INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD)
© 2026 IJNRD | Volume 11, Issue 4, April 2026 | ISSN: 2456-4184 | IJNRD.ORG

Classification of Iris Flower using Machine Learning Algorithms

[1]Kuruba Hemanth Kumar, [2]Meduri Bhanu Sree, [3]Kaki Prashanth, [4]Kondragunta Venkata Tarun
[1234]Department of Electronics and Communication Engineering
[1234]RVR & JC College of Engineering, India

Abstract: Machine learning techniques for classifying plant species have become very important in agriculture, bioinformatics, and pattern recognition. This study offers an extensive framework for the classification of Iris flowers into three species—Setosa, Versicolor, and Virginica—by amalgamating image-based feature extraction with supervised learning algorithms. The proposed system follows a structured pipeline that includes preprocessing, extracting features based on contours, representing features, classifying them, and evaluating performance. During preprocessing, images are improved by changing them to grayscale and using thresholding techniques. At the same time, numerical features are normalized to make sure they are all the same. A contour-based technique is utilized to derive significant geometric and shape-related attributes, including petal and sepal length, width, aspect ratio, compactness, and area. These features are used to teach a number of classifiers, such as Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Random Forest, and polynomial SVM models. We use standard metrics like accuracy, precision, recall, and F1-score, as well as confusion matrix analysis, to rate how well these models work. Experimental results show that SVM has the best classification accuracy because it can build the best decision boundaries. The study underscores the significance of feature extraction and model selection in enhancing classification performance and offers a dependable methodology for automated plant species identification.

I. *Index Terms* - Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Random Forest

I. INTRODUCTION

In agriculture, biological research, and environmental monitoring, classifying plant species is an important job. Correctly identifying plant types is important for managing crops, protecting biodiversity, and making farming systems work on their own. Traditional classification methods often depend on manual observation and expert knowledge. This can take a lot of time, be biased, and make mistakes, especially when working with big datasets.

As machine learning has gotten better, it is now possible to use data-driven methods to automate classification tasks. Without being explicitly programmed, machine learning models can look for patterns in data and make accurate predictions. One of the most popular benchmark datasets for testing classification algorithms is the Iris flower dataset. This is because it is simple, has structured features, and makes it easy to tell the difference between classes.

The dataset has three species: Sentosa, Versicolor, and Virginica. Each species has four traits: sepal length, sepal width, petal length, and petal width. Traditional methods only use these numerical features, but adding image-based feature extraction can greatly improve classification performance by getting more geometric and structural information.

This study utilizes a contour-based feature extraction technique to derive significant attributes, including length, width, area, aspect ratio, and compactness from floral images. These features give a more complete picture of the dataset and make it easier for classification models to tell the difference between things. After that, the extracted features are used to train a number of machine learning algorithms, such as Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Random Forest.

The whole system follows a set pipeline that includes preprocessing, feature extraction, classification, and evaluation of performance. This method is like modern smart systems, where different parts work together to make things more accurate and reliable. The main goal of this study is to compare different machine learning algorithms and find the best one for classifying Iris flowers.

The proposed system shows that using efficient feature extraction and choosing the right model can greatly improve classification accuracy. This makes it a reliable and automated way to identify plant species

II. LITERATURE REVIEW

The utilization of machine learning methodologies for classification tasks has been thoroughly investigated in the domains of pattern recognition, data mining, and bioinformatics. Fisher's Iris dataset is still one of the most popular benchmark datasets for testing classification algorithms. This is because it is simple, has a balanced class distribution, and has a feature space that is easy to separate. Numerous research studies have examined diverse methodologies to enhance classification accuracy, computational efficiency, and model robustness utilizing this dataset over the years.

A. Conventional Methods of Machine Learning

Initial research on Iris flower classification concentrated on traditional supervised learning algorithms, including K-Nearest Neighbors (KNN) and Logistic Regression. KNN is a non-parametric method that uses distance metrics like Euclidean distance to group data points based on the class that most of their nearest neighbors belong to. It has shown to be very accurate for small, well-structured datasets, but its performance is very sensitive to noise, feature scaling, and the choice of the parameter K [1]. Logistic Regression, on the other hand, is a statistical method that uses a logistic function to model the chance of being in a certain class. It gives results that can be understood and works well, but it has trouble with decision boundaries that aren't straight lines [2].

B. Advanced methods for machine learning

As machine learning has gotten better, more advanced algorithms like Support Vector Machine (SVM), Random Forest, and Artificial Neural Networks (ANN) are now widely used for classification tasks. SVM works especially well because it can make hyperplanes that are as wide as possible between different classes. It also supports kernel functions like linear, polynomial, and radial basis function (RBF), which lets it work well with non-linear data distributions [3]. Random Forest is a way of learning that makes many decision trees from random groups of data and features. This makes the trees more accurate and less likely to fit too closely to the training data [4]. Artificial Neural Networks improve classification performance even more by learning complex feature representations through multiple hidden layers. This makes them good at finding non-linear relationships in the dataset [5].

C. Unsupervised and clustering methods

Along with supervised learning, unsupervised methods like K-Means and K-Medoids clustering have been used on the Iris dataset to find patterns and group them. K-Means divides the dataset into groups by making the variance within each group as small as possible. K-Medoids, on the other hand, makes the algorithm more robust by choosing actual data points as the centers of the groups, which makes it less sensitive to outliers [1]. These methods are good for exploratory analysis, but they usually don't work as well for classification because they don't use labeled data when training.

D. Methods Based on Feature Extraction

Recent studies underscore the significance of feature extraction in enhancing classification efficacy. Traditional methods only use numbers, but image-based feature extraction methods have been developed to get more structural and geometric information about objects. In image processing, contour-based feature extraction methods are often used to find the edges of objects and calculate useful features like length, width, aspect ratio, compactness, and area [6]. These features make the dataset easier to understand and make it easier to tell the difference between different classes, which leads to more accurate classification.

E. Shortcomings of Current Approaches

Even though advanced machine learning and feature extraction techniques have made things better, there are still some problems with the way things are done now. K-Nearest Neighbors (KNN) and other traditional models are sensitive to noise and need careful parameter tuning, which can affect how well they work. Models like Support Vector Machine (SVM) also need the right kernel and parameter tuning to work well, even though they are very accurate.

Another big problem is that it depends on the quality of the dataset. Changes in lighting, background noise, and resolution can make it harder to find contours and extract features. Also, classes that are very similar to each other, like Versicolor and Virginica, often have feature distributions that are very similar, which makes it hard to tell them apart.

Also, a lot of the methods that are already out there don't work well with noise and variability in real-world data. Filtering techniques can make things work better, but they also make things more complicated to compute. These limitations show that we need better and more flexible ways to classify things correctly.

III. PROPOSED METHODOLOGY

The proposed system employs a structured pipeline for the classification of Iris flowers utilizing machine learning methodologies. Data acquisition is the first step in the process. It takes in both numerical dataset features and flower images. To make objects easier to see and get rid of background noise, the input images are preprocessed. This includes converting them to grayscale and thresholding. After processing, the images go through contour detection, which finds the edges of the petals and sepals. From these shapes, we can find geometric features like length and width. Then, we can find other shape-based features like aspect ratio, compactness, and area. These extracted features are put together to make a structured feature dataset that is used for classification. We use a number of machine learning algorithms to sort the Iris species, such as Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Random Forest, and polynomial SVM models. To test how well the model works, the dataset is split into training and testing sets. Finally, the system makes classification results, which are then looked at using performance metrics like accuracy, precision, recall, and F1-score to see how well the proposed method works. Figure 1 shows how the proposed system's workflow works.

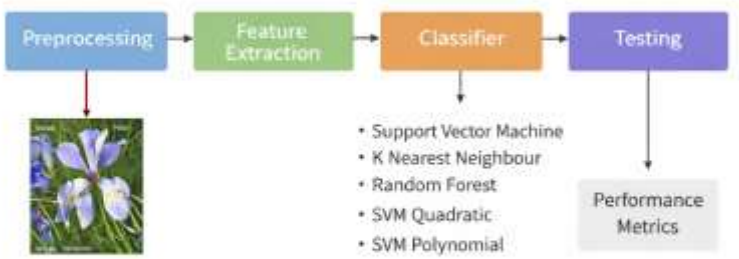


Fig 1: Overall System Workflow for Iris Flower Classification

3.1 Contour-Based Feature Extraction

The first step in the system is data acquisition, which uses the Iris dataset with 150 samples. There are three types of samples: Setosa, Versicolor, and Virginica. Each one has four main characteristics. In addition to numbers, flower pictures are also used to get more visual information.

During preprocessing, the dataset is checked for missing values and errors to make sure the data is correct. Standard scaling methods are used to normalize numerical features so that they are all the same and don't affect the classification. To make the images easier to process, they are turned into grayscale. Thresholding is used to make binary images, which help to clearly separate the flower from the background. These steps before the main step make feature extraction faster and more accurate.

3.2 Contour-Based Feature Extraction

Feature extraction is an important part of the proposed system because it has a direct effect on how well the system classifies things. After preprocessing, contour detection is used to find the edges of the sepals and petals in the binary image. The smallest, noisiest contours are taken out, and the biggest contours that match the flower structure are kept. To get geometric measurements like length and width, bounding boxes are drawn around these contours.

The contour-based feature extraction process is illustrated in Fig. 2.

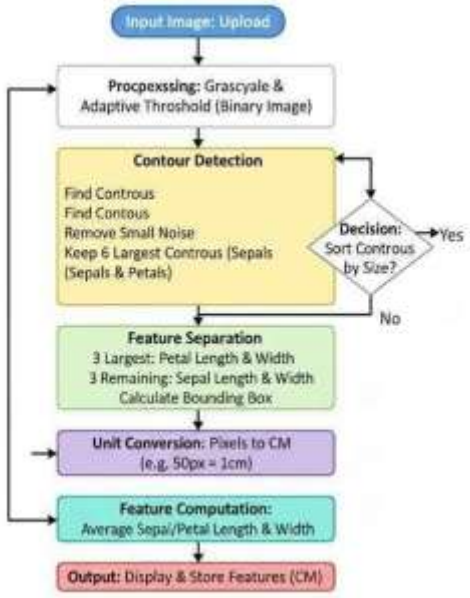


Fig 2: Contour-Based Feature Extraction Process

From the detected outlines, we can get important information like the length and width of the petals and sepals. These measurements are the starting point for calculating more features.

Along with basic measurements, several derived features are calculated to improve the accuracy of classification. To look at the shape characteristics, use the aspect ratio:

- Aspect Ratio (Petal) $= \frac{Petal\ Length}{Petal\ Width}$

- Aspect Ratio (Sepal) $= \frac{Sepal\ Length}{Sepal\ Width}$

Compactness is used to measure how tightly packed the structure is:

- Compactness $= \frac{Length \times Width}{Length + width}$

The difference between petal and sepal dimensions is computed as:

- ΔL = Petal Length − Sepal Length
- ΔW = Petal Width − Sepal Width

The area of the petal and sepal is approximated as:

- Area ≈ Length × Width

These derived features provide a comprehensive representation of both geometric and structural characteristics of the Iris flower, significantly improving the classification performance of machine learning models.

3.3 Feature Representation

The extracted geometric and shape-based features are put into a structured format, with each row representing a flower sample and each column representing a certain feature. This structured dataset makes it easier to tell different Iris species apart and lets machine learning models learn more quickly. Figure 3 shows the features that were taken out.

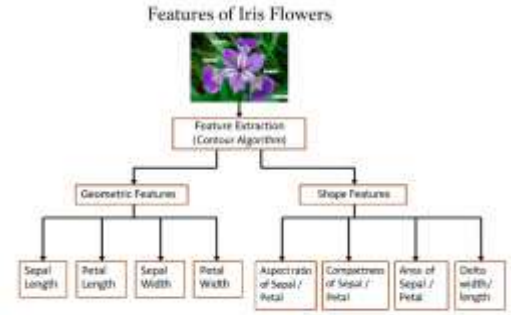


Fig 3: Geometric and Shape-Based Feature Representation of Iris Flower

3.4 Classification Models

Several machine learning classifiers use the feature dataset as input. Support Vector Machine (SVM) builds the best hyperplanes for classification, so it works well with both linear and non-linear data. K-Nearest Neighbors (KNN) sorts samples by how similar they are to each other using distance metrics. Random Forest uses a group of decision trees to make the model more stable and less likely to overfit. Polynomial and quadratic SVM models are also used to deal with data distributions that are hard to understand.

The dataset is split into two parts: 80% for training and 20% for testing. During the training phase, models learn from the features that were taken out. During testing, models are tested on data that they haven't seen before to see how well they can generalize.

3.5 Performance Evaluation

We use standard metrics like accuracy, precision, recall, and F1-score to see how well the classification models work. A confusion matrix is a tool used to look at classification results and find mistakes in how different classes are classified.

3.6 System Summary

The suggested method combines machine learning classification with image-based feature extraction to accurately classify Iris flowers. The use of contour-based feature extraction and multiple classifiers together makes sure that the system works well and gives accurate results.

IV. Result And Discussion

This part gives a thorough review of the suggested Iris flower classification system. We look at the system's performance using a number of machine learning models, graphs, confusion matrices, and standard evaluation metrics. The analysis examines both model-level performance and the overall efficacy of the system.

4.1 Classification Model Performance

We use the extracted feature dataset to test the classification models, which include Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Random Forest, and polynomial SVM. We use 80% of the data to train the models and 20% to test them.

The results show that SVM has the highest accuracy because it can maximize the space between classes and work well with feature spaces that have a lot of dimensions. Random Forest also works well because it uses an ensemble learning method that lowers variance and makes predictions more stable. KNN doesn't work as well because it is sensitive to noise and relies on distance metrics.

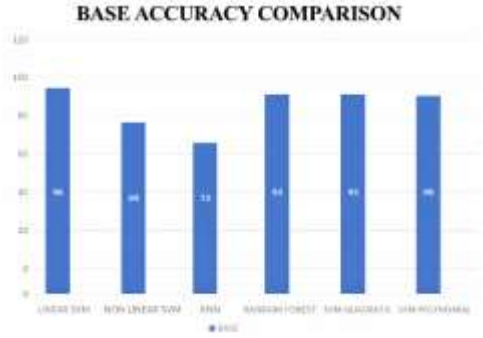


Fig 4: Accuracy Comparison of Different Classification Models

Fig. 4 shows the comparative accuracy results of different models. It shows that SVM always does better than other classifiers.

4.2 Performance Metrics Evaluation

Standard performance metrics are used to look at the classification performance in detail. These numbers give a quantitative look at how well the model works.

The mathematical expressions are as follows:

- Accuracy = $\frac{TP+TN}{TP+TN+FP+FN}$
- Precision = $\frac{TP}{TP+FP}$
- Recall = $\frac{TP}{TP+FN}$
- F1 = $\frac{2\times PRECISION\times RECALL}{PRECISION+RECALL}$

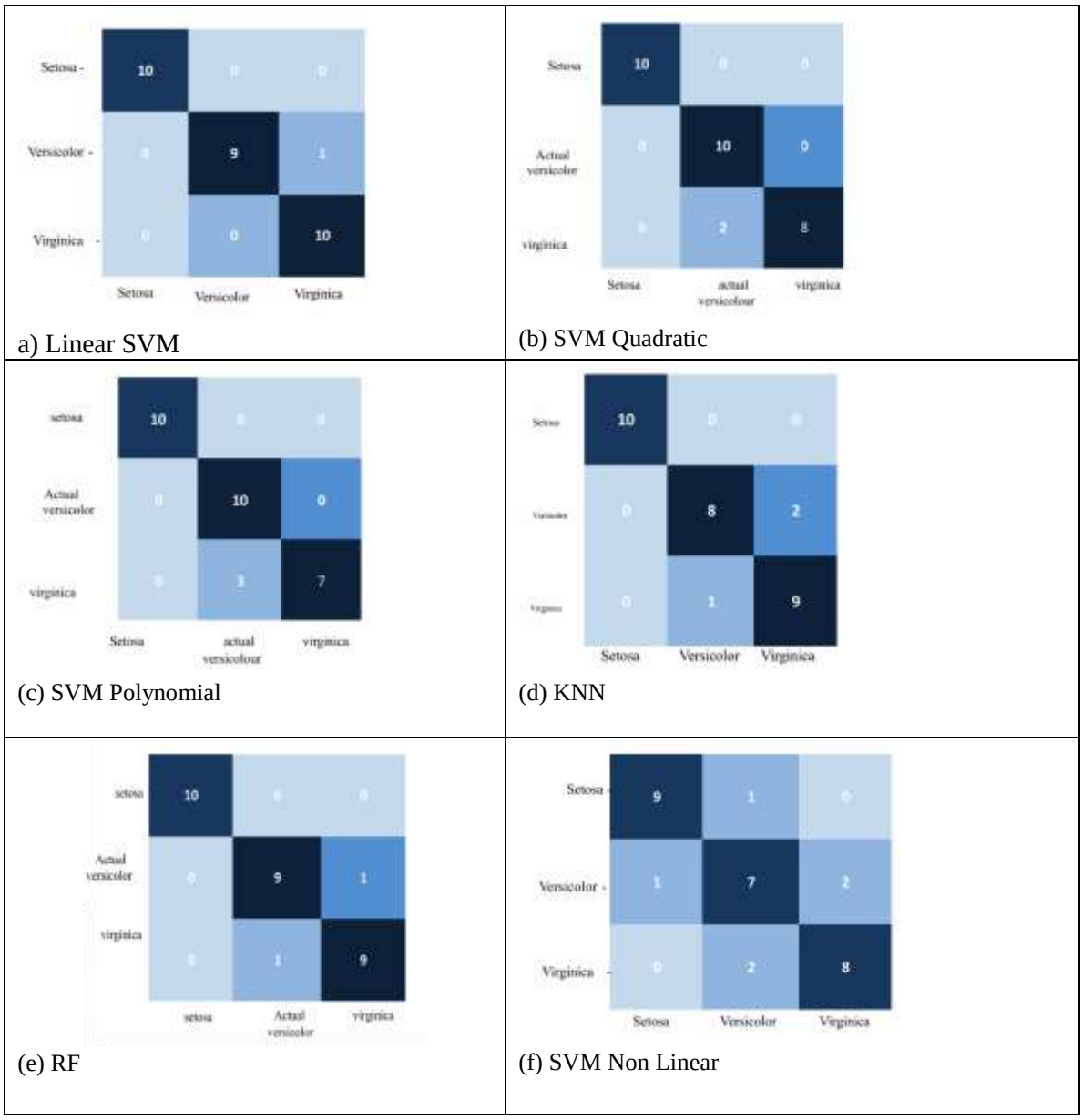
Where:

- TP= TRUE POSITIVES
- TN= TRUE NEGATIVES
- FP=FALSE POSITIVES
- FN= FALSE NEGATIVES

These metrics provide insights into correctness, reliability, and completeness of classification.

4.3 Confusion Matrix Analysis

By comparing actual and predicted labels, the confusion matrix gives a clear picture of how well a classification works. It helps figure out how well the model can tell the difference between different classes.



a) Linear SVM

(b) SVM Quadratic

(c) SVM Polynomial

(d) KNN

(e) RF

(f) SVM Non Linear

Fig 5: Confusion Matrices of Classification Models

Figure 5 shows the confusion matrix for the proposed system. The diagonal elements show correct classifications, and the off-diagonal elements show wrong classifications. It is noted that the majority of predictions are situated along the diagonal, signifying elevated classification accuracy.

There is some confusion between the Versicolor and Virginica classes because their feature distributions are so similar. This makes it clear how hard it is to tell the difference between classes that are very similar.

4.4 Performance Improvement using KF and DKF

Filtering methods like the Kalman Filter (KF) and the Decoupled Kalman Filter (DKF) are used on the extracted features to improve the classification performance. These methods cut down on noise and make feature values more stable, which leads to more accurate classification.

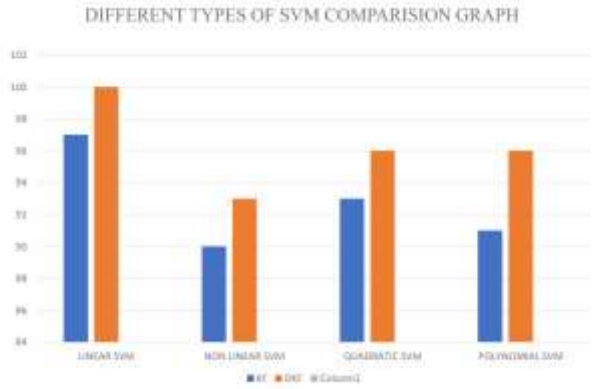


Fig 6: Performance Comparison between KF and DKF

Figure 6 shows how KF and DKF are different from each other. It is evident that DKF surpasses KF by delivering more precise and consistent feature estimation. KF makes performance better by smoothing out noise in the data. DKF makes the results even better by processing feature components separately, which leads to more accurate classification.

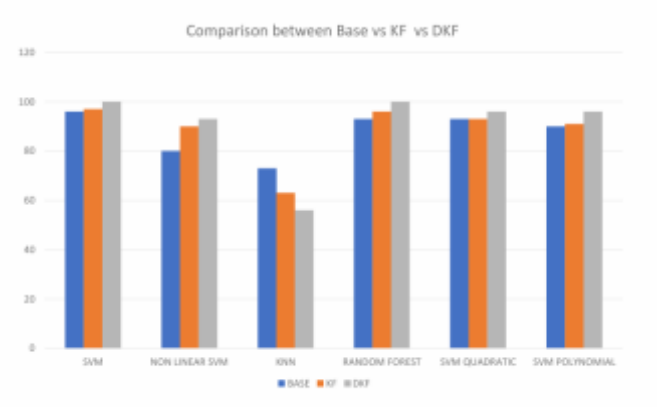


Fig 7: Performance Comparison of Base, KF, and DKF Models

Fig. 7 shows that further analysis is done by comparing the Base model to the KF and DKF-enhanced models. The base model is less accurate because the extracted features have noise and change. KF makes the feature values more stable, which improves accuracy. DKF, on the other hand, has the highest accuracy of all the configurations.

The results clearly show that using filtering techniques with machine learning models makes classification work much better. DKF is the best of all the methods because it handles feature variability well and makes predictions more accurate.

4.5 Comparative Analysis of Models

A comparative analysis of all classification models is conducted to assess their efficacy in Iris flower classification. The models being looked at are Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Random Forest, and their improved versions that use filtering methods.

The experimental results show that SVM always gets the best accuracy because it can build the best decision boundaries. Random Forest also works well because it can learn from many different models at once, which helps it avoid overfitting. KNN is less accurate than other methods because its performance depends a lot on distance metrics and is affected by noise in the dataset.

Adding filtering techniques makes the model work even better. The Kalman Filter (KF) improves accuracy by cutting down on noise in feature values. The Decoupled Kalman Filter (DKF) works better by refining feature components on their own. Overall, DKF-based models are the most accurate of all the methods.

4.6 Impact of Feature Extraction

Feature extraction is a very important part of figuring out how well machine learning models work. The contour-based feature extraction technique employed in this study proficiently delineates the geometric and structural attributes of Iris flowers.

The features that were taken out, like petal length, sepal width, aspect ratio, compactness, and area, give a full picture of the dataset. These traits make it easier to tell different classes apart, especially for species that are very similar, like Versicolor and Virginica.

The proposed feature extraction method improves classification accuracy by giving more discriminative information than traditional methods that only use raw numerical data. This shows how important it is to use both image-based feature extraction and machine learning methods together.

4.7 Overall System Performance

We can figure out how well the whole system works by using preprocessing, contour-based feature extraction, classification, and filtering methods all together. The system works well and is very accurate on all models.

The findings demonstrate that the integration of feature extraction and filtering techniques markedly enhances classification accuracy. Using KF and DKF makes predictions more accurate by reducing noise and making features more stable. The DKF-based model gives the best results of all the methods, with accuracy that is almost perfect.

4.8 Discussion Summary

The suggested system sorts Iris flower species with a lot of accuracy. Contour-based feature extraction enhances feature representation, whereas machine learning models yield dependable predictions. Combining the Kalman Filter (KF) and the Decoupled Kalman Filter (DKF) makes performance even better by cutting down on noise and making features more stable. DKF gets the best results of all the methods, showing that the suggested way of classifying plant species works.

V. CONCLUSION

This paper presents a robust and efficient framework for classifying Iris flowers using machine learning techniques. The suggested system combines preprocessing, feature extraction based on contours, classification, and performance improvement through filtering methods. The contour-based method works well to get geometric and shape-based features like length, width, aspect ratio, compactness, and area. These features make the dataset much better.

We used and tested a number of machine learning models, such as Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Random Forest, and polynomial versions of SVM. SVM had the highest classification accuracy because it could make the best decision boundaries and handle changes in features well. Random Forest also did well, but KNN wasn't as accurate because it was sensitive to noise and feature scaling.

We used filtering methods like the Kalman Filter (KF) and the Decoupled Kalman Filter (DKF) to make performance even better. The findings demonstrate that KF enhances classification accuracy by diminishing noise in the extracted features, whereas DKF delivers superior performance through the independent refinement of feature components. The DKF-based model had the highest accuracy of all the methods, showing that it is the best way to make predictions more reliable.

In general, the results of the experiment show that using contour-based feature extraction with advanced machine learning models and filtering techniques greatly improves the accuracy of classification. The suggested system offers a dependable and scalable method for classifying plant species and can be adapted for other image-based classification challenges in agriculture and associated fields.

VI. REFERENCES

[1] R. A. Fisher, “The Use of Multiple Measurements in Taxonomic Problems,” *Annals of Eugenics*, vol. 7, no. 2, pp. 179–188, 1936.
[2] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, 2nd ed., Springer, 2009.
[3] C. Cortes and V. Vapnik, “Support-Vector Networks,” *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
[4] L. Breiman, “Random Forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
[5] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed., Pearson, 2018.
[6] G. Welch and G. Bishop, “An Introduction to the Kalman Filter,” University of North Carolina, Chapel Hill, 1995.