistics:”)

Print(iris\_df.describe())

Print(“\nCorrelation Matrix:”)

Print(iris\_df.corr())

Print(“\nClass Distribution:”)

Print(iris\_df[‘target’].value\_counts())

# Pairplot for visualization

Sns.pairplot(iris\_df, hue=’target’)

Plt.show()

# Preprocessing (No preprocessing needed for this dataset)

# Model Building and Evaluation

X\_train, X\_test, y\_train, y\_test = train\_test\_split(iris\_df.drop(‘target’, axis=1), iris\_df[‘target’], test\_size=0.2, random\_state=42)

# Initialize and train the RandomForestClassifier

Rf\_classifier = RandomForestClassifier(n\_estimators=100, random\_state=42)

Rf\_classifier.fit(X\_train, y\_train)

# Make predictions

Predictions = rf\_classifier.predict(X\_test)

# Evaluate the model

Print(“\nClassification Report:”)

Print(classification\_report(y\_test, predictions))

Print(“\nAccuracy:”, accuracy\_score(y\_test, predictions))

**Output**:

Summary Statistics:

Sepal length (cm) sepal width (cm) petal length (cm) petal width (cm) target

Count 150.000000 150.000000 150.000000 150.000000 150.000000

Mean 5.843333 3.057333 3.758000 1.199333 1.000000

Std 0.828066 0.435866 1.765298 0.762238 0.819232

Min 4.300000 2.000000 1.000000 0.100000 0.000000

25% 5.100000 2.800000 1.600000 0.300000 0.000000

50% 5.800000 3.000000 4.350000 1.300000 1.000000

75% 6.400000 3.300000 5.100000 1.800000 2.000000

Max 7.900000 4.400000 6.900000 2.500000 2.000000

Correlation Matrix:

Sepal length (cm) sepal width (cm) petal length (cm) petal width (cm) target

Sepal length (cm) 1.000000 -0.117570 0.871754 0.817941 0.782561

Sepal width (cm) -0.117570 1.000000 -0.428440 -0.366126 -0.426658

Petal length (cm) 0.871754 -0.428440 1.000000 0.962865 0.949035

Petal width (cm) 0.817941 -0.366126 0.962865 1.000000 0.956547

Target 0.782561 -0.426658 0.949035 0.956547 1.000000

Class Distribution:

2 50

1. 50
2. 0 50

Name: target, dtype: int64

[Visualization of pairplot]

Classification Report:

Precision recall f1-score support

0 1.00 1.00 1.00 10

1 1.00 1.00 1.00 9

2 1.00 1.00 1.00 11

Accuracy 1.00 30

Macro avg 1.00 1.00 1.00 30

Weighted avg 1.00 1.00 1.00 30

Accuracy: 1.0