Dissertation on

DESIGN AND DEVELOPMENT OF AN IMPROVED FLOWER RECOGNITION SYSTEM

Thesis submitted towards partial fulfilment of the requirements for the degree of

Master of Technology in IT (Courseware Engineering)

Submitted by
Sreyasee Biswas

EXAMINATION ROLL NO.: M4CWE22026
UNIVERSITY REGISTRATION NO.: 154508 of 2020-2021

Under the guidance of
Prof. Ranjan Parekh

School of Education Technology
Jadavpur University

Course affiliated to
Faculty of Engineering and Technology
Jadavpur University
Kolkata-700032
India

2022

M.Tech. IT (Courseware Engineering)
Course affiliated to
Faculty of Engineering and Technology
Jadavpur University
Kolkata, India

CERTIFICATE OF RECOMMENDATION

This is to certify that the thesis entitled **"DESIGN AND DEVELOPMENT OF AN IMPROVED FLOWER RECOGNITION SYSTEM"** is a bonafide work carried out by Sreyasee Biswas under our supervision and guidance for partial fulfilment of the requirements for the degree of Master of Technology in IT (Courseware Engineering) in School of Education Technology, during the academic session 2020-2022.

SUPERVISOR
School of Education Technology
Jadavpur University,
Kolkata-700 032

DIRECTOR
School of Education Technology
Jadavpur University,
Kolkata-700 032

DEAN - FISLM
Jadavpur University,
Kolkata-700 032

M.Tech. IT (Courseware Engineering)
Course affiliated to
Faculty of Engineering and Technology
Jadavpur University
Kolkata, India

CERTIFICATE OF APPROVAL **

This foregoing thesis is hereby approved as a credible study of an engineering subject carried out and presented in a manner satisfactory to warranty its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by this approval the undersigned do not endorse or approve any statement made or opinion expressed or conclusion drawn therein but approve the thesis only for purpose for which it has been submitted.

Committee of final examination for evaluation of Thesis

** Only in case the thesis is approved.

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC ETHICS

I hereby declare that this thesis contains literature survey and original research work by the undersigned candidate, as part of her **Master of Technology in IT (Courseware Engineering)** studies.

All information in this document has been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by this rule and conduct, I have fully cited and referenced all materials and results that are not original to this work.

NAME: Sreyasee Biswas

EXAMINATION ROLL NUMBER: M4CWE22026

THESIS TITLE: **DESIGN AND DEVELOPMENT OF AN IMPROVED FLOWER RECOGNITION SYSTEM**

SIGNATURE: DATE:

Dedicated to
My Parents, My Husband
and
my daughters Srishti and Shriya

Acknowledgement

This preface is to express gratitude of the author to all those individuals whose co-operation guided her in every aspect to make the thesis work successful.

In presenting this thesis report, the author would like to express her sincere gratitude and indebtedness to her Thesis Advisor ***Prof. Ranjan Parekh****, Professor, School of Education Technology, Jadavpur University, Kolkata for his valuable suggestion and requisite guidance in various aspects of this project work. It was a great pleasure to work under his supervision. The author would like to express gratitude for his advice on both researches as well as on her career which has been priceless.*

The author is thankful to ***Prof. Matangini Chattopadhyay****, Professor, School of Education Technology, Jadavpur University,* ***Dr. Saswati Mukherjee****, Assistant Professor, School of Education Technology and* ***Mr. Joydeep Mukherjee****, Assistant Professor, School of Education Technology, Jadavpur University for their support in academic matters.*

The author would also like to thank ***Mr. Susovan Jana****, Research Scholar- School of Education Technology, for his co-operation, help, motivation and who supported him for writing, and incented him to strive towards his goal.*

The author would also like to thank her family members for their constant support and encouragement for making this Thesis report possible at all.

Lastly the author would like to thank all who have assisted her directly or indirectly to accomplish this work.

Dated:

Jadavpur University

Kolkata

With Regards,

Sreyasee Biswas

Contents

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

Acronym	**Full-Name**
ANN	Artificial Neural Network
CNN	Convolutional Neural Network
DB	Data Base
DCNN	Deep Convolutional Neural Networks
DWT	Discrete Wavelet Transform
FDR	False Discovery Rate
FN	False Negative
FP	False Positive
GIST	global image descriptor
GLCM	Gray Level Co-occurrence Matrix
KNN	K-Nearest Neighbour
LBP	Local Binary Pattern
mRMR	minimal Redundancy Maximum Relevance
MSER	Maximally Stable Extremal Regions
PCA	Principle Component Analysis
PCANet	PCA Network
PNG	Portable Network Graphics
RGB	Red, Green, Blue
ROI	Region of Interest
SURF	Speeded-Up Robust Features
SVM	Support Vector Machine
TP	True Positive
VGG	Visual Geometry Group

NOMENCLATURE

μ	mean (average colour in an RGB image)
$\mu(i)$	mean for i-th colour channel
μ_r	mean for red colour channel
μ_g	mean for green colour channel
μ_b	mean for blue colour channel
σ	standard deviation
$\sigma(i)$	standard deviation for the i-$_{th}$ colour channel
σ_r	standard deviation for the red colour channel
σ_g	standard deviation for the green colour channel
σ_b	standard deviation for the blue colour channel
$p(i,j)$	colour intensity value of the j-th pixel of the i-th colour channel
F_{col}	colour feature descriptor
G_e	GLCM energy
G_h	GLCM homogeneity
G_c	GLCM contrast
G_r	GLCM correlation
p_{ij}	Element i,j of the normalized symmetrical GLCM
μ_{glcm}	GLCM mean
${\sigma_{glcm}}^2$	variance of the intensities of all reference pixels
F_{glcm}	GLCM feature descriptor
F_{gist}	GIST descriptor
F_{surf}	salient features of SURF
I_1	First invariant moment of Hue’s seven invariant moment
I_2	Second invariant moment of Hue’s seven invariant moment
I_3	Third invariant moment of Hue’s seven invariant moment
I_4	Fourth invariant moment of Hue’s seven invariant moment
I_5	Fifth invariant moment of Hue’s seven invariant moment
I_6	Sixth invariant moment of Hue’s seven invariant moment
I_7	Seventh invariant moment of Hue’s seven invariant moment

μ_{ij}	central moments of any order are invariant with respect to translations
η_{ij}	Invariants oments of μ_{ij}
F_{im}	Hue’s invariant moment feature descriptor
F_{comb1}	first combined feature descriptor
F_{comb2}	second combined feature descriptor
F_{comb3}	third combined feature descriptor
F_{comb4}	fourth combined feature descriptor
F_{comb5}	fifth combined feature descriptor

EXECUTIVE SUMMARY

Flower recognition from natural images has always been a challenging task as same flower can be of different colours and different flowers can be of same colour. Also, there are similarities among different types of flowers with respect to their shape and petal count. Hence this is very difficult to categorize flowers based on only colour, Shape or number of petals. For example, Hibiscus, Rose, Morning glory, Daisy can be of red, pink, and white colour. Similarly, Daisy, Calendula, Sunflower is mostly found in yellow colour and they have similarities in their petal shape and all of the three flower species are having multiple petals looks alike. Also, natural images include the surrounding background along with the flower. In this paper, a flower recognition model based on its colour, texture, SURF and GIST features is proposed. Standard database of medium size RGB copped flowers is used for experiment. The proposed system only considers the centre part of the flower image and ignores the other part as most of the flowers have different colour, texture and shape in the centre part of the flower. However, most of the flowers of same species have similarities in the centre part of it even though the petals are of different colour. Pre-processing like resizing has been done on the input images to reduce computational loads. Segmentation is not required as the proposed system uses cropped flower as input and only considers the centre part of the flower image. From the input RGB image SURF (Speeded-Up Robust Features) feature is extracted from the 10 strongest points. Then the RGB image is divided into 9 non overlapping equal blocks (3X3) and the 5 the block is cropped from the RGB image and considered to be the Region of Interest (ROI) for further processing.

Colour moments Mean and Standard Deviation of each R, G, B channel are extracted from the ROI as they are scaling and rotation invariant. Hue's seven invariant moments are calculated as they are invariant to translation, scale, and rotation, and reflection. GIST feature is also extracted from the ROI to gather various important statistics from the centre part of the flower. Texture feature like energy, homogeneity, contrast and correlation are extracted using

GLCM (Gray Level Co-occurrence Matrix) method from the grey image. All of the features extracted from the ROI are represented as mathematical feature vectors using combined features and 6 different feature vectors are created for 6 different combinations of features (colour moments, texture features, GIST features, SURF features and Hue's Moments).

Classification is done using both Support Vector Machine (SVM) and K-Nearest Neighbour (KNN). Eleven different classification models are created from the 6 combined feature vectors and 2 classifiers. When a query image is sent to the system, all 11 classification models predict a class and then 11 predicted class is sent to the Decision Engine to determine the highly predicted class of the query input. Final prediction of the input image is done based on the maximum occurrences of the predicted class determined from all 11 classification models. The trained model is able to classify 12 different species of flowers (Hibiscus, Rose, Poppy, Daisy, Calla Lily, Plumeria, Morning Glory, Marigold, Tagar (Crape jasmine), Sunflower, Gazania and Calendula) which are known to the system and also identify the flowers which are not known to the system. The proposed approach displays an overall accuracy of 89.9%.

Chapter 1

INTRODUCTION

1.1. Overview

Flowers are the most attractive part of the plant and they are usually beautiful and colourful. Most of the flowers have wonderful fragrances and many of them are used for preparation of perfume and other fragrant. Even though the shapes, sizes and colours of the flowers vary, all of the flowers have some features common as the biological function of flower is to facilitate reproduction for the survival of the plant species. A typical flower has four parts such as sepal, petal, stamen, and carpel. Stamen is the male part of the flower and carpel is the female part of the flower. Strong pleasant smell, colour, shape and size of the flowers attract the pollinators (birds, bees, wasps, butterflies, moths, beetles, etc.) to have mutual beneficial towards reproduction for their survival. Colours of the flowers come from various pigments present in the flower. Carotenoids produce yellow, brown and orange colours. Anthocyanin produces red, pink, purple and blue colours.

Most people like flowers not only for its beauty but also for its medical benefits. The sight of beautiful fresh flowers has a positive and calm effect on the recipient. Fragrance of flowers triggers the chemicals (Dopamine, Oxytocin, and Serotonin) in human brain. Since long medical practitioners from various cultures across the world use the therapeutic properties of medicinal flowers to heal people. Also, flowers such as weld, cosmos, dahlias, coreopsis, marigold, sunflower etc. are considered as dye flowers and are being used to prepare different natural plant dyes.

More than three lakhs varieties of flowers exist in the earth and they belong to different flower families and species. Sunflower family Asteraceae (24,000 species), orchid family Orchidaceae (20,000 species) and pea family Fabaceae

(18,000 species) are the three largest flowering plant families in the world. Due to these wide varieties of same family, manual recognition of flowers is a very challenging task. There are variations of colour in similar flower species, and similarities in colour in different flower species. Similarly, there are similarities in shape of different flower species. For example, Hibiscus, Rose, Morning glory, Daisy can be of red, pink, and white colour (**Fig.1.1**). Similarly, Daisy, Calendula, Sunflower and Gazania are mostly found in yellow colour and they have similarities in their petal shape and all of the three flower species are having multiple petals looks alike (**Fig.1.2**).

Fig. 1.1 Interclass similarities in petal colour of flower

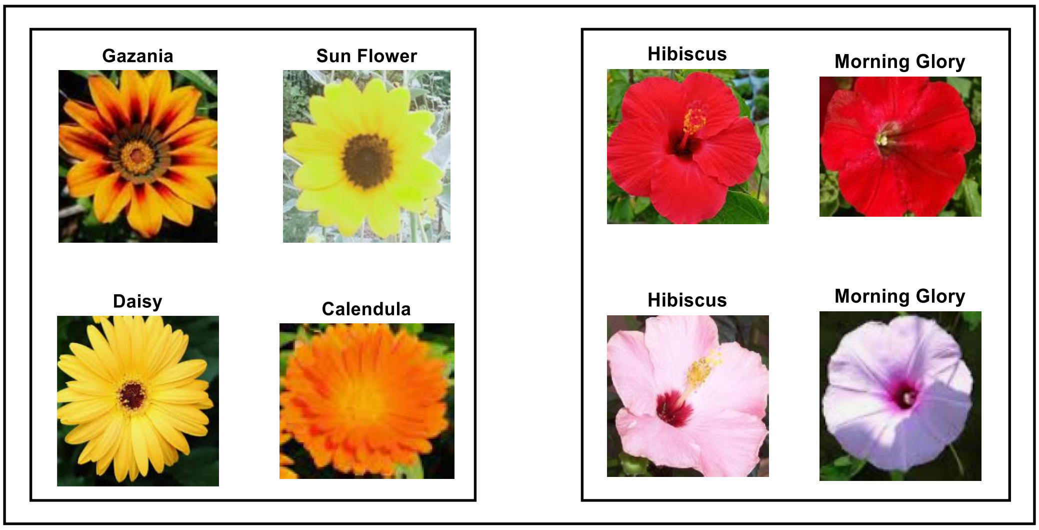


Fig. 1.2 Interclass similarities in shape of flower

Hence people often get confused while identifying a particular flower for a specific purpose. Hence an automatic flower recognition system is really needed to identify a specific flower from the image of that flower. But designing an automatic flower recognition system from the image of the flower is very difficult due to the same challenges human being faces to recognize the flowers. It is very important to identify the unique features of flower to strengthen the model for identifying flowers.

1.2. Applications

The proposed flower recognition system can play an important role in the fields of forestry informatization and plant medicine. Also, the system can be leveraged to prepare Catalogue of flowers for conservation. Farmers can also get details of a flower from any android phone to know about the flower by capturing the image of the flower without knowing anything about the flower. This will help them to decide if they are interested to grow that flower and can make contribution in horticulture.

The proposed system can be utilized to design

- Automated method of Flower harvesting using Robot based system.
- Catalogue of flowers for conservation purpose

- Design a system to provide botanical information about flowers from photos captured from Android phone to help farmers in the field of horticulture

1.3. Challenges

Flower recognition from natural images has always been a challenging task due to the wide verity of flowers in different species. Flowers from same family can be of different colour and flowers from different family can be of same colour. Similarly, there are similarities in shape of the flowers from different flower family. Also, natural images include the surrounding background like leave, branches etc. along with the flower (ROI). Automatic flower recognition system has been developed using several sets of features such as colour, texture, shape etc. But it is observed that one particular feature set may work for identifying a particular species of flowers but may not work well for other species. Similarities between different classes, and dissimilarities among the same class made the flower recognition system one of the most challenging works in the image processing field. Input images need to be pre-processed properly in order to segment out the foreground (ROI) from the background so that recognition could be done more accurately. Also, combinations of different features need to be considered to identify different species of flowers. Apart from that another major challenge is to determine the portion of the image where the actual flower is displayed as the images from natural sources contains the background of the flower like leave, branches, etc. from the plant. In the present work a new approach of flower recognition is proposed which considers the central part of the flower image addressing some of the challenges mentioned above and improve the reliability of the system. The proposed system extracts statistical colour feature, texture feature, global GIST feature and Hue’s invariant feature from the central part of the flower and SURF feature from the RGB flower image. Eleven classification models are designed based on the different combinations of the features and two different classifiers Support Vector Machine (SVM) and K-Nearest Neighbour

(KNN). The system extracts the relevant features from the entire RGB flower image. From each of the eleven classification models one predicted class is suggested for the unknown input flower image. The final prediction is done by another decision engine based on the maximum occurrence of the predicted class.

1.4. Problem Statement

Design and develop an improved flower recognition system using different inherent features of the flower to identify wide variety of flowers and overcome the limitations of the previous work done in the field of computer vision-based flower recognition system.

1.5. Objectives

The objectives of the present work are as follows.

i. To study the existing techniques of flower classification and modify them to overcome their limitations.

ii. To identify the best features for flower classification that will subsequently solve other problems.

iii. To develop an improved flower recognition system to overcome the limitations of previous work done in this field.

iv. To validate the system with different sets of experimental data along with the data available in open literature.

v. To validate the system with wide variety of flower species.

vi. To compare the performance of the proposed system against different existing models of flower recognition available in literature.

1.6. Organization of the Thesis

In the thesis, different chapters are provided for each phase of the work done.

Chapter 1 explains the overview of the proposed work, general applications, challenges and objectives of the present work.

Chapter 2 presents related literature review and the previous works carried out by other researchers in this field. It summarizes the contribution of other researchers to identify the scope of the present work.

Chapter 3 presents the proposed approach of the improved flower recognition system and solution methodology of the system.

Chapter 4 presents the different experimentations using different techniques of flower recognition system and the corresponding result analysis.

Chapter 5 explains the comparative analysis against the different models proposed by other researchers and the present work.

Chapter 6 highlights the conclusions and scope of future work.

Chapter 2

LITERATURE SURVEY

2.1. Review of Literature

Computer vision and image processing is not a new field in the present era. Now a day's lots of objects are identified and sorted to not only to minimize the human effort but also to improve system efficiency. But it is more convenient to identify a machine-made product compared to natural objects. Recognition of flowers using computer vision, image processing and machine learning is identified one of research area as per literature review though initial development was started long back.

Das *et al.* (1999) [1] proposed a method for indexing flowers that included domain knowledge with a colour-based algorithm. They used to categorise different flower species only on the basis of colour. An automatic iterative segmentation method with domain knowledge-driven feedback was used to separate the flower region from the backdrop. However, it was limited in its ability to distinguish a wide range of flowers and species with similar colours.

Later, **Saitoh and Kaneto (2003) [2]** created an autonomous wild flower recognition system based on two input images: flower and leaf. A clustering algorithm was used to extract the leaf and bloom from each image. After discovering 10 characteristics in the photo of a flower and 11 features in the picture of a leaf, recognition was performed using a piecewise linear discriminant function. They tested 20 sets of 34 wild flower species and found that when all of the traits (21) were applied, the recognition percentage was 96.0 percent. Limited datasets were used for testing and research.

Nilsback and Zisserman (2008) [3] created an automated floral classification system that can handle a large number of different classes. They extracted four separate attributes for the flowers, each of which described a different aspect: local shape/texture, boundary shape, overall spatial distribution of petals, and colour. They used a multiple kernel framework and an SVM classifier to merge the features. Learning the greatest kernel combination of several characteristics greatly improves performance, going from 55.1 percent for the best single feature to 72.8 percent for the combination of all features, according to the findings.

Using a K-nearest neighbour (KNN) classifier, **Guru *et al.* (2010) [4]** suggested a method for automatically classifying floral photos. The texture feature was recovered from a picture using Gray Level Co-occurrence Matrix (GLCM) and Gabor, or a mixture of both, in their research. Their own collection of 25 classes, each containing 50 flower photos, was used to classify them using k-nearest neighbour. They discovered that when comparing individual features, the combined features performed better. When compared to other results in the literature, they found that only texture features showed acceptable classification accuracy. However, colour, form, and other textural characteristics were not taken into account in their work.

Mohd-Ekhsan *et al.* (2010) [5] suggested an improved technique to remove the extraneous data obtained during the feature extraction phase before employing a neural network to classify the images. Flower picture categorization was defined and described in their study using low-level elements such as colour and texture to identify and explain the image content. The classification performance of the dataset, which consists of 180 patterns with 7 attributes for each species of flower, is also examined in this study using a backpropagation neural network. The initial categorization result revealed a very low accuracy percentage. As a result, additional research was carried out, including various trials, to determine the source of the low accuracy percentage. The findings and analyses revealed that a large enough image dataset has an impact on categorization accuracy.

Hsu *et al.* (2011) [6] introduced an interactive flower recognition system in which photos acquired with a digital camera were segmented using a bounding box specified by the users. The floral boundary was correctly traced using a boundary tracing method. Not only were colour and shape characteristics retrieved from the flower region, but also from the pistil/stamen sections. They found that their suggested method surpasses existing systems in terms of recognition rate when tested on two datasets with 24 and 102 species, respectively.

Hong and Choi (2012) [7] used edge and colour-based contour detection to develop an autonomous flower recognition system for smart phone users. Color groups and K-means clustering were used to classify the data. For 500 pictures from 100 species, they had a 94.8 percent success rate. Flowers with a larger and more irregular range of forms and patterns have lower accuracy.

Using an Artificial Neural Network (ANN) classifier, **Mukane and Kendule (2013) [8]** classified flowers based on textural features like gray level co-occurrence matrix (GLCM) and discrete wavelet transform (DWT). A threshold-based approach was used to segment the flower image, and a 50-sample ANN was trained to classify five different types of flowers. Using only GLCM characteristics, they were able to achieve classification accuracy of above 85%.

The concept of flower recognition using colour and edge properties of floral photos was established by **Tiay *et al.* (2014) [9].** The colour and edge properties were determined using Hu's seven-moment technique and histogram. They used K-nearest neighbour (KNN) for classification. KNN algorithm is a popular non-parametric algorithm used for solving classification as well as regression problems. The inputs to KNN are training feature vector and testing feature vector and the output of a KNN classifier is the predicted class. The object is classified based on the majority vote of its neighbours and the object is assigned to the class that consists of the highest number of common elements among its nearest neighbours. K is the number of neighbours and it is a positive integer. In KNN, one member of the test set is compared against the entire train set. The class for that member of the test set is allocated to the train set class with which it receives the greatest number of nearest neighbours as shown in **Fig.2.1**. The algorithm consists of the below steps:

Step1. Select the number of neighbours (K).

Step2. Calculate the Euclidian distance of K nearest neighbours from the member of test data to be classified. Euclidian distance (P, Q) can be computed as in Eq. (2.1).

$$(P, Q) = \sqrt{\sum_{1}^{n}(pi - qi)^2} \tag{2.1}$$

Here P and Q are two n-dimensional vectors where P represents the training feature and Q represents the testing feature. P and Q can be represented as below.

$$P = \{p1, p2, \dots, pn\}$$

$$Q = \{q1, q2, \dots, qn\}$$

Step3. The new data point is assigned to the category where it finds the maximum neighbours.

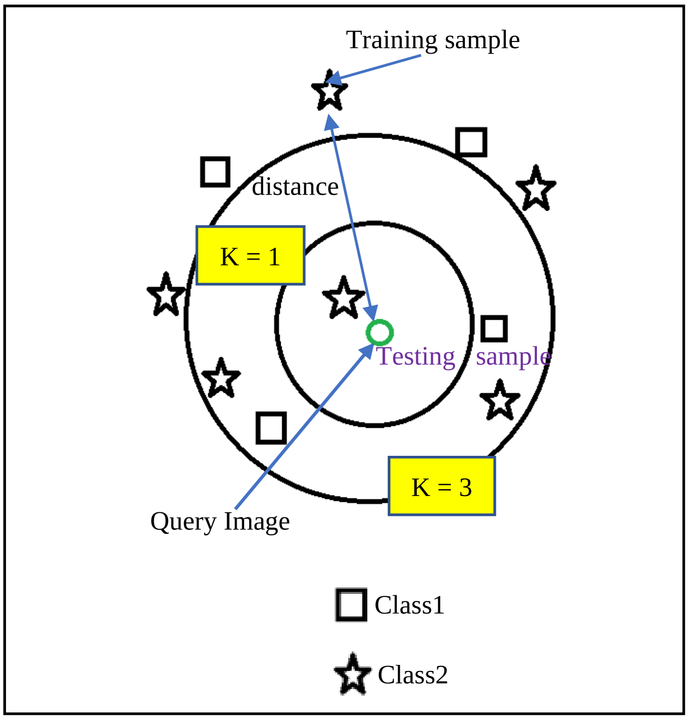


Fig. 2.1 KNN classifier

he system's accuracy was over 80%. However, it has been discovered that flowers of similar colour and shape have poorer accuracy.

Based on SURF detectors, **Ben Mabrouk *et al.* (2014) [10]** developed a new approach for colour feature extraction. They used concise and precise descriptions to describe the visual content of the floral photos. These characteristics were merged, and the learning process was carried out with the help of a multiple kernel framework and an SVM classifier. The proposed method was evaluated on a dataset given by the University of Oxford and produced superior results in terms of classification rate and execution time than the version of **Nilsback and Zisserman's (2008) [3]** method.

Lodh and Parekh (2017) [11] came up with a flower recognition system by segmenting the flower images from their natural background. They devised and implemented a colour segmentation approach based on average colour and colour distribution variation. Using a combination of colour and GIST properties, flower photos were encoded into mathematical feature vectors. Individually and in combination, the elements, as well as a variety of other visual attributes, were put to the test. It was discovered that the recommended representation is the most effective for the task. Finally, a Support Vector Machine (SVM)-based classification model was trained. Support Vector Machine (SVM) is a supervised machine learning algorithm which is used for classification problems and below mentioned steps are followed in SVM.

Step1. Plot each data item as a point in an n-dimensional space (where n is number of features) with the value of each feature being the value of a particular coordinate.

Step2. Choose the hyper plane in such a way that the sum of distances between two boundary points of two classes has to be maximum.

Step3. Select the hyper-plane that differentiates the two classes with the maximum margin (**Fig. 2.2**).

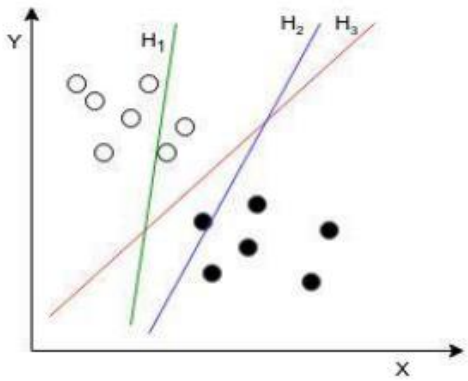


Fig. 2.2 SVM classifier (Among the three hyper-planes (H1, H2 and H3) H3 separates the training data with maximum margin.)

The trained model can recognise 12 different flower classifications. The proposed method was found to be accurate to the tune of 85.93 percent.

Shaparia *et al.* (2017) [12] used texture feature and colour features for flower classification using a standard database of flowers for experiments. The segmented photos are used to extract texture and colour information. The colour feature was retrieved using the colour moment and the texture feature was extracted using the GLCM (Gray Level Co-occurrence Matrix) approach. A neural network classifier was utilised for classification. The system's overall accuracy was 95.0 percent.

Arafah and Moghli (2017) [13] proposed an image identification technique based on efficient invariant moments and Principle Component Analysis (PCA) for grey and colour images with varying numbers of invariant moments for grey and colour images. They employed twelve moments for each grey image and Hu's seven moments for colour photos to reduce the problem's dimensionality to six PCAs for grey and five PCAs for colour images, and therefore the recognition time. PCA was subsequently used to reduce the problem's dimensionality and, as a result, the recognition time, which was our primary goal. Karhunen-transformation Loeve's was used to create the PCA. PCA tends to locate a K-dimensional subspace whose basis vectors match to the largest variance direction in the original image space given an

N-dimensional vector representation of each image. Normally, this new subspace would be lower dimensions (KN). They did a compromise between the quality of recognition and the average time recognition for grey dataset images in the experimental findings with less than six PCAs. When the number of PCAs was reduced, time and storage space were immediately influenced, and the recognition rate was reduced with less than six PCAs. They found that Hu's goals were met with colour photos, that the recognition process was faster than previously, that 5 PCAs were saved, and that storage space was lowered more than before.

Based on a multi-layer technique, **Lee and Hong (2017) [14]** proposed a method for identifying and retrieving a flower species in the natural environment. The goal of the experiment was to improve the most efficient method of feature extraction for colour, texture, and shape. They eventually came up with a flower-image automatic-recognition technique that could be used in a mobile setting. Between 2011 and 2014, they experimented on 29,463 photos of 300 species of blooming flowers taken in South Korea. They discovered that the first-ranking recognition of the floral image was 91.26 percent, and the sixth-ranking recognition was 97.40 percent. The colour, texture and shape elements of the flower photos were found to be the most effective.

Hiary *et al.* (2018) [15] proposed a novel two-step deep learning classifier for distinguishing wide range of flowers species. First, the floral region was automatically segmented to allow the minimal bounding box to be located around it. In a fully convolutional network framework, the proposed flower segmentation method was treated as a binary classifier. Second, they create a robust convolutional neural network classifier to differentiate between the various flower varieties. They recommended new steps for ensuring robust, accurate, and real-time classification throughout the training stage. Three well-known flower datasets were used to test their strategy. On all datasets, their classification results above 97 percent, which was better than the state-of-the-art in this field.

Cibuk *et al.* (2019) [16] developed a hybrid method based on deep convolutional neural networks (DCNN) to categorise floral species. Initially, they used a pre-trained DCNN model for feature extraction, but later, they used two popular DCNN architectures, AlexNet and VGG16, to create efficient feature sets and features. Finally, the more efficient features were chosen using a feature selection process called the minimal Redundancy Maximum Relevance (mRMR) method. The collected features were used to categorise the flower species using a support vector machine (SVM) classifier with a Radial Bases Function (RBF) kernel. The Flower17 and Flower102 datasets, both of which contain a large number of categories, were used in the experiments. The accuracy performance of Flower17 was 96.39% and accuracy performance of Flower102 was 95.70%.

Using LBP and SURF features and SVM as a classifier, **Dhar (2019) [17]** developed a system to classify flowers. The input image was pre-processed to improve image quality. After that, using the active contour segmentation approach, the image was segmented. LBP and SURF characteristics were extracted after the image was segmented. MSER regions were used to extract SURF features. After that, both features were combined. The SVM classifier was used to classify the concatenated features. To train these features and categorise them, a quadratic SVM was used. They also experimented with alternative classifiers, but the results were disappointing. The accuracy of the proposed quadratic SVM is 87.2 percent, which is considerable and comparable for this classification problem.

To tackle the challenge of huge inter-class similarity and inner-class difference, **Yangyang and Xiang (2019) [18]** suggested a novel approach based on Saliency Map and PCANet. They employed a combination of saliency and gray-scale maps to select flowers and used the flower region as input to train the PCANet, a simple deep learning network for automatically learning flower features. On the Oxford 17 Flowers dataset, their method has an accuracy of 84.12%. The results demonstrated that a floral image classification problem may be solved using a combination of Saliency Map and a simple deep learning network called PCANet.

Bae *et al.* (2020) [19] introduced the multimodal convolutional neural networks model (modified m-CNN) that was originally developed for image-text matching to be used for classification of not only pictures but also texts (2018) []. To begin, visual characteristics were extracted and each word in a text was represented as a vector using pre-trained CNN and word embedding models. Second, using a CNN model for text data, textual features were retrieved. Finally, pairs of features recovered using the text and image CNNs were concatenated and fed into a convolutional layer, allowing for a better learn of the relevant feature information in the combined image and text representation. To achieve classification, features collected from the convolutional layer were fed into a fully connected layer. The suggested technique outperformed conventional data fusion algorithms for flower classification using data from photographs of flowers, according to the results of the experiments. More specifically, the suggested approach outperformed m-CNN and multimodal recurrent neural networks algorithms by 10.1 percent and 14.5 percent, respectively.

To categorise flowers, **Cao and Song (2021) [20]** devised an image recognition system based on Deep Convolutional Neural Networks (DCNN). Because building a deep learning model normally necessitates a large number of training samples, they used image rotation and cropping to enhance the training sample. As a training set, the enhanced images and the original image were combined. They suggested a visual attention-driven deep residual neural network made up of several weighted visual attention learning blocks. To improve the learning ability and discriminating ability of the entire network, each visual attention learning block was formed of a residual connection and an attention connection. Finally, in the testing set, the model was trained in the fusion training set and detected flowers. The recognition accuracy of their new method on public flowers of 17 was 85.7%.

Bozkurt (2021) [21] investigated CNN Based Transfer Learning for Flower Species Recognition using VGG16, VGG19, SqueezeNet, DenseNet-121, DenseNet-201, and InceptionResNetV2 as pretrained learning approaches. In the experimental results, their classification performances were evaluated on the identical flower dataset. In studies, it was discovered that the InceptionResNetV2 model outperforms other models. The InceptionResNetV2 model for the flower dataset has the highest accuracy (92.25 percent). The Inception-ResNet model was a cross between the Inception modules of the Inception architecture and the ResNet architecture's performance. The network was 164 layers deep, with 55,855,205 trainable parameters in the InceptionResNetV2 model. The number of trainable parameters was enhanced in their research by increasing the number of deep layers. Since number of trainable parameters increased by increasing the number of deep layers, accuracy rate of flower classification increased.

Lv *et al.* (2021) [22] created a floral classification model that incorporated saliency detection and the VGG16 convolutional neural network. To refine the model, they used a stochastic gradient descent approach that avoided over-fitting. Experiments with the Oxford flower-102 international public flower recognition data set revealed that the proposed model outperformed existing standard network models.

Tog˘açar *et al.* (2021) [23] proposed a hybrid technique for classifying flower species that used feature selection methods and Convolutional Neural Network (CNN) models. CNN models were utilised to extract features in the proposed model. These models' features were integrated, and efficient features were chosen using feature selection methods. The classification performance was compared to the results of the trials using feature selection methods. The Support Vector Machine (SVM) approach was reported to have a classification success rate of 98.91 percent.

Kaur *et al.* (2022) [24] advocated that a flower recognition system be designed to improve the strategy for applying machine learning techniques.

In their work, they used the Texture Feature, RST-Invariant Feature, Pattern Classification, and K-Closest Neighbor computations. This method uses the k-nearest neighbour image to recognise and classify sunflowers, with an overall accuracy of 88.52 percent.

For flower categorization, **Zhang *et al.* (2022) [25]** presented an attention mechanism and a multi-loss attention network. Based on Xception, they designed the embedding of the spatial attention module and channel attention model in the Xception framework. In the network loss layer, the network was optimised by combining Triplet Loss and Softmax Loss to obtain a feature embedding space with excellent intra-class compactness and inter-class separation. They tested their model on two flower image data sets (Oxford 17 flowers and Oxford 102 flowers), and found that it was 0.39 percent, 0.50 percent, and 0.72 percent higher on the Oxford 17 flowers data set, and 0.52 percent, 0.63 percent, and 0.85 percent higher on the Oxford 102 flowers data set.

2.2. Scope of the Present Work

Nowadays, researchers apply the modern sophisticated software techniques such as deep learning, AI for image-based object recognition system. But challenges are still there. Most of the testing and experimentation results are obtained based on the datasets provided for training and testing the classification models. Often, accuracy of the model differs when the data sets are changed. In the field of flower recognition, most of the data sets are obtained from natural sources that lie with a number of variations in each of the samples. Keeping view of the above fact and based on the literature survey, a tried has been made to design and develop a unique model that can provide good prediction with minimum computational effort with high reliability. The proposed model has been developed based on the different features available on literature. Different features are combined together for better classifications. Finally, the results are validated and the performance of the model is analysed.

In the present work similar approach proposed by **Lodh and Parekh [11]**, **Hsu *et al.* (2011) [6], Dhar (2019) [17]** and **Shaparia *et al.* (2017) [12]** are considered and a new model is designed to address the limitations of their model.

When the same algorithm proposed by **Lodh and Parekh [11]** is used to recognize the new dataset of 12 flowers, the solution gave less than 60% accuracy and there were many misclassifications. This model requires better segmentation technique and for many cases the segmentation was not done properly and caused error while identifying an image of a flower.

Classification algorithm proposed by **Hsu *et al.* (2011) [6]** requires segmentation and floral boundary identification which is very challenging for many of the flower species. Identifying the pistil and stamen area has its own challenges.

Shaparia *et al.* (2017) [12] used texture and colour features for flower classification using a standard database of flowers for experiments, but using these 2 features on the entire flower image was not able to identify the flowers in the new dataset of 12 different types of flowers which has similarities in their petal colours and number of petals.

Dhar (2019) [17] used LBP and SURF characteristics on the entire flower images and had good accuracy on their dataset, whereas the same feature gave more than 99% match in the interclass flowers in the new dataset and it was very difficult to determine the threshold of the SURF matched percentage.

All of the above-mentioned approaches require proper segmentation to classify the flowers. In the proposed model that challenges are mitigated by focussing on the centre part of the image and combining different features suggested by many researchers. Finally, a new model is proposed to combine the solutions suggested by other researchers to determine the highly predicted class of a flower using the centre part of the image.

Chapter 3

PROPOSED APPROACH

3.1. Motivation

Computer vision has been recognized as one of the reliable solutions for identifying flowers. The colour, shape, size and texture are most commonly used features used by traditional computer vision system. Many researchers worked on the flower recognition as it is quite critical and challenging area in the computer vision field. Even though there are good results of some of the solutions, the same solutions are not suitable for other group of flowers. Previous work done in this area mostly focused on segmented flower images to remove background from the ROI. But if we closely look at the different flower image then it is observed that even though even though there are dissimilarities in different flowers of same flower family, there are common features in the centre part of the flower of same flower family irrespective of their petal colour. Also, there are dissimilarities in the centre part of the flowers in different flower families (**Table 4.6**) which can be used to identify a particular species of flower ignoring the other parts of a flower.

It has been understood from a survey that a human usually identifies the flower from the front view of the flower and human brain mostly classifies flowers on the basis of the centre and the shape of the flower as same flower can be of different colour. Hence a robust and reliable computer vision-based flower recognition system with improved performance can be designed focussing on the centre part of the flower. This will also eliminate the challenges of segmentation because most of the cases the centre part of the flower image does not have any background as the background is already covered by the flower petals itself (**Fig. 4.14** to **Fig. 4.19**).

3.2. System Components

The proposed flower recognition system consists of five major components (Flower Image Database, Crop Module, Feature Database, Classification Model and Decision Engine. Detailed of each component is discussed as below.

3.2.1. Flower Image Database

Synthetic colour images of different species of flower are used to train the classification model. All of the images are cropped properly to ensure that the background of the flowers is discarded to the best level. The flower image database consists of different colour variations of each species of flower and different species of flowers having similarities in shape and colour. For the training and testing purpose all of the flower images are labelled against the correct class they belong to. Similarly, test dataset is prepared to test the system and calculate the accuracy of the system.

3.2.2. Crop Module

Crop module is necessary to crop the centre part of each flower images and this is essential as the proposed system analyse the centre part of each flower images and ignores the shape and petal colour of the flower. It also eliminates the need of segmentation to separate the background from the actual flower image. Each of the flower image are divided into 9 equal blocks and the centre block is considered as the ROI (Region of Interest) for the actual image processing and classification of flower.

3.2.3. Feature Database

In this phase, the images of the training database are processed and the relevant features to uniquely identify a particular species of flower is extracted from the flower image. Any colour image can be represented with many features. But all the features are not relevant for identifying a particular species of flower as same species of flower can have different colour and different species of flowers can be of same petal colour. Also, the shape of each species is not unique. Different species of flowers can have similar shape. Hence, for a flower recognition system design, features must be chosen wisely to uniquely identify the flower. During feature extraction, the colour images of different flowers from training dataset as well as testing dataset are processed and relevant features from each image is extracted and represented by a matrix or vector of a certain length. Each row of the matrix or vector is the feature data of a particular image from the flower image database. The vector representing relevant features of all samples of training database is considered as Training Feature Database. Similarly, the vector representing relevant features of all samples of testing database is considered as Training Feature Database. In the proposed model, colour features, Hues Seven Invariant Moments, texture feature, and GIST feature are extracted from the ROI (Region of Interest). Different combined feature descriptors are prepared with the combinations of different features extracted from the ROI to represent each of the flower images in the flower image database. Apart from that SURF feature is also extracted from the entire flower image to make the system more reliable.

3.2.4. Classification Model

Different classification models are designed to classify a flower from its RGB image. Each model is designed from different combinations of feature and a classifier (either SVM or KNN). In this phase the features extracted from the input flower image are compared against the Feature DB and each sample from test set is compared against all the samples of train set. Whichever sample of the training dataset will have the maximum similarity with the input image, the species of that sample will be considered as the species of the input image. As the query image can be of unknown species, a threshold is employed against a metric and a sub-decision system is designed to classify the image as known and unknown to the system. Each classification model predicts a class for the unknown input flower image. Also, another prediction is done based on the maximum match percent of the SURF feature.

3.2.5. Decision Engine

This module may be regarded as the 'decision' module of any recognition system. In this phase all of the predicted classes from each model along with the predicted class from SURF model is sent to the final decision engine to determine the highly predicted class of the flower based on the mode of all predicted class suggested by all classifications model.

3.3. Block Diagram

The block diagram of the proposed system is shown in **Fig. 3.1**.

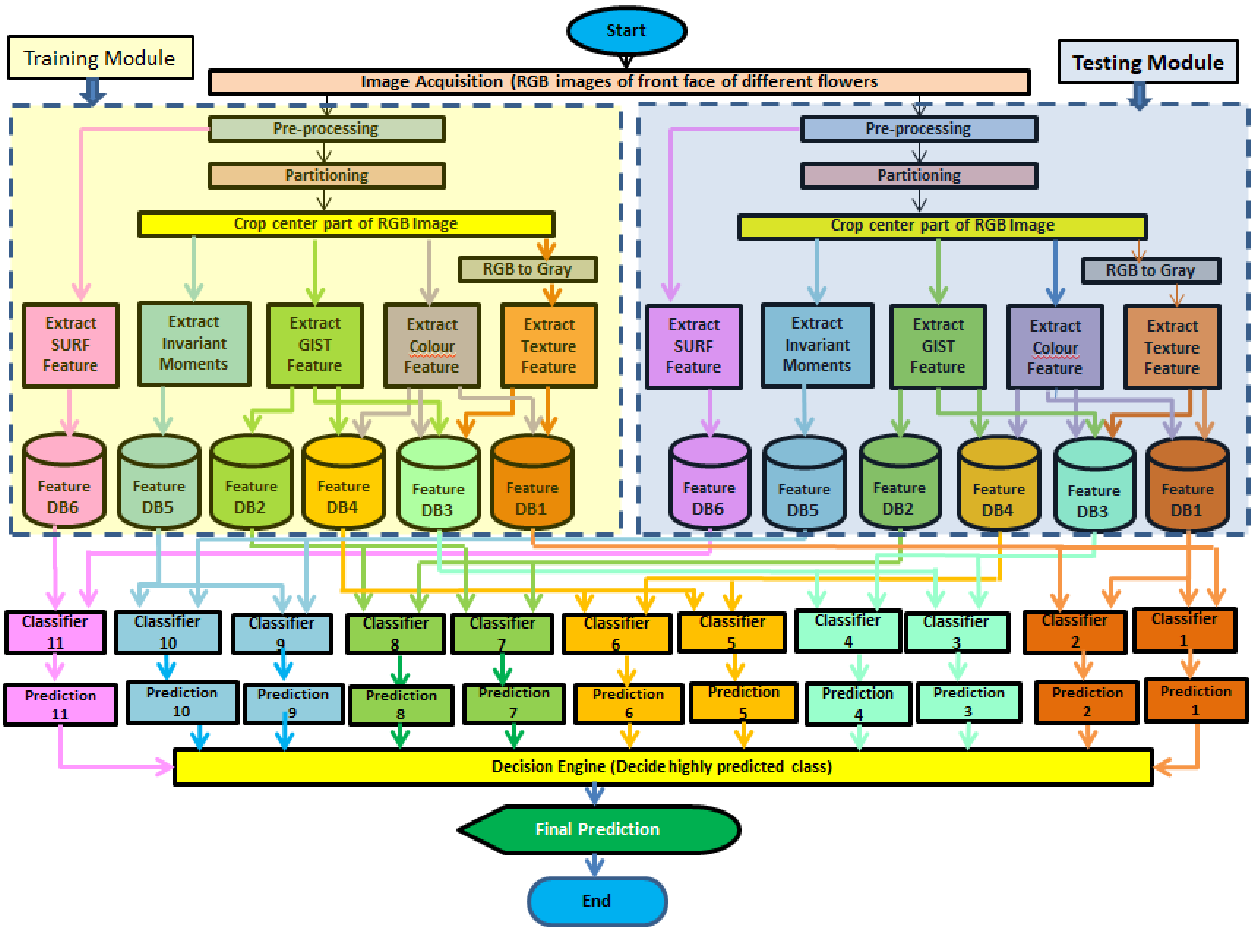


Fig. 3.1 Block diagram of the proposed system

3.4. Design

The design steps of the proposed system are described in this section.

3.4.1. Image Acquisition

In this step involves collecting medium sized synthetic colour images of top view of different types of flowers with different verity of colours in PNG format. These images are required to train and test the proposed model.

3.4.2. Pre-processing Images

To reduce computational loads, all of the RGB images are resized to increase the efficiency of the model. In the proposed model each flower image is resized to M rows x N columns dimension keeping the resolution same as the actual image. Value of M and N needs to be determined based on the dimension of the original input image to avoid any distortion due to image resize.

3.4.3. Partitioning

Partitioning is done in order to separate the centre part of the flower image. Each image is split into non-overlapping rectangular cells and the centre block is cropped for the actual image processing.

The following steps are performed to crop the centre part of the RGB images:

Step 1. Get the dimensions of the image from the RGB image.

Step 2: Get the rows and columns to split at.

Step 3. Determine the starting and ending columns of each block.

Step 4. Determine the starting and ending rows of each block.

Step 5. Crop the centre part of the input image using the starting and ending columns along with the starting and ending rows of the centre block.

3.4.4. Feature Extraction

In this module all the features chosen for flower classification are discussed. There are many features which are mostly used for colour image processing but colour, texture, GIST and SURF feature are some of the features which shows good result when it comes to flower image recognition. Hence the proposed model is designed considering these features.

3.4.4.1. Colour Moments

Colour moments are one of the commonly used in image retrieval applications since they encode both colour and shape information. The first two colour moments mean (μ), and standard deviation (σ) are scaling and rotation invariant and contains most of the colour distribution information. Hence, in this work, the aforementioned two-colour moments of 3 colour channels (Red, Green and Blue) are computed for RGB images.

Mean (μ), represents the average colour in an RGB image and mean for i-th colour channel $\mu(i)$ is computed as in Eq. (1).

$$\mu(i) = \sum_{j=1}^{N} \frac{1}{N} p(i,j) \tag{1}$$

Here, N represents the number of pixels in an RGB image, and *p(i,j)* represents the colour intensity value of the *j-th* pixel of the *i-th* colour channel.

The standard deviation (σ) is computed by taking the square root of the variance of the colour distribution and computed as in Eq. (2).

$$\sigma(i) = \sqrt{\left(\frac{1}{N}\sum_{j=1}^{N}\left(p(i.j) - \mu(i)\right)^2\right)} \tag{2}$$

Here, $\sigma(i)$ represents the standard deviation for the *i-th* colour channel and $\mu(i)$ is the mean value of the i-th colour channel of the image.

The aforementioned colour moments obtained from each of the three colour channels are combined in a single vector and a 6 elements colour feature descriptor (F_{col}) representing the RGB image is formed and represented as below in Eq. (3).

$$F_{col} = \left[\mu_r, \mu_g, \mu_b, \sigma_r, \sigma_g, \sigma_b\right] \tag{3}$$

3.4.4.2. Texture Features

Texture feature is extracted by using GLCM method on the grey scale image. Mostly two steps are performed for the Co-occurrence texture features extraction. In the first step GLCM is created a by using spatial co-occurrences of pixels in pair, which is separated by a particular angle and distance. In the second step, the computed GLCM is used to calculate the correlation, contrast, energy, homogeneity etc. In the proposed system features such as energy (G_e), homogeneity (G_h), contrast (G_c) and correlation (G_r) are considered and the same can be computed as in Eq. (4) –Eq. (8).

$$G_e = \sum_{i,j=0}^{N-1} \left(p_{ij}\right)^2 \tag{4}$$

$$G_h = \sum_{i,j=0}^{N-1} \frac{p_{ij}}{1+(i-j)^2} \tag{5}$$

$$G_c = \sum_{i,j=0}^{N-1} p_{ij}(i-j)^2 \tag{6}$$

$$G_r = \sum_{i,j=0}^{N-1} p_{ij} \frac{(i-\mu_g)(j-\mu_g)}{{\sigma_g}^2} \tag{7}$$

Where, $p_{ij}$is the Element i,j of the normalized symmetrical GLCM

N is Number of grey levels in the image.

$\mu_{glcm}$is the GLCM mean and calculated as Eq. (8):

$$\mu_{glcm} = \sum_{i,j=0}^{N-1} i p_{ij} \tag{8}$$

${\sigma_{glcm}}^2$is the variance of the intensities of all reference pixels and calculated as Eq. (9):

$${\sigma_{glcm}}^2 = \sum_{i,j=0}^{N-1} p_{ij}(i-\mu_{glcm})^2 \tag{9}$$

Finally, GLCM feature descriptor can be represented as eq. (10)

$$F_{glcm} = [G_e, G_h, G_c, G_r] \tag{10}$$

3.4.4.3. GIST Descriptor

GIST descriptors were proposed by **Oliva and Torralba [26]** in 2001 to recognition of real-world scenes. It is used to represent a low-dimensional image with enough information to identify the scene in it. Using GIST an image to be represented in a very tiny size. GIST descriptor is computed by convolving the image with 32 Gabor filters at 4 scales and 8 orientations. It produces 32 feature maps of same size as the input image. Then each feature map is divided into 16 regions by a 4×4 grid, and the average value for each region is computed. Finally, by concatenating the 16 averaged values for each of the 32 feature maps, a 512 element GIST descriptor (F_{gist}) is obtained. Thus, GIST provides a description of the scene by summarizing the gradient information of different parts of the RGB image.

3.4.4.4. SURF Feature

Speeded-Up Robust Features (SURF) is a fast and reliable algorithm for representing and comparing images in a local, similarity invariant manner. In SURF interest points of a given image are considered as salient features (F_{surf}). SURF is a more advanced variant of the SIFT technique. SURF outperforms SIFT in terms of processing speed. When utilising SURF, a rotated object cannot have a negative effect. This is a significant benefit of the SURF function. Sub-areas of equal size are created by separating nearby regions. A calculation of Haar-wavelet responses is performed for each region. From the extracted feature vector, this system extracted the strongest characteristics.

3.4.4.5. Hue's Seven invariant moments

Hue's invariant moments are invariant with respect to translation, scale, and rotation. The following formula is used to compute the Seven Moments: (Eq.11-17)

$$I_1 = \eta_{20} + \eta_{02} \tag{11}$$

$$I_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \tag{12}$$

$$I_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \quad (13)$$

$$I_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \quad (14)$$

$$I_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \quad (15)$$

$$I_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \quad (16)$$

$$I_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2 - (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \quad (17)$$

The central moments μ_{ij} of any order are invariant with respect to translations by definition. Where invariants η_{ij} can be generated from central moments by dividing via a suitably scaled zero-th central moment in terms of both translation and scale:

$$\eta_{ij} = \frac{\mu_{ij}}{\mu_{00}^{\left(1+\frac{i+j}{2}\right)}} \quad (18)$$

Hue's invariant moment feature descriptor (F_{im}) can be represented as eq. (19)

$$F_{im} = [I_1, I_2, I_3, I_4, I_5, I_6, I_7] \quad (19)$$

3.4.4.6. Combined Features

Combined features of GLCM Statistics (F_{glcm}) Colour moments (F_{col}), and GIST (F_{gist}) are considered in a single feature descriptor (5). There are multiple combined feature vectors in the proposed work.

$$F_{comb1} = [F_{glcm}, F_{col}] \quad (20)$$

$$F_{comb2} = [F_{gist}] \quad (21)$$

$$F_{comb3} = [F_{glcm}, F_{col}, F_{gist}] \quad (22)$$

$$F_{comb4} = [F_{col}, F_{gist}] \quad (23)$$

$$F_{comb} = [F_{im}] \quad (24)$$

3.5. Create Training and Testing Feature Database

Once the features are extracted from all of the images from training database, a feature vector is created to train the flower classification model using the training feature set. Similarly, the features are extracted from all of the images from testing database, a feature vector is created to test the flower classification model using the testing feature set.

3.6. Classification

Both Support Vector Machine (SVM) and K- Nearest Neighbour (KNN) are used to train the classification model using the training features and different classification models are designed by varying the features of flower images. Below are the details of each model designed in the proposed solution.

Chapter 4

EXPERIMENTATIONS AND RESULTS

This chapter depicts all experimentations carried out in the present work using different features and classifiers to recognize flower. Results obtained from different experimentations are analysed and discussed. The experiments are largely carried out in the MATLAB R2018a environment.

The flower image dataset used in the present work are the medium sized flower image taken from standard database of google. Each colour images are edited using photo editing tool (photoshop) to remove the background and crop the flower part from the image. The dataset contains 12 different classes of flowers (Hibiscus, Rose, Poppy, Daisy, Calla Lily, Plumeria, Morning Glory, Marigold, Tagar(crape jasmine), Sunflower, Gazania and Calendula). The dataset contains both inter-class similarities and intraclass differences as shown in **Fig. 1.1** and **Fig. 1.2** The separation ratio of training and testing dataset is considered as 5:1. Training dataset consists of 45 medium sized images of each flower species with different colour variation of each species. Testing dataset consists of 9 medium sized images of each flower species with different colour variation of each species.

Three rounds of experiment are conducted to identify the best model among different classification models. Detailed steps of each experiment have been presented in this section.

4.1. Experimentations of flower recognition considering the entire flower image

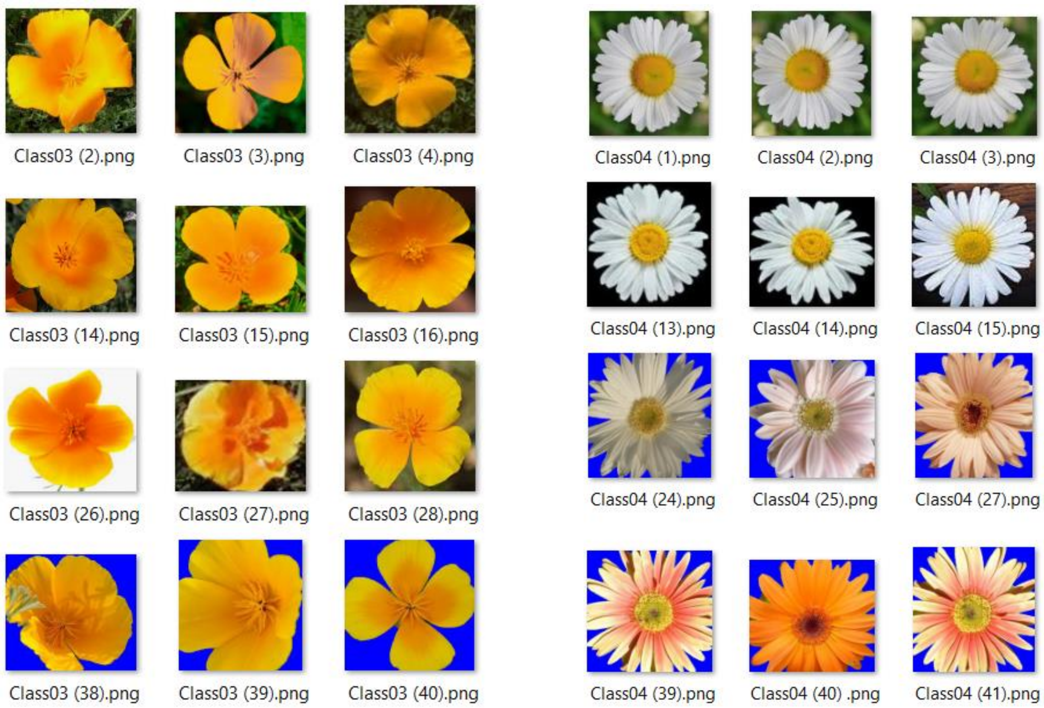
In this experiment details description of each step performed during the test is described and the results are analysed.

4.1.1. Image Acquisition

648 synthetic colour images of top view of 12 different types of flowers with different verity are saved in PNG format. Training dataset is prepared with 45 samples of each flower class to train the flower classification model. Similarly, testing dataset is prepared with 9 samples of each flower class to test the flower classification model (**Fig. 4.1** to **Fig. 4.6**).

(a) (b)

Fig. 4.1 (a &b) Sample flowers for training and test database (Class01: Hibiscus and Class02: Rose)



(a) (b)

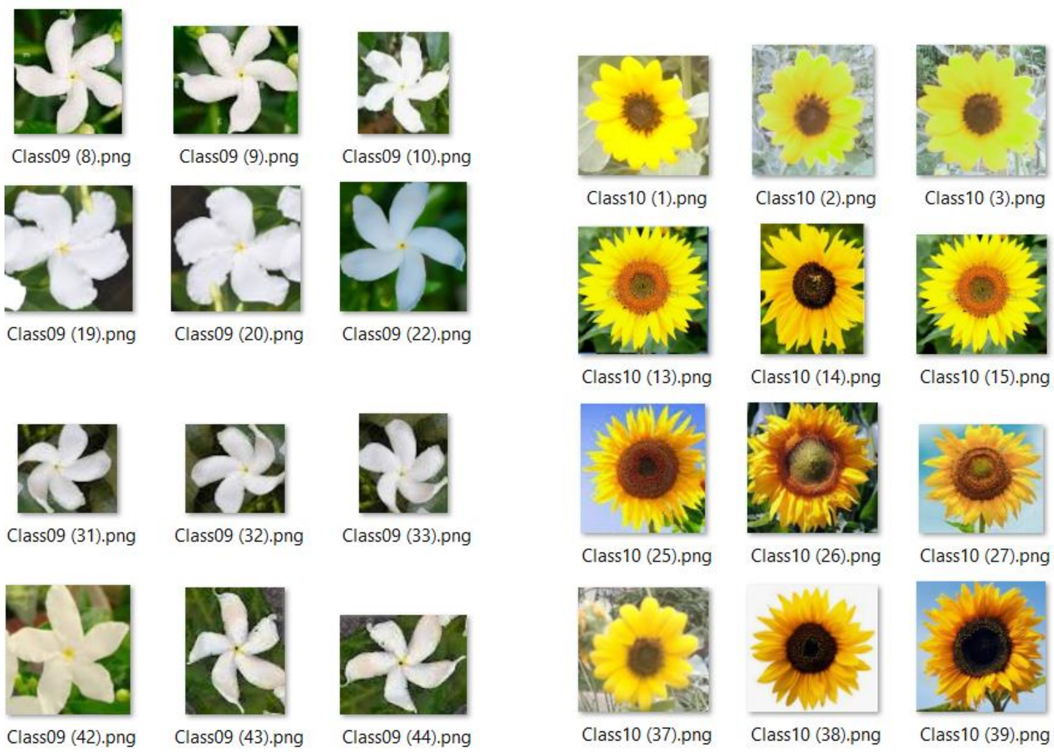
Fig. 4.2 (a &b) Sample flowers for training and test database (Class03: Poppy and Class04: Daisy)

(a) (b)

Fig. 4.3 (a &b) Sample flowers for training and test database (Class05: Calla Lily and Class06: Plumeria)

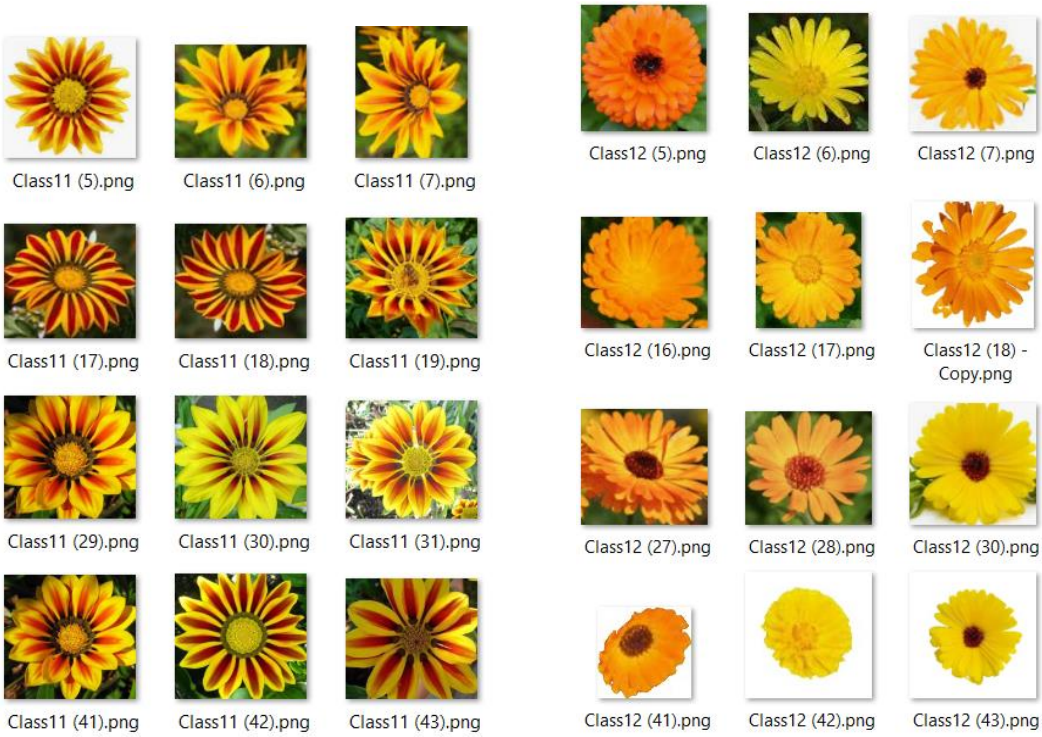
(a) (b)

Fig. 4.4 (a &b) Sample flowers for training and test database (Class07: Morning Glory and Class08: Marigold)



(a) (b)

Fig. 4.5 (a &b) Sample flowers for training and test database (Class09: Tagar and Class10: Sunflower)



(a) (b)

Fig. 4.6 (a &b) Sample flowers for training and test database (Class11: Gazania and Class12: Calendula)

4.1.2. Pre-processing Images

To reduce computational loads, all of the RGB images are resized to 300 rows x N columns dimension keeping the resolution same as the actual image. Value of N needs to be determined based on the dimension of the original input image to avoid any distortion due to image resize. All of the images are segmented to remove the background from the ROI (Region of interest)

4.1.3. Feature Extraction

From each image in both training dataset and testing dataset colour features, texture feature, GIST feature, SURF Feature are extracted and 5 different feature DB is created combining different features as shown in the table (**Table. 4.1**)

Table. 4.1 Feature DB created for Experiment1

Feature DB	**Number of Predictors**	**Feature**
$\mathbf{F_{comb}1}$	10	$G_e, G_h, G_c, G_r, \mu_r, \mu_g, \mu_b, \sigma_r, \sigma_g, \sigma_b$
$\mathbf{F_{comb}2}$	512	F_{gist}
$\mathbf{F_{comb}3}$	522	$G_e, G_h, G_c, G_r, \mu_r, \mu_g, \mu_b, \sigma_r, \sigma_g, \sigma_b, F_{gist}$
$\mathbf{F_{comb}4}$	518	$\mu_r, \mu_g, \mu_b, \sigma_r, \sigma_g, \sigma_b, F_{gist}$
$\mathbf{F_{comb}5}$	7	$I_1, I_2, I_3, I_4, I_5, I_6, I_7$
$\mathbf{F_{comb}6}$	10 strongest points	F_{surf}

4.1.4. Separating the Feature Database

The feature database is divided into two parts, in 5:1 ratio, for training and testing. The classifier is trained using the training feature set and the test feature set used to test the accuracy of the classification model. 45 flowers of each class are used for training the model and 9 flowers of each class are used for testing the model. The principal components are identified for all of the combined feature vectors and the training samples are plotted in a two-dimensional space to observe the training dataset as shown in **Fig. 4.7** to **Fig. 4.11.** Similarly, the maximum and minimum % match of SURF features is calculated against each response class in the training samples to observe the similarities among all flower classes as depicted in **Fig. 4.12(a)** to **Fig. 4.12(l).** It is observed that the maximum match % among the training samples in same class is 100% and the minimum match % among the training samples in same class is approximate 96%. Whereas the maximum match % among the training samples in different class is 99.9% which is much greater than the minimum match % of the samples in same class.

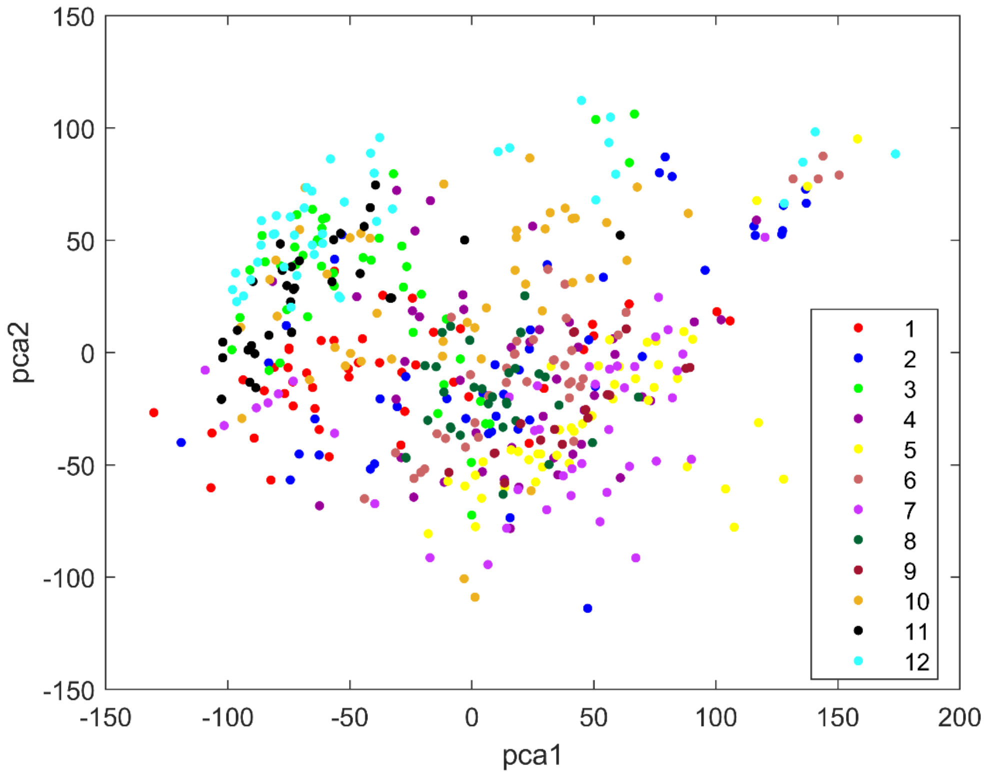


Fig. 4.7 Visualizing the training data using F_{Comb1} for entire flower image (1-12 represents the flower class index)

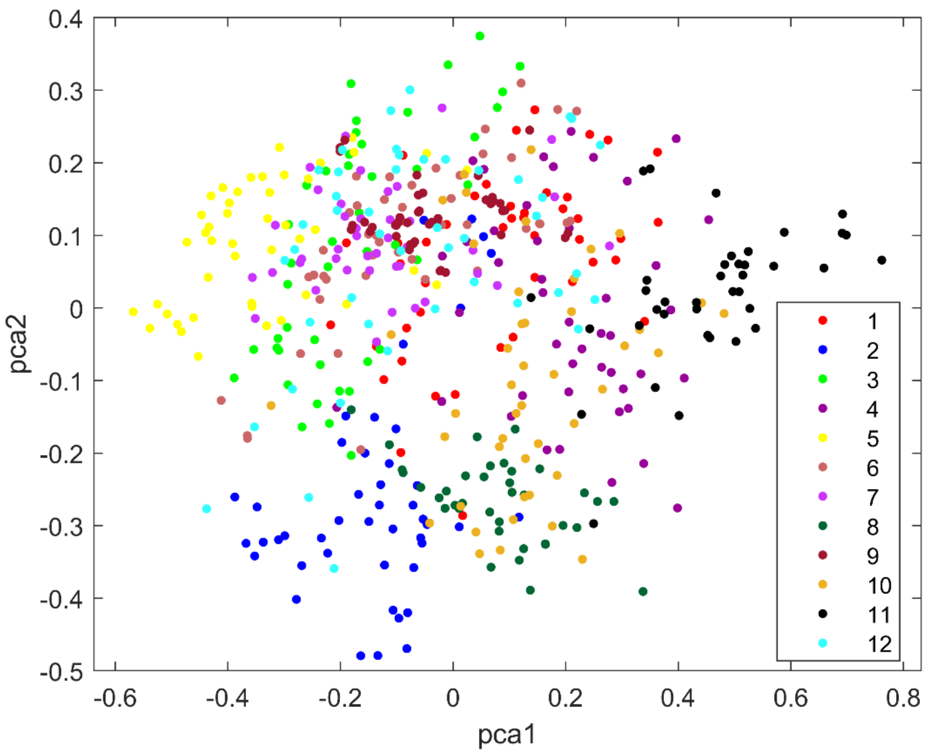


Fig. 4.8 Visualizing the training data using F_{Comb2} for entire flower image (1-12 represents the flower class index)

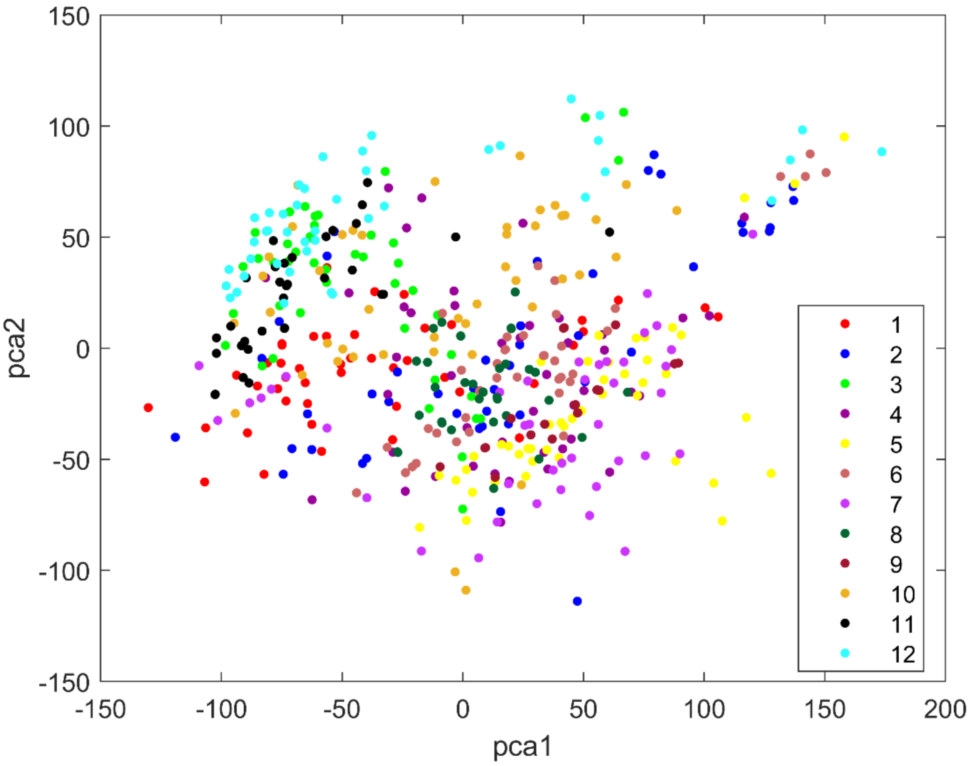


Fig. 4.9 Visualizing the training data using F_{Comb3} for entire flower image (1-12 represents the flower class index)

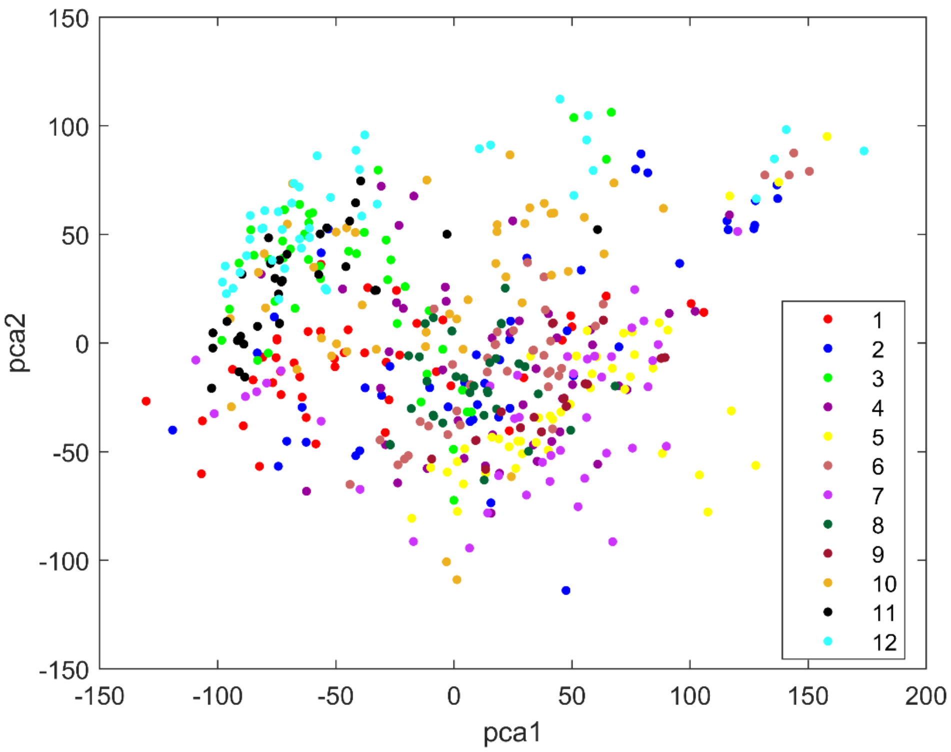


Fig. 4.10 Visualizing the training data using F_{Comb4} for entire flower image (1-12 represents the flower class index)

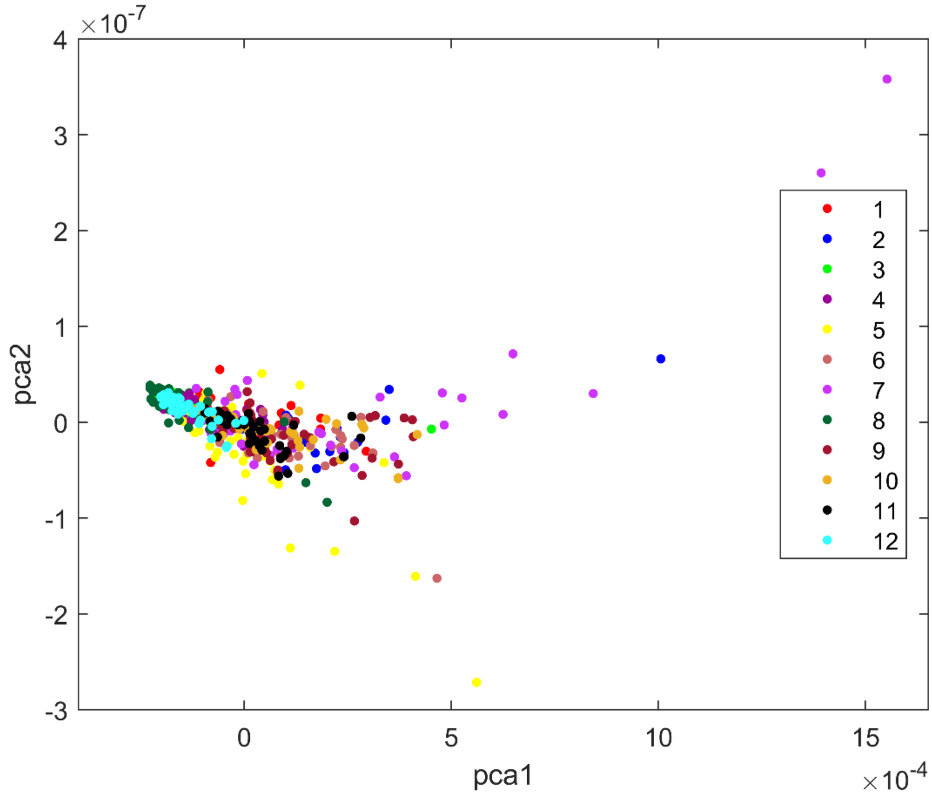


Fig. 4.11 Visualizing the training data using F_{Comb5} for entire flower image (1-12 represents the flower class index)

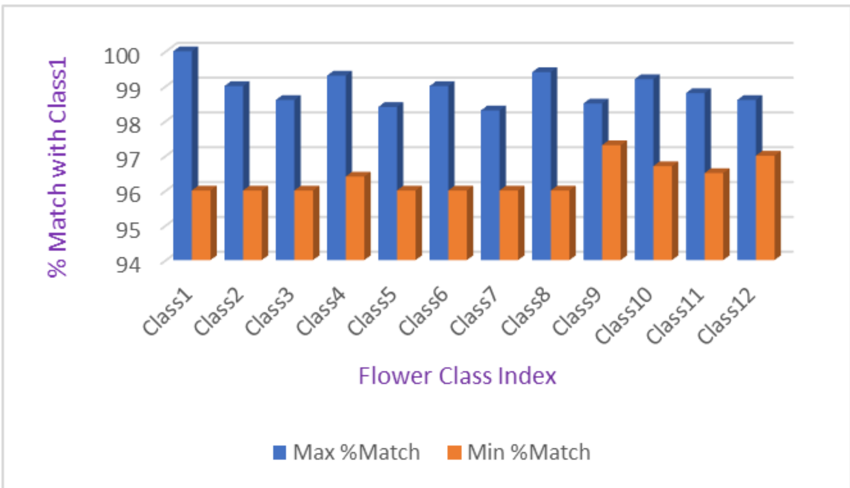


Fig. 4.12(a) Visualizing the max.% match and min.% match of training data against class1(Hibiscus) using SURF feature for entire flower image

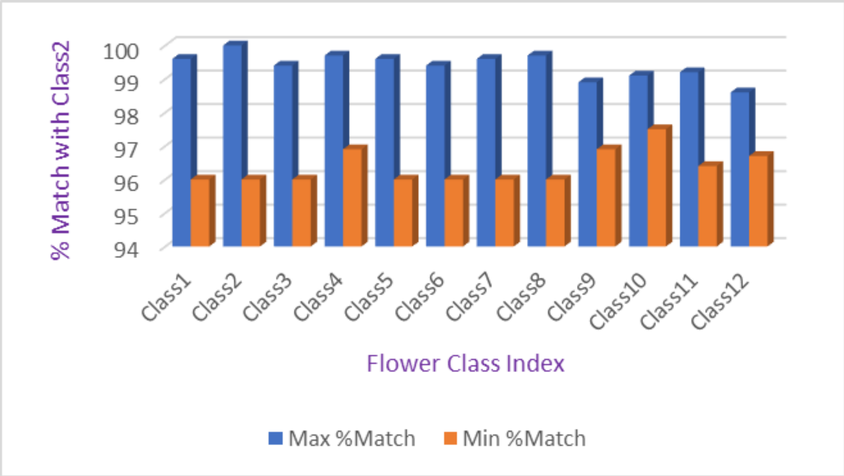


Fig. 4.12(b) Visualizing the max.% match and min.% match of training data against class2(Rose) using SURF feature for entire flower image

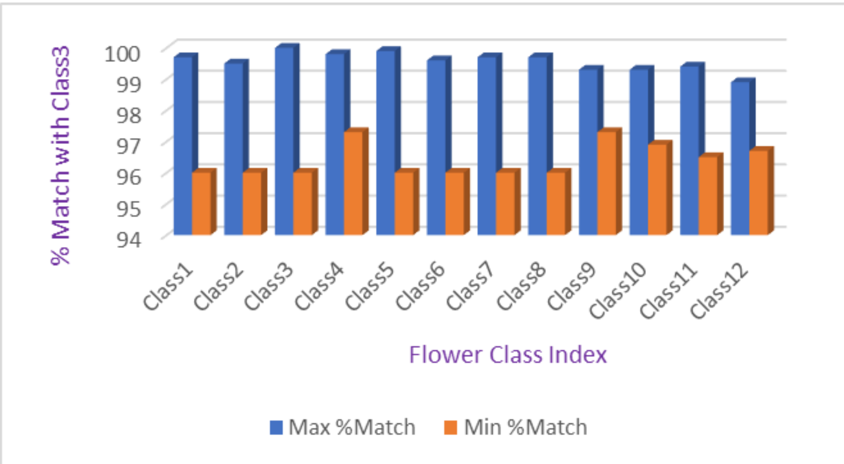


Fig. 4.12(c) Visualizing the max.% match and min.% match of training data against class3(Poppy) using SURF feature for entire flower image

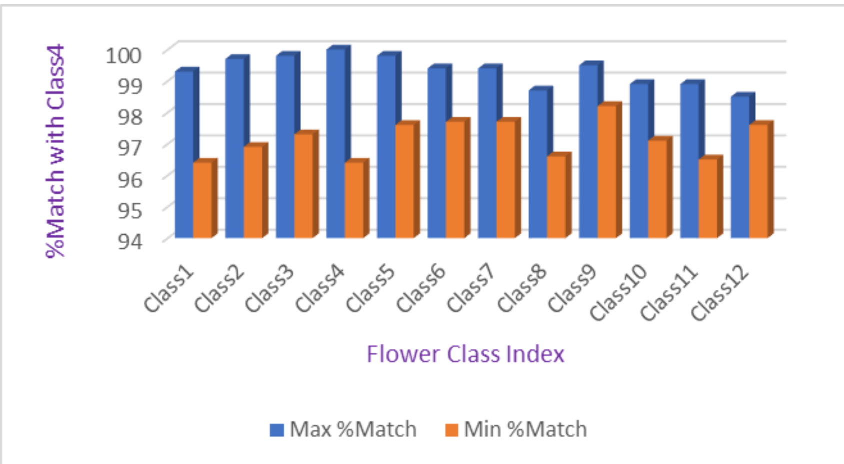


Fig. 4.12(d) Visualizing the max.% match and min.% match of training data against class4(Daisy) using SURF feature for entire flower image

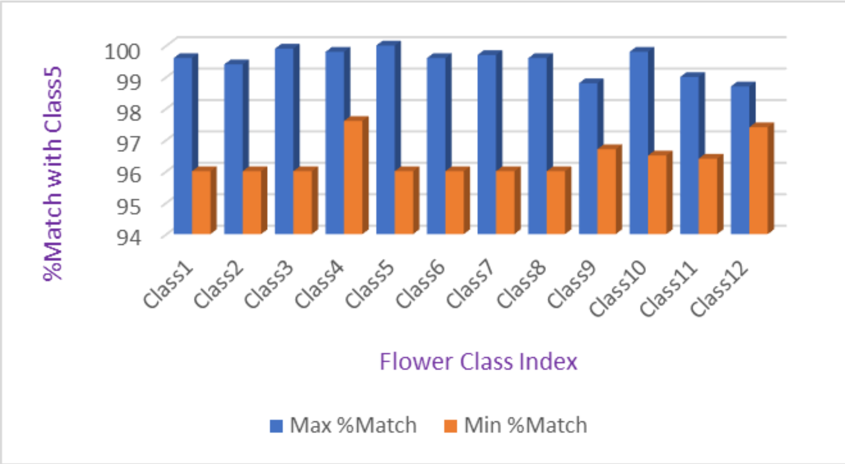


Fig. 4.12(e) Visualizing the max.% match and min.% match of training data against class5(Calla Lily) using SURF feature for entire flower image

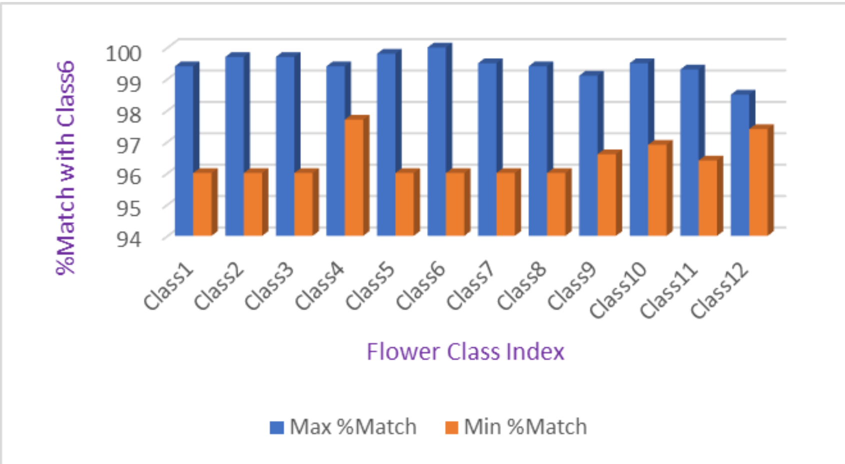


Fig. 4.12(f) Visualizing the max.% match and min.% match of training data against class6(Plumeria) using SURF feature for entire flower image

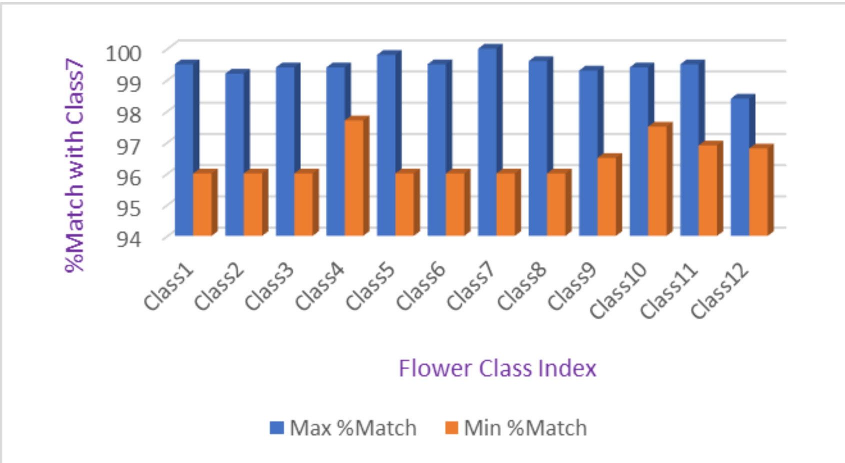


Fig. 4.12(g) Visualizing the max.% match and min.% match of training data against class7(Morning Glory) using SURF feature for entire flower image

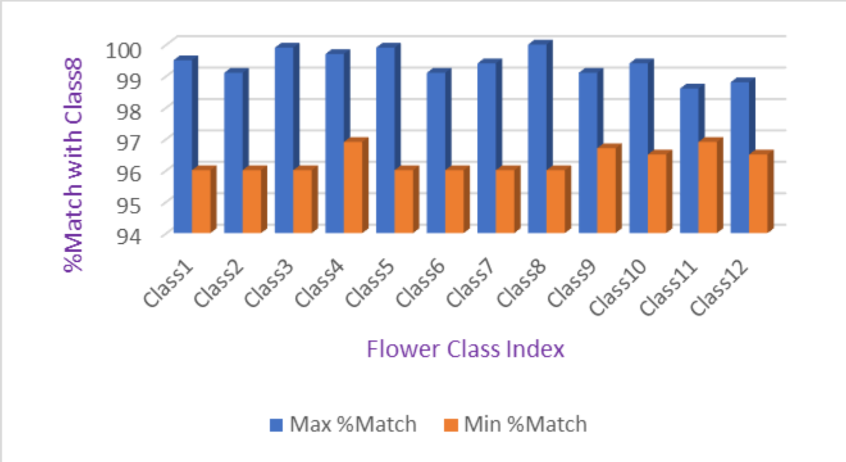


Fig. 4.12(h) Visualizing the max.% match and min.% match of training data against class8 (Marigold) using SURF feature for entire flower image

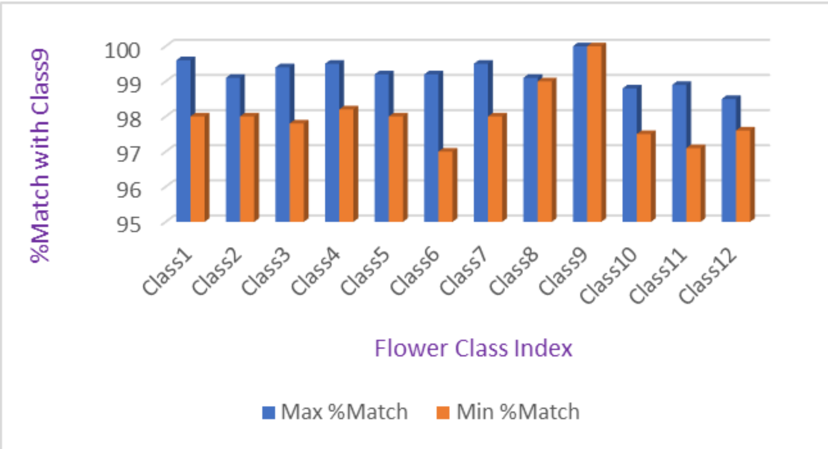


Fig. 4.12(i) Visualizing the max.% match and min.% match of training data against class9 (Tagar) using SURF feature for entire flower image

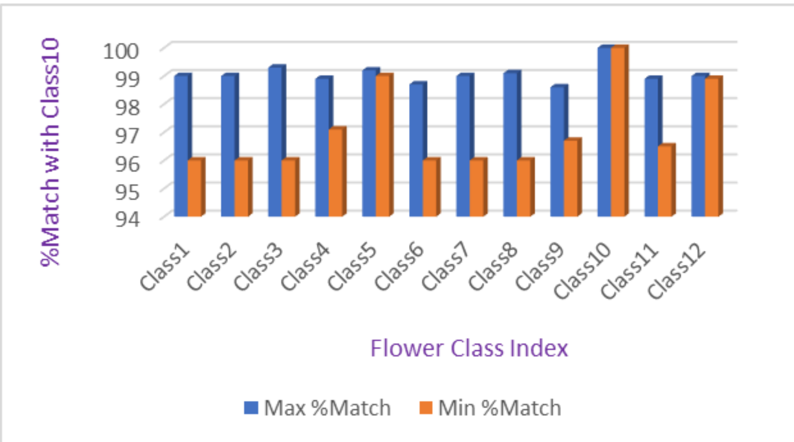


Fig. 4.12(j) Visualizing the max.% match and min.% match of training data against class10 (Sunflower) using SURF feature for entire flower image

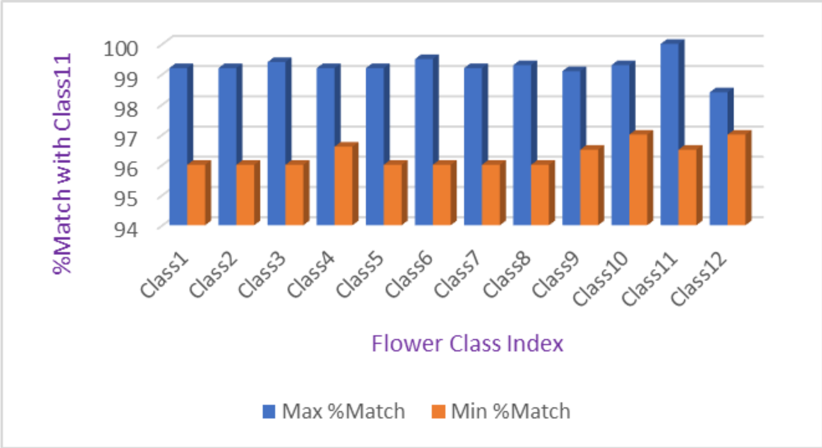


Fig. 4.12(k) Visualizing the max.% match and min.% match of training data against class11 (Gazania) using SURF feature for entire flower image

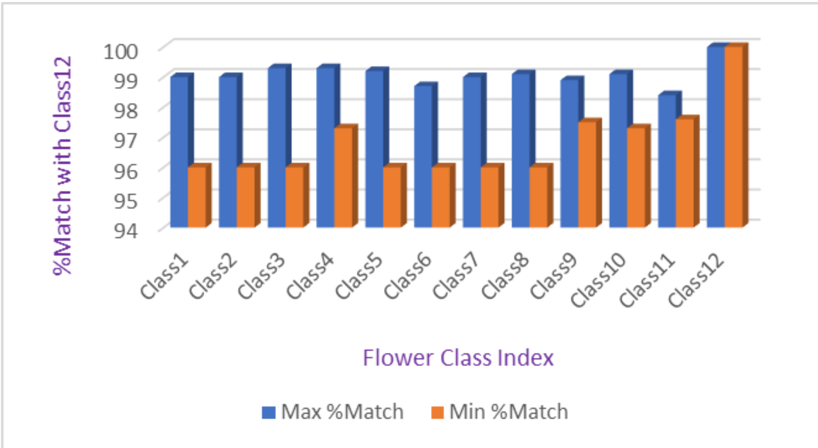


Fig. 4.12(l) Visualizing the max.% match and min.% match of training data against class12 (Calendula) using SURF feature for entire flower image

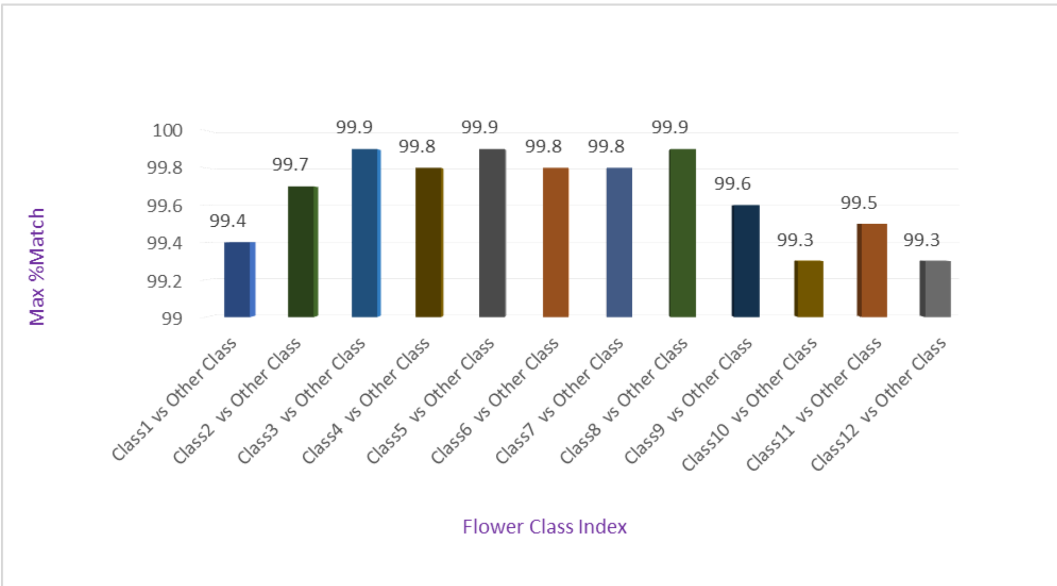


Fig. 4.13 Visualizing the interclass similarities of training data using SURF feature for entire flower image

It is observed from **Fig. 4.13** that the maximum match % among the training samples in different class is 99.9% (Class 3 Vs Other Class, Class 5 Vs Other Class and Class 8 Vs Other Class). The minimum match % among the training samples in different class is 99.3% (Class 10 Vs Other Class, Class 12 Vs Other Class and Class 8 Vs Other Class) which is much greater than the minimum match % of the samples in same class. Therefore, the threshold to classify different flowers should be greater than 99.9% to avoid any misclassification.

4.1.5. Classification

Classification is done based on different classifier such as Decision Tree, Linear Discriminant, Quadratic Discriminant, Ensemble, Support Vector Machine (SVM) and K- Nearest Neighbor (KNN). The accuracy of the prediction is measured and the result is analysed in **Table. 4.2(a)**.

Table. 4.2(a) Comparison of accuracy of different classifier for each combined feature

Combined Feature	Decision Tree	Linear Discriminant	Quadratic Discriminant	SVM	KNN	Ensemble
$F_{comb}1$	56.3	63.7	68	72.6	70.7	71
$F_{comb}2$	53.3	44.1	Failed	78.7	69	71.7
$F_{comb}3$	61	55.2	Failed	82.2	72.7	77.6
$F_{comb}4$	58.3	51.1	Failed	82	71.1	76.9
$F_{comb}5$	23	Failed	Failed	29	31.7	34

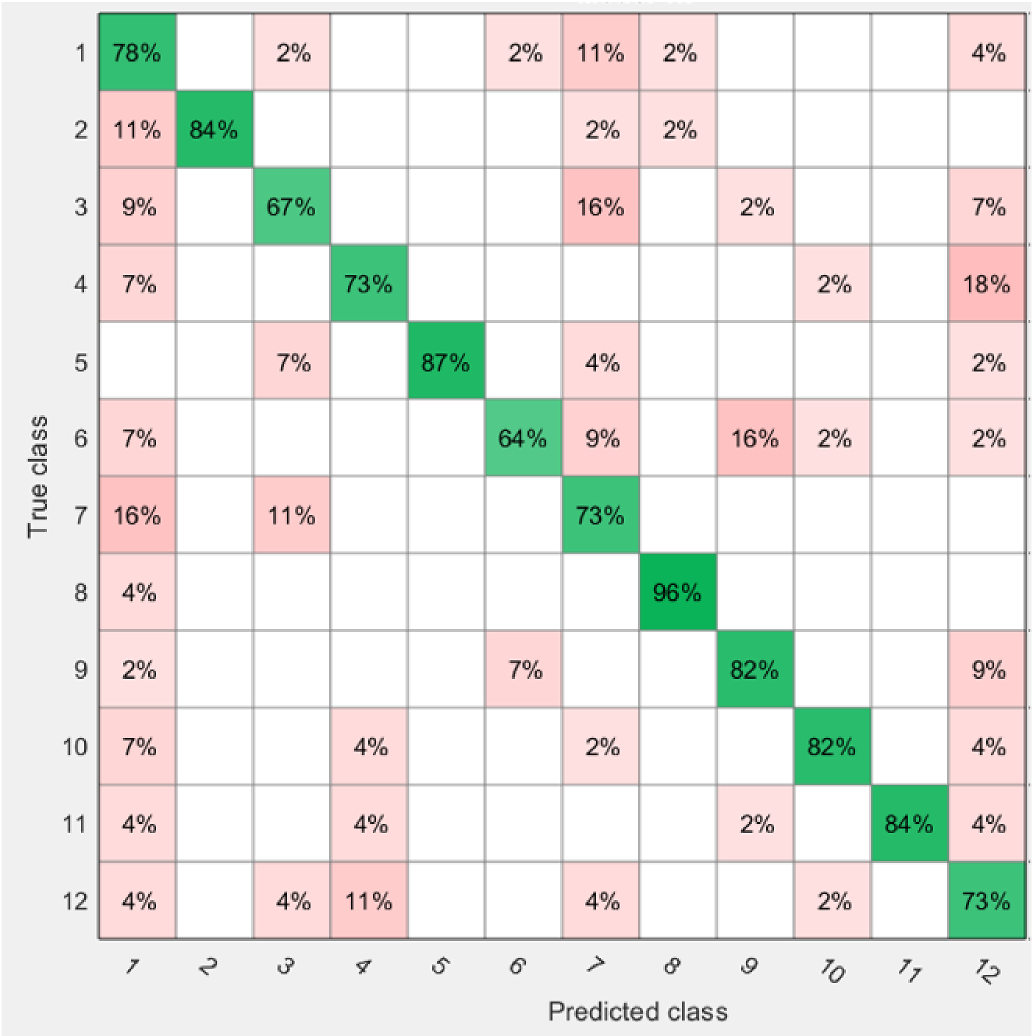
It is observed from the **Table. 4.2(a)**, SVM and KNN classifier performs better compared to other classifiers and the accuracy is much better than the other classifiers on each of the combined feature vector. It is also observed that SVM classifier performs even better than KNN. Hence the confusion matrix is generated for all models created using SVM and are presented for further analysis (**Table. 4.2(b)** to **Table. 4.2(f)**)

Table. 4.2(b) Confusion matrix of model created using F_{Comb1} for entire flower image

True class \ Predicted class	1	2	3	4	5	6	7	8	9	10	11	12
1	53%	20%		7%		4%	11%			4%		
2	11%	67%			2%	9%	4%			7%		
3		2%	60%							11%		27%
4	13%	4%		62%		4%	2%	4%	2%	4%	2%	
5		2%			73%	4%	2%	2%	13%			2%
6						80%			13%	4%		2%
7	7%	2%		4%	4%	2%	69%		11%			
8	2%			7%				89%		2%		
9							4%		96%			
10	4%		4%			2%		2%		71%	7%	9%
11										7%	91%	2%
12		2%	22%			2%				11%	2%	60%

From **Table. 4.2(b)** it is observed that classification model created using F_{Comb1} and SVM classifier, is suitable for Marigold (8), Tagar (9) and Gazania (11) But the same model is not suitable for classifying Hibiscus (1), Rose (2) , Poppy (3), Daisy (4), Morning Glory (7) and Calendula (12).

Table. 4.2(c) Confusion matrix of model created using F_{Comb2} for entire flower image



True class \ Predicted class	1	2	3	4	5	6	7	8	9	10	11	12
1	78%		2%			2%	11%	2%				4%
2	11%	84%					2%	2%				
3	9%		67%				16%		2%			7%
4	7%			73%						2%		18%
5			7%		87%		4%					2%
6	7%					64%	9%		16%	2%		2%
7	16%		11%				73%					
8	4%							96%				
9	2%					7%			82%			9%
10	7%			4%			2%			82%		4%
11	4%			4%					2%		84%	4%
12	4%		4%	11%			4%			2%		73%

From **Table. 4.2(c)** it is observed that classification model created using F_{Comb2} and SVM classifier, is suitable for Rose (2), Calla Lily (5), Marigold (8), Tagar (9) and Gazania (11) But the same model is not suitable for classifying Poppy (3), Daisy (4) and Plumeria (6).

Table. 4.2(d) Confusion matrix of model created using F_{Comb3} for entire flower image

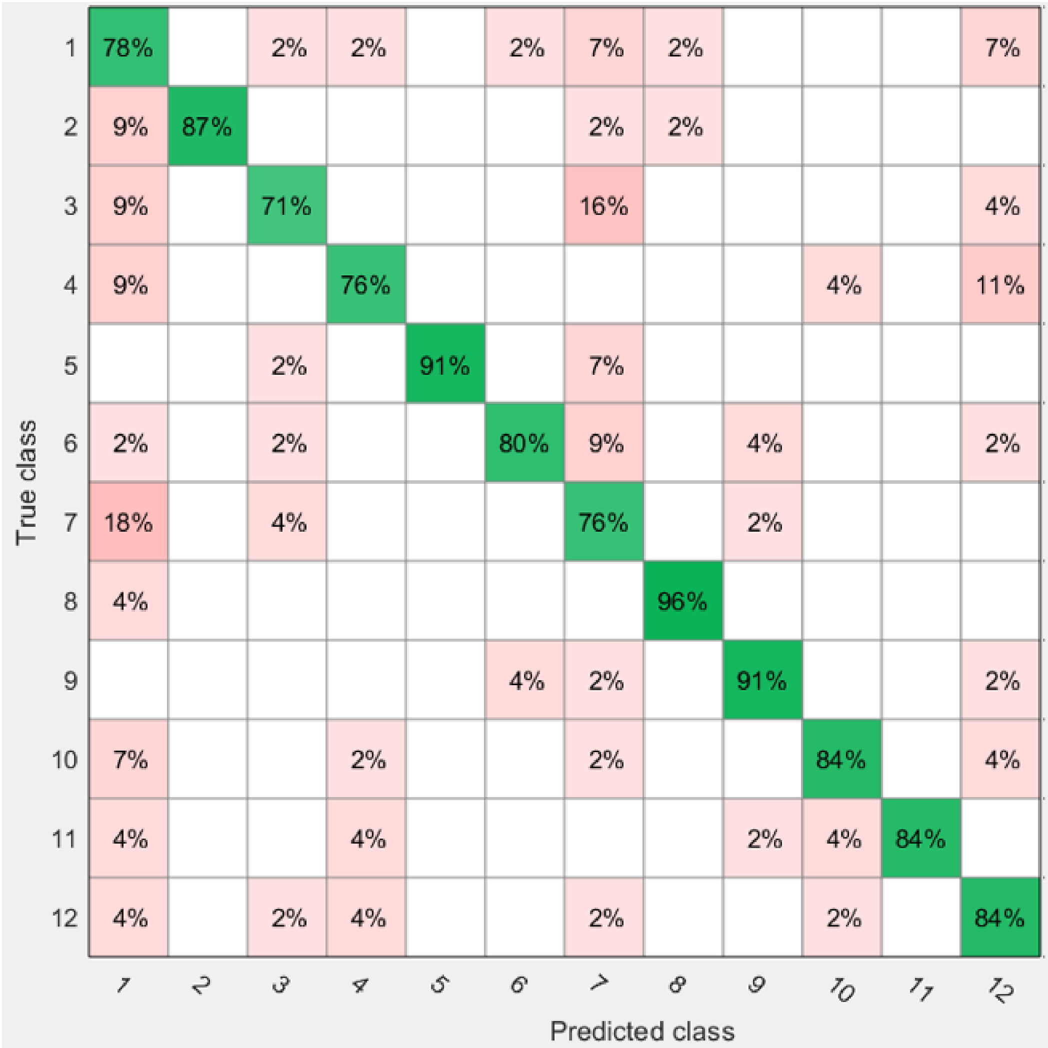


Table. 4.2(d) Confusion matrix of model created using F_{Comb3} for entire flower image

From **Table. 4.2(d)** it is observed that classification model created using F_{Comb3} and SVM classifier, is suitable for Rose (2), Calla Lily (5), Marigold (8), Tagar (9) But the same model is not suitable for classifying Hibiscus (1), Daisy (4) and Poppy (3).

Table. 4.2(e) Confusion matrix of model created using F_{Comb4} for entire flower image

True class \ Predicted class	1	2	3	4	5	6	7	8	9	10	11	12
1	82%	2%	2%			2%	4%	2%				4%
2	9%	87%					2%	2%				
3	7%		78%				11%					4%
4	7%			78%						2%		13%
5			9%		87%							4%
6	4%					76%	7%		9%			4%
7	20%		2%				78%					
8	7%							93%				
9	2%					4%	2%		87%			4%
10	7%			4%			2%			82%		4%
11	4%			4%					2%	2%	84%	2%
12	7%	2%	2%	9%			2%			2%		76%

From **Table. 4.2(e)** it is observed that classification model created using F_{Comb4} and SVM classifier, is suitable for Rose (2), Calla Lily (5), Marigold (8), Tagar (9). Sunflower (10) and for Gazania (11). But the same model is not suitable for classifying Poppy (3), Daisy (4), Plumeria (6), Morning Glory (7) and Calendula (12).

Table. 4.2(f) Confusion matrix of model created using F_{Comb5} for entire flower image

True class \ Predicted class	1	2	3	4	5	6	7	8	9	10	11	12
1	11%	9%	9%	13%	7%	4%		2%	2%	20%	16%	7%
2	7%	22%	7%	2%	4%	7%	4%		9%	11%	16%	11%
3	9%		22%	13%	4%		2%	13%	2%	4%	2%	27%
4	9%	7%	22%	7%		4%	2%	4%	4%	13%	11%	16%
5	13%	7%	2%	2%	56%		2%		4%	2%	4%	7%
6	9%	7%		2%	11%	7%	11%		16%	20%	7%	11%
7	9%	2%		9%	9%	7%	22%	4%	4%	11%	22%	
8	11%		13%	7%	4%			49%	2%	2%		11%
9	2%		2%	2%		4%	11%	2%	47%	9%	20%	
10	7%	11%	9%			11%	2%		16%	31%	13%	
11	9%	9%		2%	4%	4%	7%		11%	22%	31%	
12	2%	7%	22%	7%	2%		2%	4%	2%	4%	2%	44%

From **Table. 4.2(f)** it is observed that classification model created using F_{Comb5} and SVM classifier is not suitable for classifying any of the flower (Hibiscus (1), Rose (2), Poppy (3), Daisy (4), Calla Lily (5), Plumeria (6), Morning Glory (7), Marigold (8), Tagar (9), Sunflower (10), Gazania (11), Calendula (12)) selected for the present work.

For further analysis, other parameters like Precision, Recall and False Discovery Rate (FDR) are calculated against each model created using SVM classifier. The result is displayed in **Table. 4.3**.

Table. 4.3 Comparison of Precision and Recall value of different classifier for each combined feature vectors

Combined Feature	Accuracy	Precision (TP / (TP + FP))	Recall (TP / (TP+FN))	TP	FN	FDR	FP
$F_{comb}1$	72.6	0.73	0.85	72.60	13.09	0.27	26.85
$F_{comb}2$	78.7	0.82	0.79	78.70	21.30	0.18	17.28
$F_{comb}3$	82.1	0.85	0.82	82.10	17.90	0.15	14.49
$F_{comb}4$	82	0.85	0.82	82	18.00	0.15	14.47
$F_{comb}5$	29	0.29	0.29	29	71.00	0.71	71.00

From **Table. 4.3** and the above five confusion matrixes it is clear that the classification model designed using $F_{comb}3$ and $F_{comb}4$ using SVM as classifier performs better than other combined features *($F_{comb}1$, $F_{comb}2$ and $F_{comb}5$).*

Another round of test is also conducted to observe the classification performance of SURF feature on the same flower image database and the result is shown in **Table. 4.4**.

Table. 4.4 Accuracy of classification model created using SURF feature

Hibiscus (1)	Rose (2)	Poppy (3)	Daisy (4)	Calla Lily (5)	Plumeria (6)	Morning Glory (7)	Merigold (8)	Tagar (9)	Sunflower (10)	Gazania (11)	Calendula (12)	Overall Accuracy
11	22	100	44	78	78	100	100	78	78	89	89	72.2

From the **Table. 4.4** it is observed that SURF feature is suitable for Poppy (3), Morning Glory (7), Marigold (8) and Sunflower (10). But it is not suitable for recognizing Hibiscus (1), Rose (2) and Daisy (4). For further analysis confusion matrix is generated and the same is shown in **Fig. 4.19**

Table. 4.4(a) Confusion matrix of model created using SURF feature for entire flower image

True Class

11	11	11	11	33	0	0	0	0	11	11	0
22	22	11	0	22	0	11	0	0	0	11	0
0	0	100	0	0	0	0	0	0	0	0	0
0	0	22	44	11	0	0	0	0	0	11	11
0	0	11	0	78	11	0	0	0	0	0	0
0	11	11	0	0	78	0	0	0	0	0	0
0	0	0	0	0	0	100	0	0	0	0	0
0	0	0	0	0	0	0	100	0	0	0	0
11	0	0	0	0	0	11	0	78	0	0	0
11	0	11	0	0	0	0	0	0	78	0	0
0	11	0	0	0	0	0	0	0	0	89	0
11	0	0	0	0	0	0	0	0	0	0	89

Predicted Class

From the Confusion matrix (**Table. 4.4(a)**) it is observed that there are huge confusion between all of the classes and a proper threshold need to be determined to avoid those confusion. After analysis the threshold is claclulared and the threshold is set to 99.9% match. Another round of testing is carried out with the threshold value and only 18% of the total testing flower image is identified by SURF. Rest of the flower images are not recognized to any class as shown in **Table. 4.5**. But in this case there was no confusion as similarities less than 99.9% are not considered for any classification.

Table. 4.5 Accuracy of classification model created using SURF feature after threshold determined

Hibiscus (1)	**Rose (2)**	**Poppy (3)**	**Daisy (4)**	**Calla Lily (5)**	**Plumeria (6)**	**Morning Glory (7)**	**Merigold (8)**	**Tagar (9)**	**Sunflower (10)**	**Gazania (11)**	**Calendula (12)**	**Overall**
0.00	0.00	0.93	0.00	0.93	0.93	1.85	8.33	1.85	1.85	1.85	0	**18.52**

The above analysis shows tha a single classification model is not suitable for large number of flower species. Hence a hybrid classification model is required which will extract different features from the input image and from the extracted feature different classification model will predict a class for the image. Another decision engine will be used to determine the highly predicted class based on the maxim number of prediction done by different classification model.

4.2. Experimentations of flower recognition considering the centre part

Based on the survey conducted from 40 kids from 6 to 12 years age group it is observed that mostly human brain identifies different flowers from the shape of the flower and the centre part of the flowers. They identified some unique features of the centre part of each flower. Their observation on the 12 different flowers selected for the proposed model as depicted in **Table 4.6**

Table. 4.6. Comparison of unique features in the centre part of flowers

Sl.no.	Flower Class	Picture	Unique Flower
1	Hibiscus (1)		Centre is dark and a red stick with yellow dots on it is protruding out of the centre
2	Rose (2)		The petals form a spiral at the centre of the flower
3	Poppy (3)		Centre has a flower like protrusion
4	Daisy (4)		Centre looks like a pom-pom
5	Calla Lily (5)		Centre has a yellow stick protruding out.
6	Plumeria (6)		Centre is yellow that smudges with the petals into white

7	Morning Glory (7)		Centre is white. There is a small stick protruding out
8	Marigold (8)		Centre is scrunched up like the rest of the flower
9	Tagar (9)		Centre is yellow with one hole in the middle
10	Sunflower (10)		Centre is yellow with one hole in the middle
11	Gazania (11)		Centre is brown and darker also brown and yellow stripe is coming out from the central part
12	Calendula (12)		Central part looks like brown disc surrounded by spoon-shaped petals.

Based on the observation another experiment is conducted focussing on the centre part of the flower image. Detailed description of each step performed during this experiment is presented in this section.

4.2.1. Image Acquisition

Dataset used for the second experiment as mentioned in section 4.1.1 is used to train and test the model for the third experiment.

4.2.2. Pre-processing Images

All of the training and testing images are resized same dimension as mentioned in section 4.1.2. But the segmentation is not done because the centre part of the flower image is not having any background as the petals of the flowers cover the background.

4.2.3. Partitioning

Partitioning is done in order to capture the centre part of the flower image. Each image is split into 9 non-overlapping rectangular cells and the centre block (5th block) is cropped for the actual image processing.

The following steps are performed to crop the centre part of the RGB images:

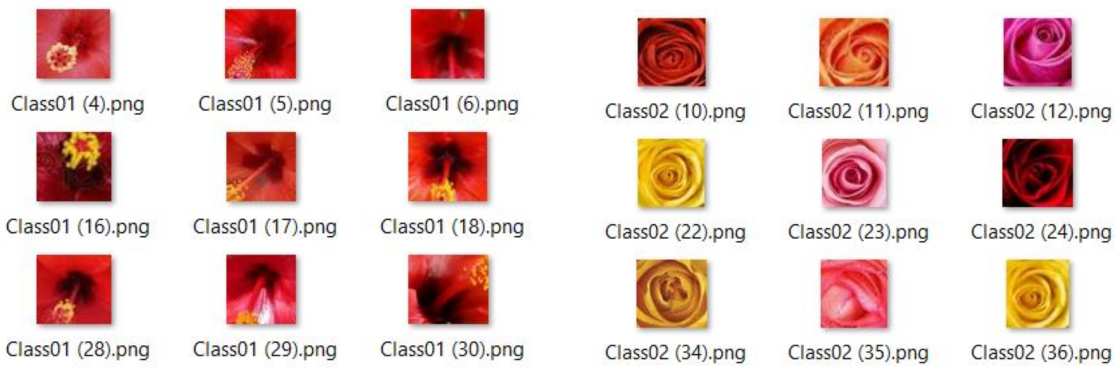
Step 1. Get the dimensions of the image from the RGB image

Step 2: Get the rows and columns to split at

Step 3. Determine the starting and ending columns of each 9 blocks

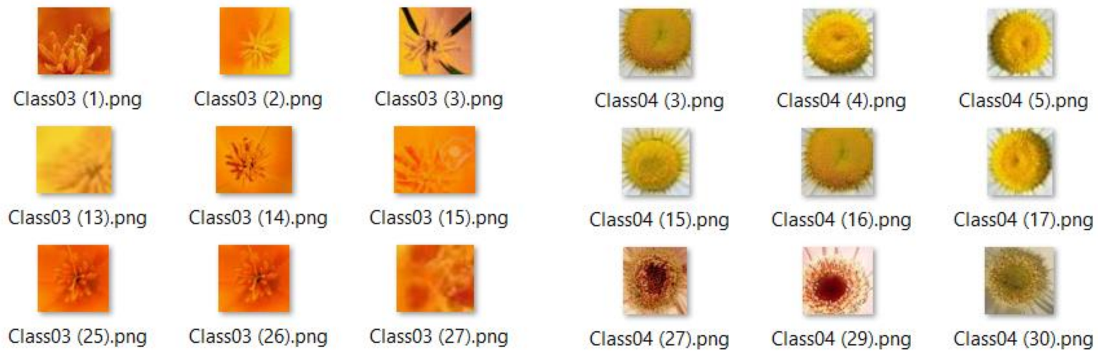
Step 4. Determine the starting and ending rows of each 9 blocks.

Step 5. Crop the centre part of the input image using the starting and ending columns along with the starting and ending rows of the 5th block as shown in sample figure **(Fig. 4.14 to Fig.4.19)**



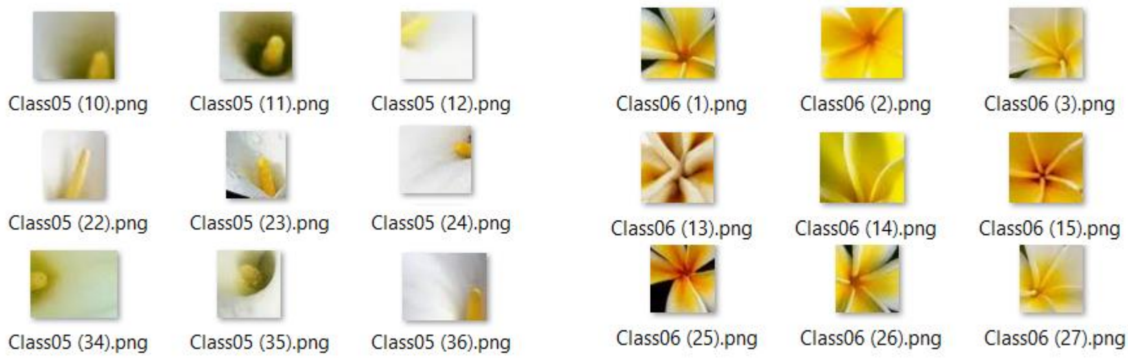
(a) (b)

Fig. 4.14(a &b) Sample images of centre part of training and test database (Class01: Hibiscus and Class02: Rose)



(a) (b)

Fig. 4.15(a &b) Sample images of centre part of training and test database (Class03: Poppy and Class04: Daisy)

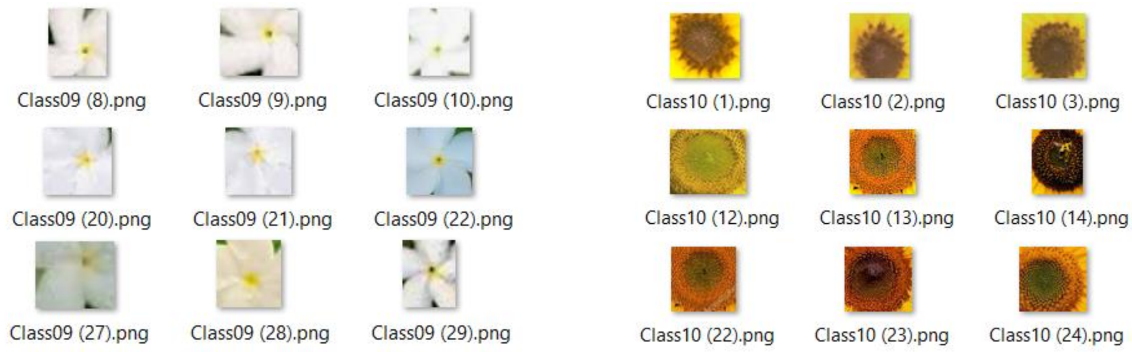


(a) (b)

Fig. 4.16(a &b) Sample images of centre part of training and test database (Class05: Calla Lily and Class06: Plumeria)

(a) (b)

Fig. 4.17 (a &b) Sample images of centre part of training and test database (Class07: Morning Glory and Class08: Marigold)



(a) (b)

Fig. 4.18 (a &b) Sample images of centre part of training and test database (Class09: Tagar and Class10: Sunflower)

(a) (b)

Fig. 4.19(a &b) Sample images of centre part of training and test database (Class11: Gazania and Class12: Calendula)

4.2.4. Feature Extraction

Same features are extracted from the centre part of each training and testing images as mentioned in section 4.1.3

4.2.5. Separating the Feature Database

The feature database is divided into two parts, in 5:1 ratio for training and testing as separated for test1. The principal components are identified for all of the combined feature vectors and the training samples are plotted in a two-dimensional space to observe the training dataset as shown in **Fig. 4.20** to **Fig. 4.25.**

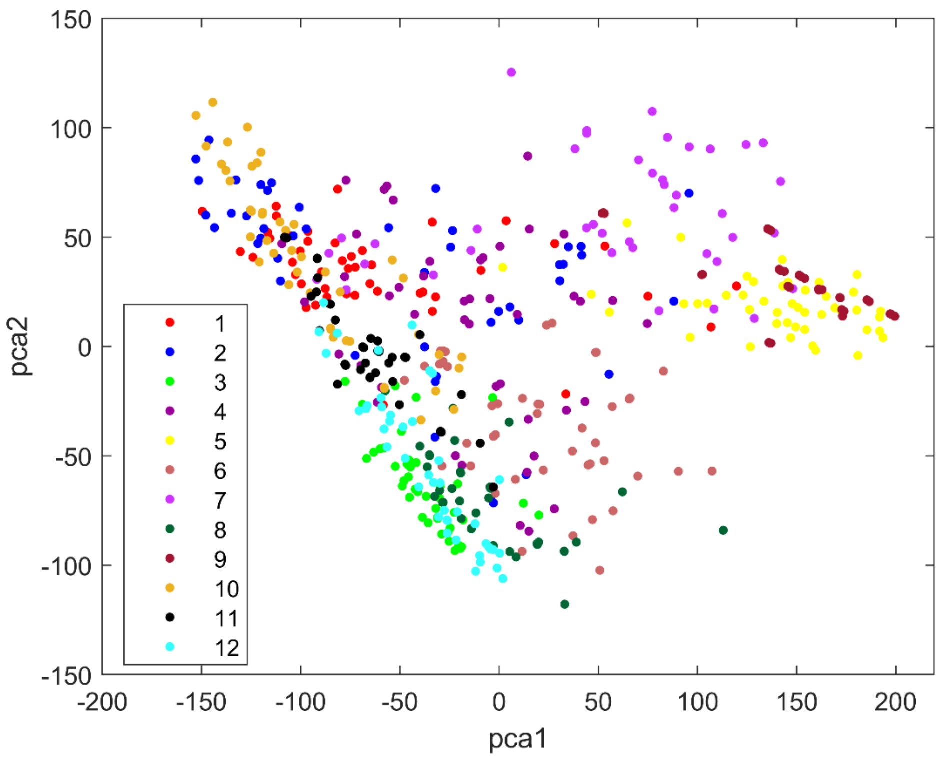


Fig. 4.20 Visualizing the training data using F_{Comb1} for centre part of flower image (1-12 represents the flower class index)

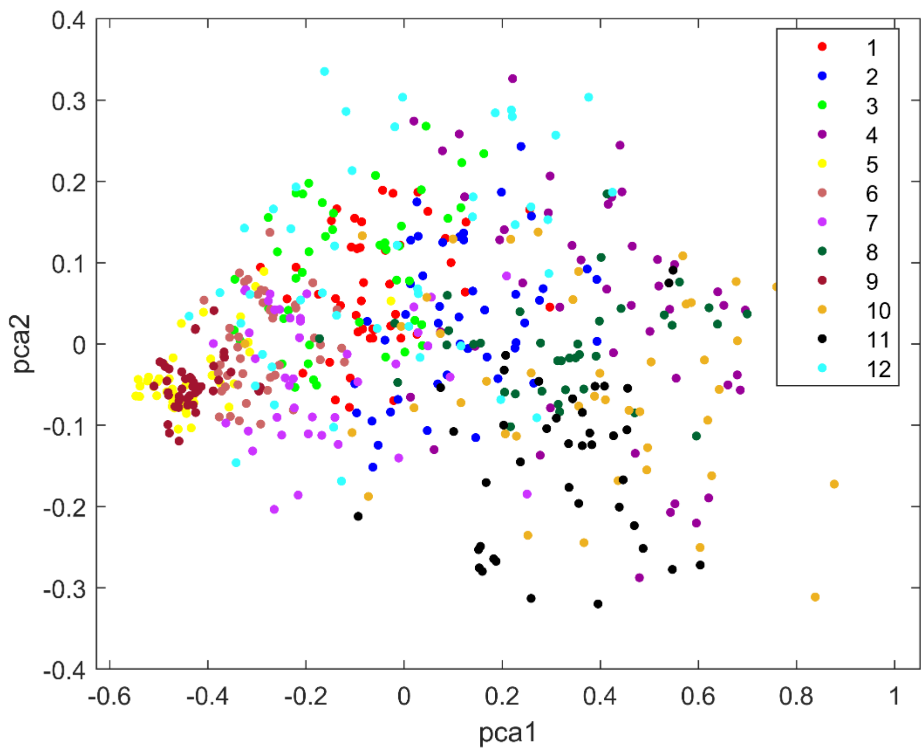


Fig. 4.21 Visualizing the training data using F_{Comb2} for centre part of flower image (1-12 represents the flower class index)

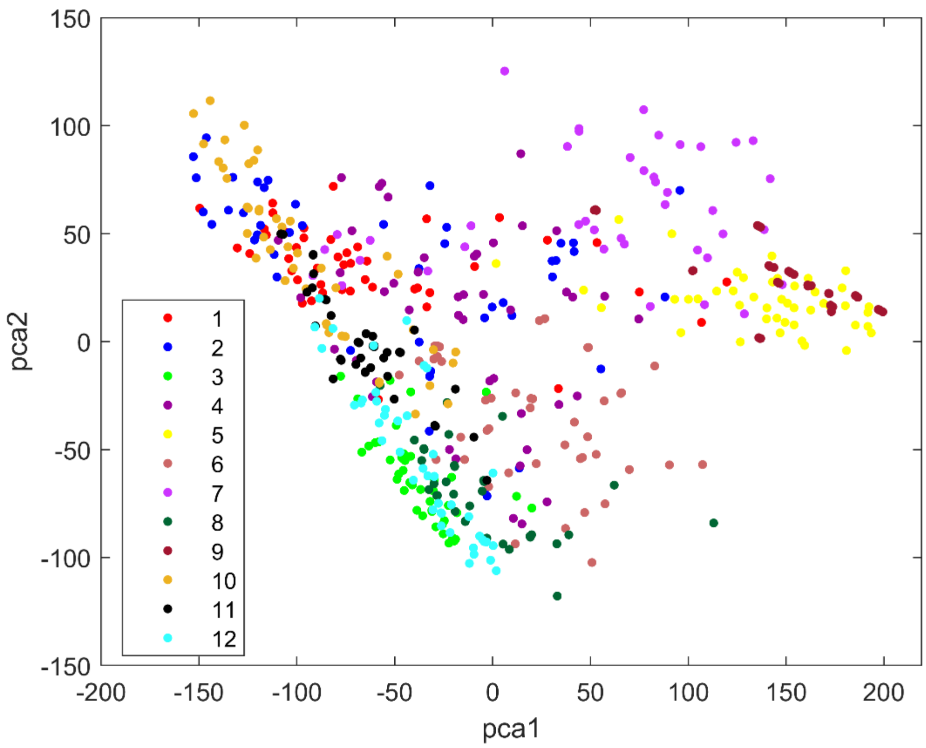


Fig. 4.22 Visualizing the training data using F_{Comb3} for centre part of flower image (1-12 represents the flower class index)

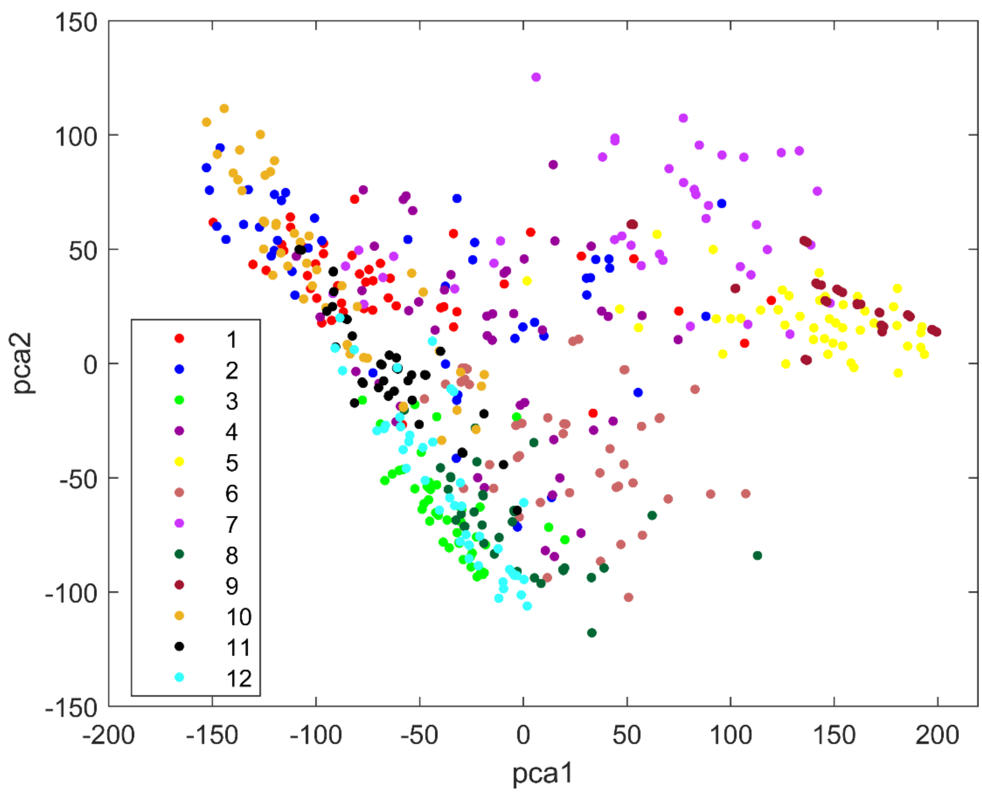


Fig. 4.23 Visualizing the training data using F_{Comb4} for centre part of flower image (1-12 represents the flower class index)

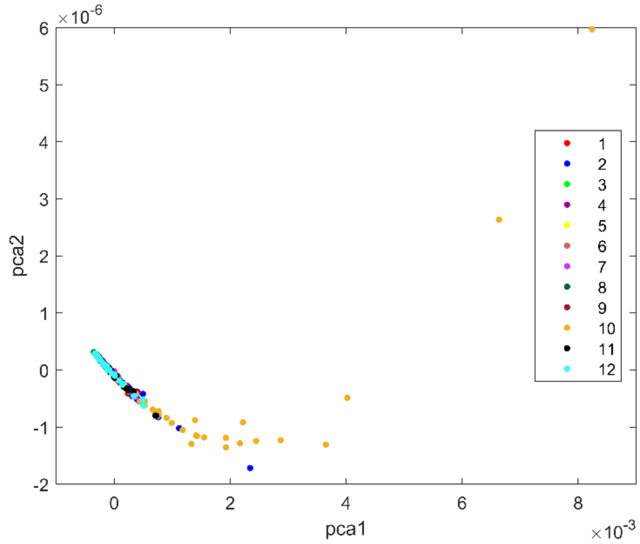


Fig. 4.24 Visualizing the training data using F_{Comb5} for centre part of flower image (1-12 represents the flower class index)

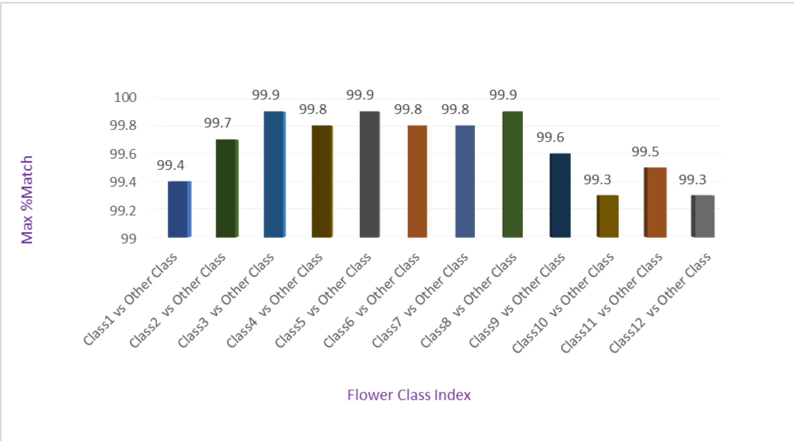


Fig. 4.25 Visualizing the training data based on SURF feature of centre part of flower

4.2.6. Classification

Classification is done based on same classifier mentioned in section 4.1.5. The accuracy of the prediction is measured and the result is analysed in **Table. 4.7**.

Table. 4.7 Comparison of accuracy of different classifier for each combined feature vectors

Combined Feature	Decision Tree	Linear Discriminant	Quadratic Discriminant	SVM	KNN	Ensemble
F_{comb}1	68.3	68	78.3	82.2	79.3	77.6
F_{comb}2	45.4	37.2	Failed	70.6	61.9	63.9
F_{comb}3	60.6	48	Failed	74.4	67	73.9
F_{comb}4	62.6	51.1	Failed	74.3	65.4	75.2
F_{comb}5	27.2	Failed	Failed	30.7	25.7	35.7

It is observed from the **Table. 4.7**, SVM classifier performs better compared to other classifiers and the accuracy is much better than the other classifiers on each of the combined feature vector. Confusion matrixes for each classification model are shown in **Table. 4.7(a)** to **Table. 4.7(e).**

Table. 4.7(a) Confusion matrix of model created using F_{Comb1} for centre part of flower image

True class \ Predicted class	1	2	3	4	5	6	7	8	9	10	11	12
1	82%	7%		7%		2%						2%
2	9%	76%	4%	7%		2%				2%		
3			80%	4%		2%						13%
4	4%	13%	2%	53%		9%		4%			7%	7%
5					78%		9%		13%			
6			2%		2%	93%		2%				
7	4%	9%			2%		84%					
8		4%		2%		2%		91%				
9									100%			
10				2%				2%	2%	78%	9%	7%
11								2%		4%	91%	2%
12	2%		11%	2%		2%				2%		80%

From **Table. 4.7(a)** it is observed that classification model created using F_{Comb4} and SVM classifier, is suitable for Poppy (3), Daisy (4), Marigold (8), Tagar (9), Sunflower (10) and for Gazania (11). But the same model does not show similar performance for classifying, Rose (2) and Calendula (12).

Table. 4.7(b) Confusion matrix of model created using F_{Comb2} for centre part of flower image

True class \ Predicted class	1	2	3	4	5	6	7	8	9	10	11	12
1	69%		7%	2%			7%	11%		2%		2%
2	2%	62%					7%	7%		20%		2%
3	13%		51%			2%	11%					22%
4	4%			71%				7%		9%	2%	7%
5			2%		73%	2%	2%		9%	2%		9%
6			11%			60%	24%		2%			2%
7	7%		2%			4%	78%	4%		2%		2%
8				4%			2%	91%		2%		
9					7%	4%	4%		84%			
10				9%				16%		67%		9%
11				4%			2%	11%			80%	2%
12	4%		7%	9%	2%		11%	4%		2%		60%

From **Table. 4.7(b)** it is observed that classification model created using F_{Comb2} and SVM classifier, is suitable for Marigold (8), Tagar (9). But the same model does not show similar performance for classifying, Hibiscus (1), Rose (2), Poppy (3), Plumeria (6), Sunflower (10) and Calendula (12).

Table. 4.7(c) Confusion matrix of model created using F_{Comb3} for centre part of flower image

True class \ Predicted class	1	2	3	4	5	6	7	8	9	10	11	12
1	82%		2%				11%			2%		2%
2	9%	73%						2%		13%		2%
3	2%		60%			4%	4%					29%
4	7%			78%				4%		9%		2%
5					64%	9%	9%		13%	2%		2%
6	4%		7%			69%	18%		2%			
7	7%			2%		13%	73%	2%		2%		
8				4%			2%	93%				
9					13%	4%			82%			
10		2%		13%			4%	2%		67%		11%
11	2%			4%			2%	9%		2%	80%	
12	7%		2%	4%		7%	2%	4%		2%		71%

From **Table. 4.7(c)** it is observed that classification model created using F_{Comb3} and SVM classifier, is suitable for Hibiscus (1), Marigold (8), Tagar (9). But the same model does not show similar performance for classifying, Rose (2), Poppy (3), Calla Lily (5), Plumeria (6), Morning Glory (7) Sunflower (10) and Calendula (12).

Table. 4.7(d) Confusion matrix of model created using F_{Comb4} for centre part of flower image

True class \ Predicted class	1	2	3	4	5	6	7	8	9	10	11	12
1	76%		4%	2%			16%					2%
2	4%	71%					7%	2%		11%	2%	2%
3	7%		67%			2%	7%					18%
4	7%			71%				9%		4%	4%	4%
5					71%	7%	2%		16%			4%
6	4%		2%			73%	20%					
7	9%					16%	73%	2%				
8				2%			2%	93%		2%		
9					13%	2%			84%			
10	2%			13%				2%		73%		9%
11				2%			2%	9%			82%	4%
12	7%		4%	11%	2%	2%	9%	4%		2%	2%	56%

From **Table. 4.7(d)** it is observed that classification model created using F_{Comb4} and SVM classifier, is suitable for Marigold (8), Tagar (9) and Gazania (11). But the same model does not show similar performance for classifying, Hibiscus (1), Rose (2), Poppy (3), Calla Lily (5), Plumeria (6), Morning Glory (7) Sunflower (10) and Calendula (12).

Table. 4.7(e) Confusion matrix of model created using F_{Comb5} for centre part of flower image

True class \ Predicted class	1	2	3	4	5	6	7	8	9	10	11	12
1	16%	7%	2%	16%	7%	2%	9%	4%		22%	7%	9%
2	4%	13%	11%	11%	16%	2%	4%		2%	24%	11%	
3		7%	33%	4%	20%	2%		24%	4%	2%	2%	
4	18%		4%	38%	2%	2%	7%	2%		16%	7%	4%
5	7%	2%	13%	4%	31%	9%	7%	18%	4%		4%	
6	7%	2%	13%	4%	13%	11%	9%	7%	11%	13%	9%	
7	9%	13%	2%	18%	2%	4%	20%	2%	2%	9%	16%	2%
8	7%	2%	20%	4%	9%		11%	42%			2%	2%
9			13%	2%	13%	7%	2%	13%	36%	4%	9%	
10				4%		2%	7%			82%	2%	2%
11	4%	11%		11%	2%	2%	13%			11%	44%	
12	13%	2%	13%	20%	11%		4%	20%		7%	7%	2%

From **Table. 4.7(e)** it is observed that classification model created using F_{Comb5} and SVM classifier, is not suitable for classifying most of the flower (Hibiscus (1), Rose (2), Poppy (3), Daisy (4), Calla Lily (5), Plumeria (6), Morning Glory (7), Marigold (8), Tagar (9), Gazania (11), Calendula (12)) selected for the present work but it is able to recognize Sunflower (10) when we consider centre part of the flower image. Whereas when we consider full part of the flower image, it showed very poor performance for all of the flower species identified for the present work.

From the confusion matrix other parameters such as Precision, Recall and False Discovery Rate (FDR) are calculated against SVM classifier. The result is displayed in **Table. 4.8**. Also, accuracy of each class is calculated for all 6 models and the same is represented in **Table. 4.9.** and **Fig. 4.26** is graphical representation of **Table. 4.10.**

Table. 4.8 Comparison of Precision and Recall of different classifier for each combined feature vectors

Model	Overall Accuracy	Precision (TP / (TP + FP))	Recall (TP / (TP + FN))	TP	FN	FDR	FP
1	82.2	0.82	0.86	82.2	13.09	0.18	18.04
2	70.6	0.74	0.71	70.6	29.4	0.26	24.81
3	74.4	0.76	0.74	74.4	25.6	0.24	23.49
4	74.3	0.76	0.74	74.3	25.7	0.24	23.46
5	30.7	0.29	0.31	30.7	69.3	0.71	75.16

Table. 4.9 Accuracy of different model created using SVM

Classification Model	Hibiscus (1)	Rose (2)	Poppy (3)	Daisy (4)	Calla Lily (5)	Plumeria (6)	Morning Glory (7)	Merigold (8)	Tagar (9)	Sunflower (10)	Gazania (11)	Calendula (12)	Overall Accuracy
Model_1	82	76	80	53	78	93	84	91	100	78	91	80	82.2
Model_2	69	62	51	71	73	60	71	98	84	67	80	60	70.6
Model_3	82	73	60	78	64	69	73	93	82	67	80	71	74.4
Model_4	76	71	67	71	71	73	73	93	84	73	82	56	74.3
Model_5	16	13	33	38	31	11	20	42	36	82	44	2	30.7

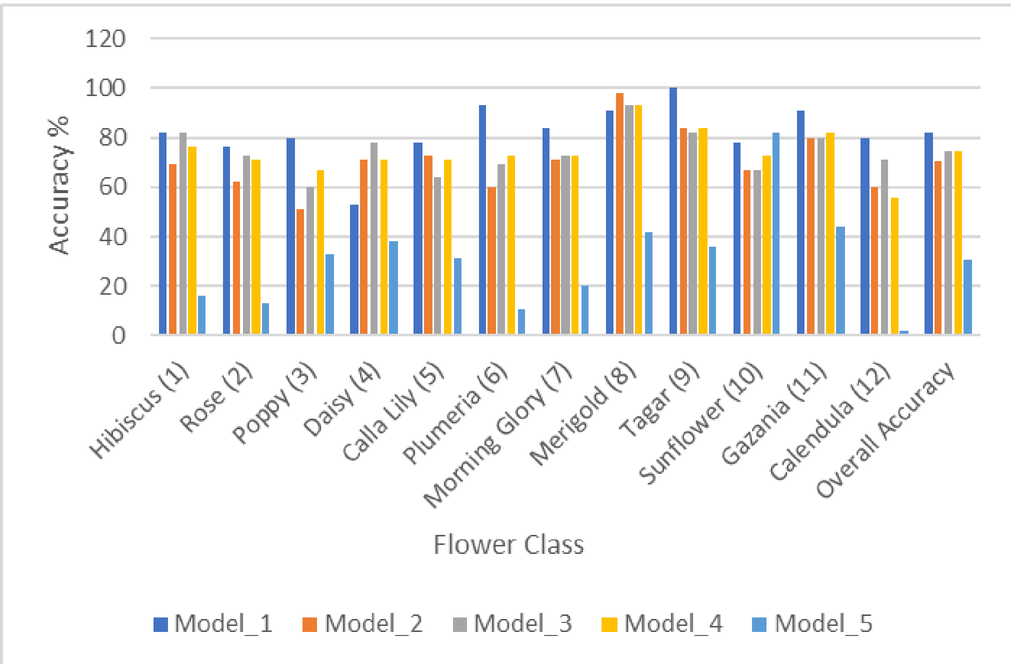


Fig. 4.26 Accuracy of different models created using SVM for centre part of flower image

During this experiment also from the **Table. 4.9** and **Fig. 4.37** it is observed that a single classification model is not suitable to classify many flower species and a hybrid model with combination of all models need to be designed so that the overall accuracy across flower species can be improved and a more reliable system can be designed. From the above two experiments it is observed that even though SVM classifier shows better performance accuracy compared to other classifiers the performance of SVM classifier is also varying based on the combined feature vectors. Hence another round of testing is conducted with different classification models focussing on centre part of the flower image as with minimum feature centre part of flower showed better result compared to full flower image. It is also observed that when classification is done based on centre part of the flower most of the classification models predicted correct class whereas when the full flower image is considered for identification, there are confusion in the predicted class suggested by different classification models as shown in **Table. 4.10** and **Table. 4.11**

Table. 4.10 Identification of Morning Glory using full image

Classification Model	Predicted Class
Classification Model1	Rose (2)
Classification Model2	Calendula (12)
Classification Model3	Hibiscus (1)
Classification Model4	Hibiscus (1)
Classification Model5	Merigold (8)
Classification Model6	Morning Glory (7)

Table. 4.11 Identification of Gazania using full image

Classification Model	Predicted Class
Classification Model1	Gazania (11)
Classification Model2	Gazania (11)
Classification Model3	Calendula (12)
Classification Model4	Calendula (12)
Classification Model5	Calendula (12)
Classification Model6	Merigold (8)

4.3. Experimentations of hybrid model considering the centre part of flower image

Based on the observation of experiment1 and experiment2, another experiment is conducted focussing on the centre part of the flower image where 11 different classification model is considered for predicting the class of the new object. The final prediction is considered for the class which has been predicted by most of the classifier. Detailed description of each step performed during this experiment is presented in this section.

4.3.1. Image Acquisition

Same dataset is used to train and test the model for experiment1 and experiment2 as mentioned in section 4.1.1

4.3.2. Pre-processing Images

All of the training and testing images are resized same dimension as mentioned in section 4.1.2.

Segmentation is not done because the centre part of the flower image is not having any background as the petals of the flowers cover the background.

4.3.3. Partitioning

Partitioning is done exactly same manner as it was done in the second experiment and described in section 4.2.3

4.3.4. Feature Extraction

Same features are extracted from the centre part of each training and testing images as mentioned in section 4.1.3

4.3.5. Separating the Feature Database

The feature database is divided into two parts, in 5:1 ratio for training and testing as separated for experiment 2 and described in section 4.2.5.

4.3.6. Classification

Classification is done based on SVM and KNN and 11 different classification models are created as described in **Table. 4.12.** Prediction from each classification model is considered for the final prediction and the highly predicted class is determined by the decision engine. The accuracy of the model is calculated and analysed. To remove the bias from the system 5-fold cross validation is performed and the overall accuracy is obtained as **89.9%**. The accuracy of the prediction is measured and the result is analysed in **Table. 4.14** and **Fig. 4.37**.

Table. 4.12 Different classification models created using SVM and KNN

Classification Model	Number of Predictors	Feature	Classifier
Model1	10	$F_{comb}1$	SVM
Model2	512	$F_{comb}2$	SVM
Model3	522	$F_{comb}3$	SVM
Model4	518	$F_{comb}4$	SVM
Model5	7	$F_{comb}5$	KNN
Model6	10	$F_{comb}1$	KNN
Model7	512	$F_{comb}2$	KNN
Model8	522	$F_{comb}3$	KNN
Model9	518	$F_{comb}4$	KNN
Model0	7	$F_{comb}5$	KNN
Model11	10 strongest points	$F_{comb}6$	% Match

Whenever any flower image is fed to the flower recognition system, different features are extracted from that image and the same is sent to the different classification model as shown in **Table. 4.12**. and a score is generated against each response class as shown in **Fig. 4.27** to **Fig. 4.36**. All of the scores are compared and the model determines the predicted class based on the highest score as shown in **Table. 4.13**.

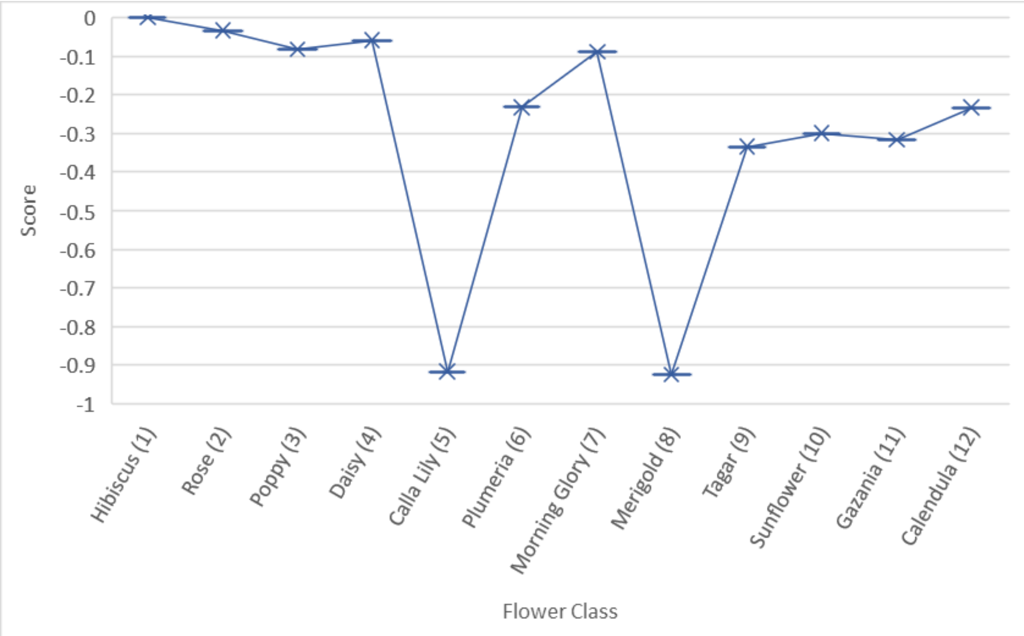


Fig. 4.27 Visualizing the score against each response class for the testing sample of Class1(Hibiscus) using Model1

From **Fig. 4.27**, it is observed that the calculated score is maximum for Hibiscus (1). Hence Model1 has classified the query image as Hibiscus (1).

Fig. 4.28 Visualizing the score against each response class for the testing sample of Class1(Hibiscus) using Model2

From **Fig. 4.28**, it is observed that the calculated score is maximum for Hibiscus (1). Hence Model2 has classified the query image as Hibiscus (1).

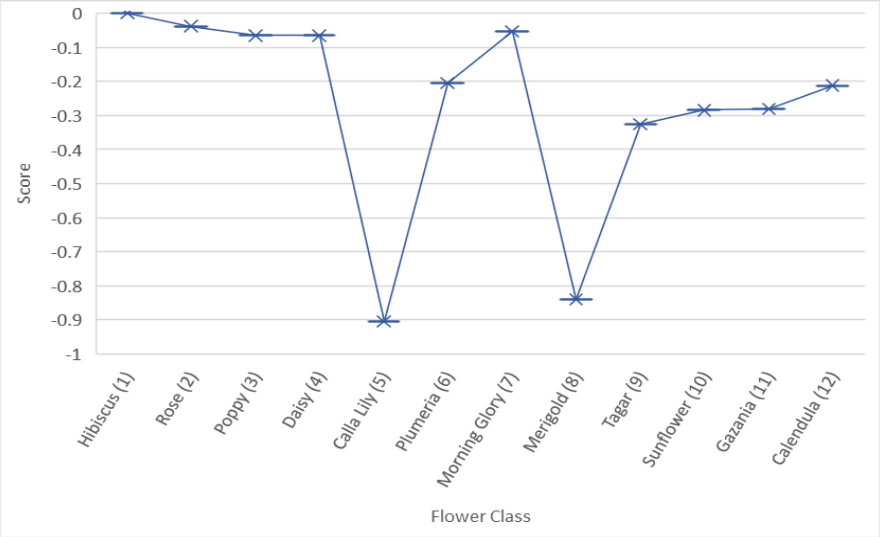


Fig. 4.29 Visualizing the score against each response class for the testing sample of Class1(Hibiscus) using Model3

From **Fig. 4.29**, it is observed that the calculated score is maximum for Hibiscus (1). Hence Model3 has classified the query image as Hibiscus (1).

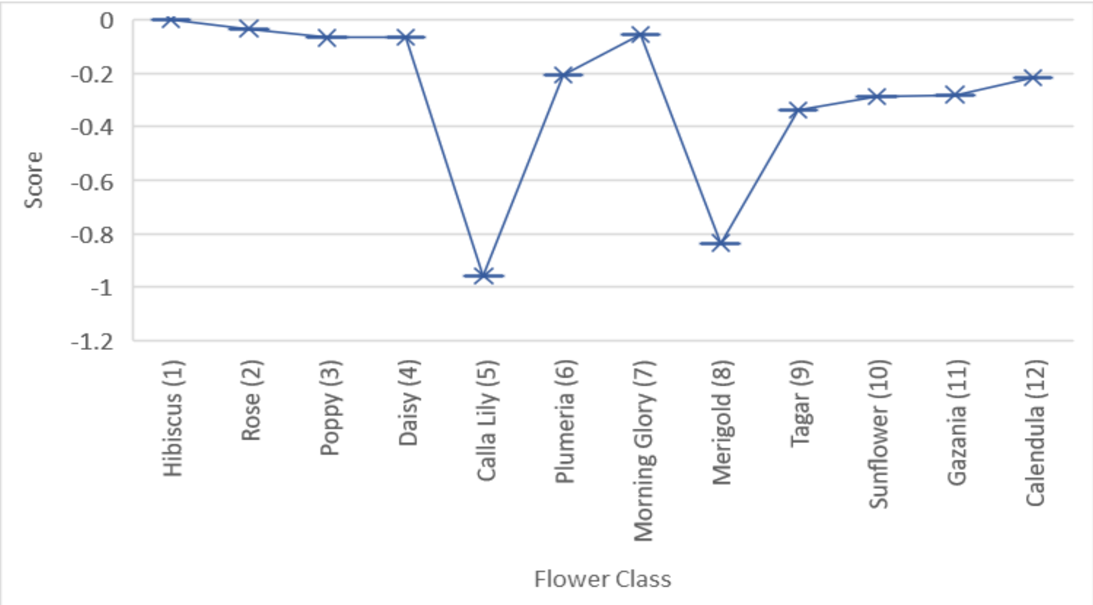


Fig. 4.30 Visualizing the score against each response class for the testing sample of Class1(Hibiscus) using Model4

From **Fig. 4.30**, it is observed that the calculated score is maximum for Hibiscus (1). Hence Model4 has classified the query image as Hibiscus (1).

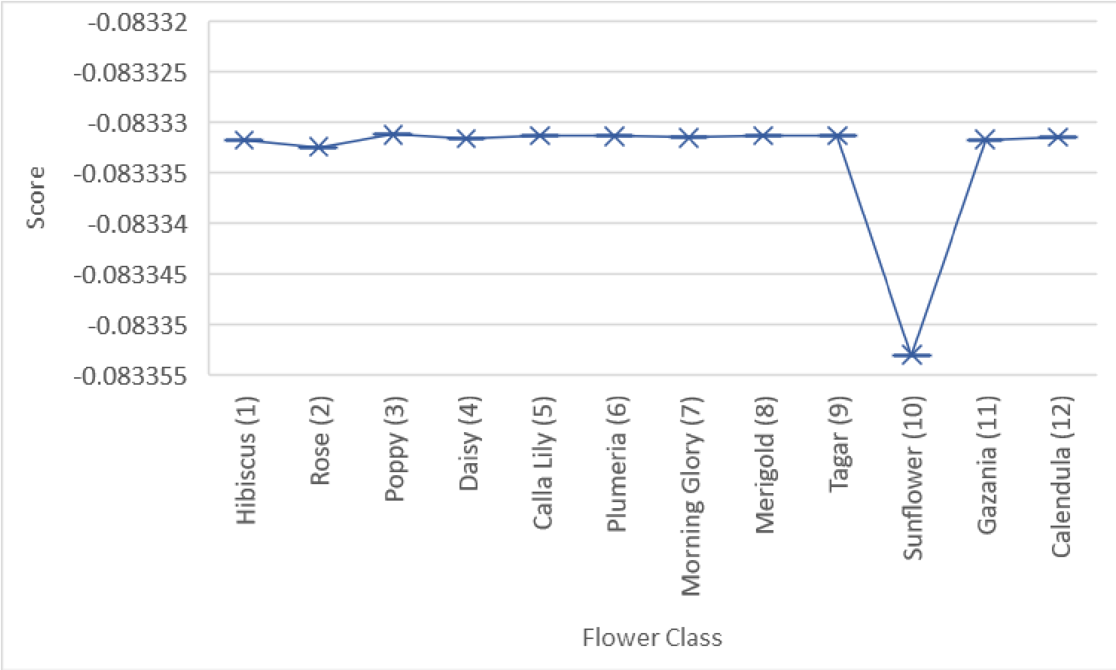


Fig. 4.31 Visualizing the score against each response class for the testing sample of Class1(Hibiscus) using Model5

From **Fig. 4.31**, it is observed that the calculated score is maximum for Poppy (3). Hence Model5 has classified the query image as Poppy (3).

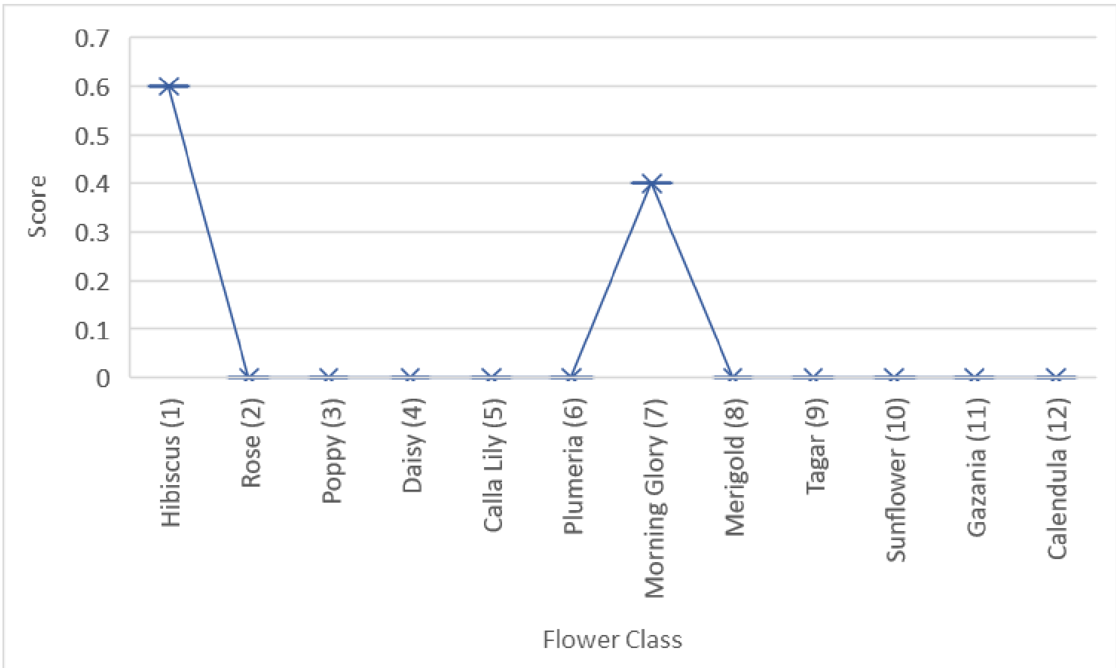


Fig. 4.32 Visualizing the score against each response class for the testing sample of Class1(Hibiscus) using Model6

From **Fig. 4.32**, it is observed that the calculated score is maximum for Hibiscus (1). Hence Model6 has classified the query image as Hibiscus (1).

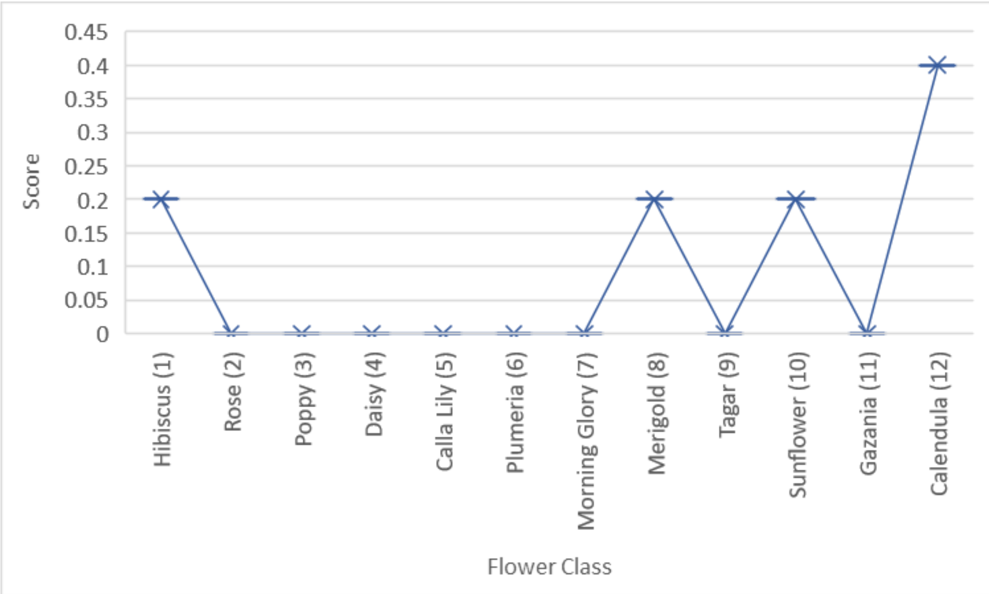


Fig. 4.33 Visualizing the score against each response class for the testing sample of Class1(Hibiscus) using Model7

From **Fig. 4.33**, it is observed that the calculated score is maximum for Calendula (12). Hence Model7 has classified the query image as Calendula (12).

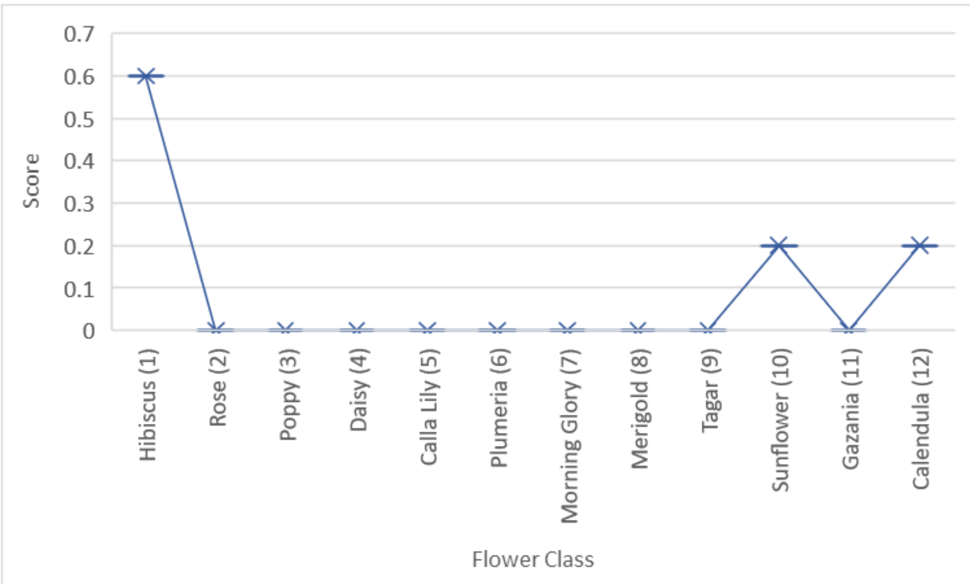


Fig. 4.34 Visualizing the score against each response class for the testing sample of Class1(Hibiscus) using Model8

From **Fig. 4.34**, it is observed that the calculated score is maximum for Hibiscus (1). Hence Model8 has classified the query image as Hibiscus (1).

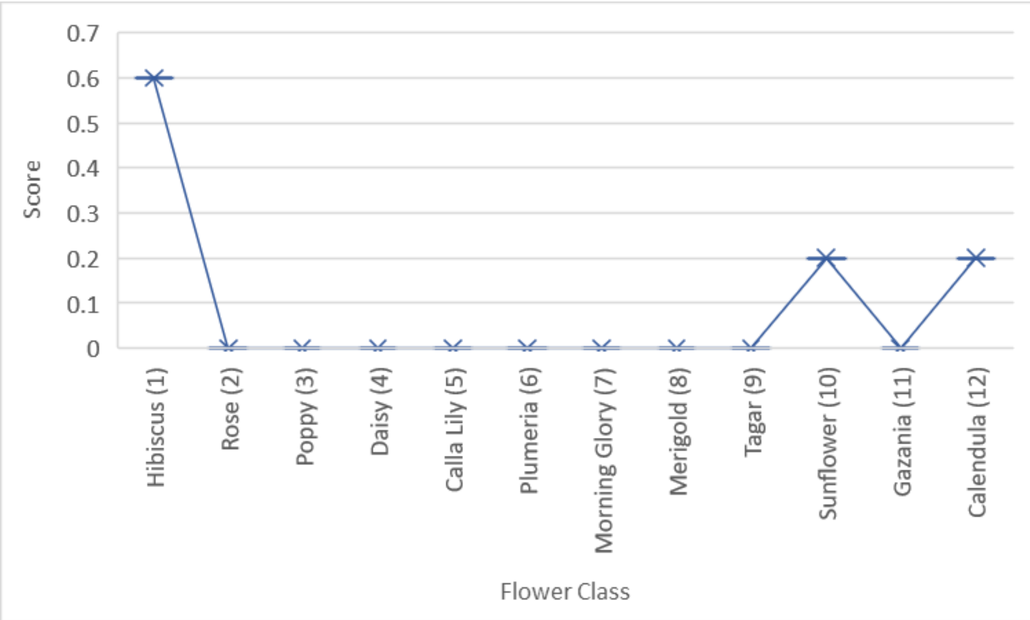


Fig. 4.35 Visualizing the score against each response class for the testing sample of Class1(Hibiscus) using Model9

From **Fig. 4.35**, it is observed that the calculated score is maximum for Hibiscus (1). Hence Model9 has classified the query image as Hibiscus (1).

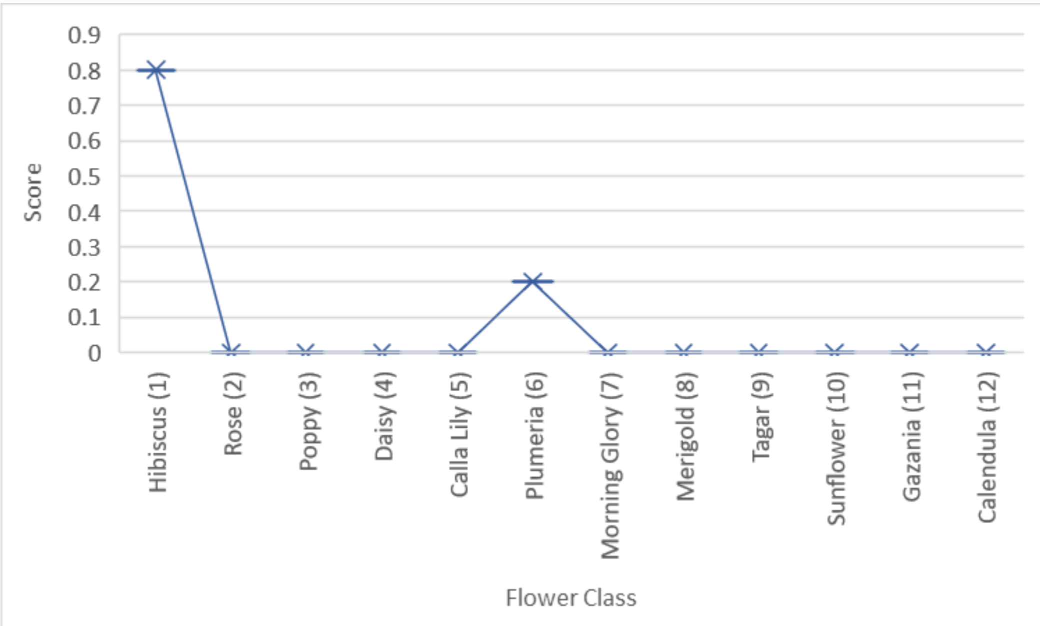


Fig. 4.36 Visualizing the score against each response class for the testing sample of Class1(Hibiscus) using Model10

From **Fig. 4.36**, it is observed that the calculated score is maximum for Hibiscus (1). Hence Model10 has classified the query image as Hibiscus (1).

Table. 4.13 Predicted class suggested by different classification models

Classification Model	Predicted Class
Classification Model1	Hibiscus (1)
Classification Model2	Hibiscus (1)
Classification Model3	Hibiscus (1)
Classification Model4	Hibiscus (1)
Classification Model5	Poppy (3)
Classification Model6	Hibiscus (1)
Classification Model7	Calendula (12)
Classification Model8	Hibiscus (1)
Classification Model9	Hibiscus (1)
Classification Model10	Hibiscus (1)

As shown in **Table. 4.13**, most of the classification models predicted the input flower as Hibiscus (1). Hence the proposed system determined the highly predicted class of the query image as Hibiscus (1). Similarly, for each flower species highly predicted class are determined and the accuracy score of the proposed hybrid model is measured and the same is presented in **Table. 4.14**.

Table. 4.14 Accuracy of proposed model

Hibiscus (1)	Rose (2)	Poppy (3)	Daisy (4)	Calla Lily (5)	Plumeria (6)	Morning Glory (7)	Merigold (8)	Tagar (9)	Sunflower (10)	Gazania (11)	Calendula (12)	Overall Accuracy
100	89	78	89	89	100	89	100	100	89	78	78	**89.9**

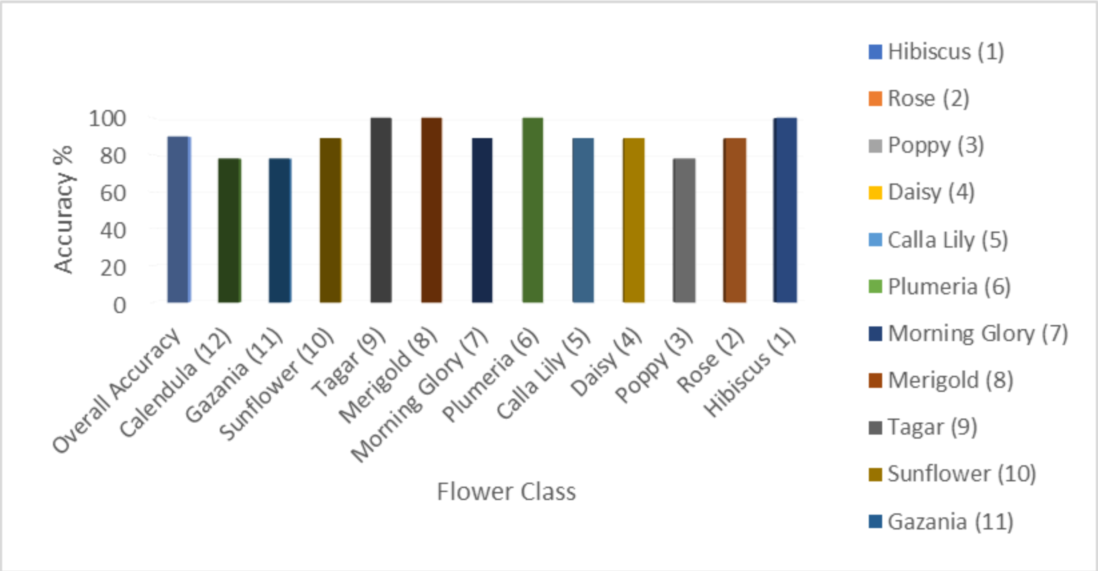


Fig. 4.37 Graphical representation of aaccuracy of proposed model

4.3.7. Validation Result against Query images

The proposed hybrid model is validated with different images of each flower class from the testing dataset and the results are analysed. Some of the results are presented in **Fig. 4.38** to **Fig. 4.50**

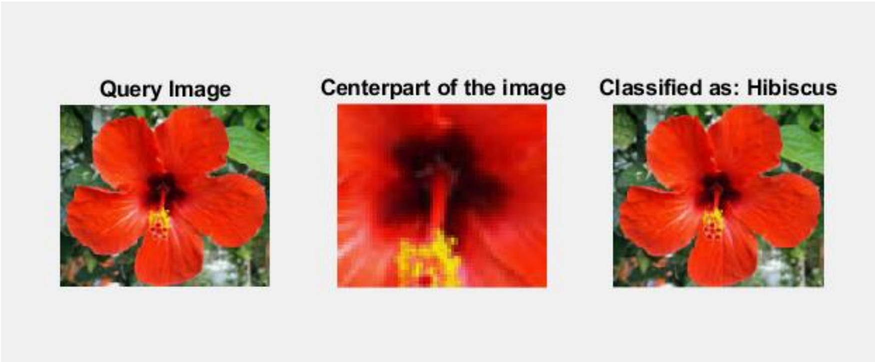


Fig. 4.38 Identification of Hibiscus (Class01)

Fig. 4.39 Identification of Rose (Class02)

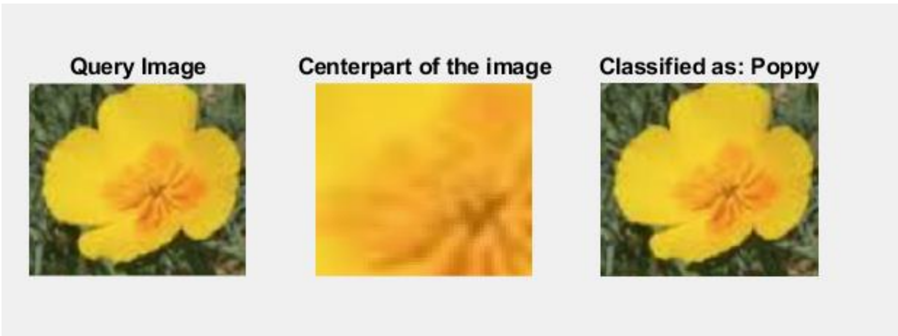


Fig. 4.40 Identification of Poppy (Class03)

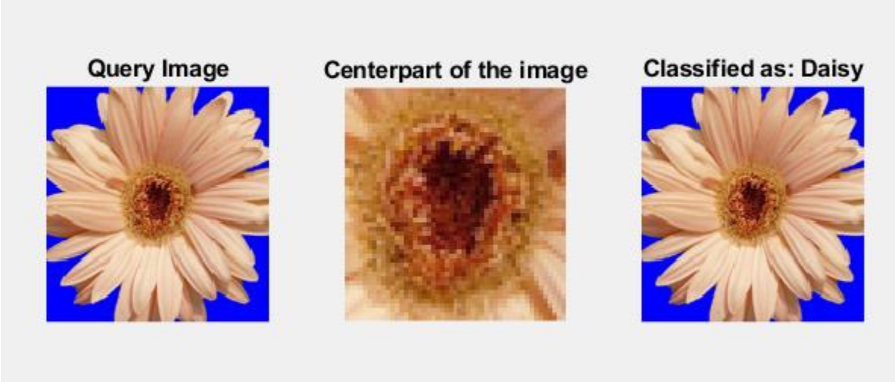


Fig. 4.41 Identification of Daisy (Class04)

Fig. 4.42 Identification of Calla Lily (Class05)

Fig. 4.43 Identification of Plumeria (Class06)

Fig. 4.44 Identification of Morning Glory (Class07)

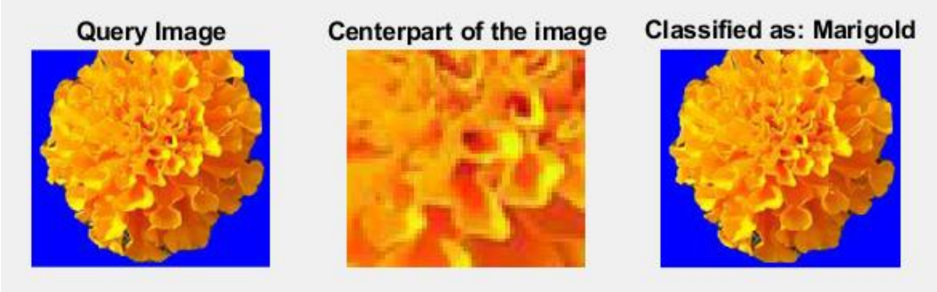


Fig. 4.45 Identification of Marigold (Class08)

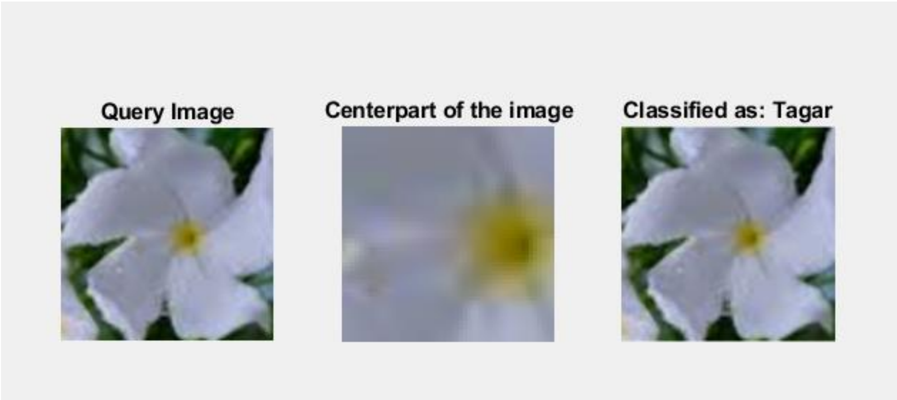


Fig. 4.46 Identification of Tagar (Class09)

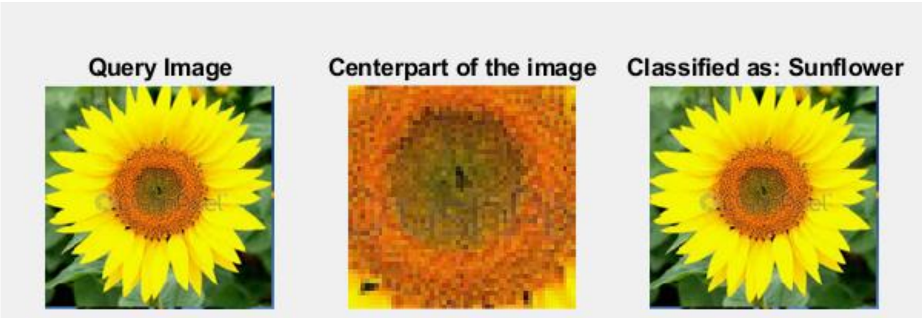


Fig. 4.47 Identification of Sunflower (Class10)

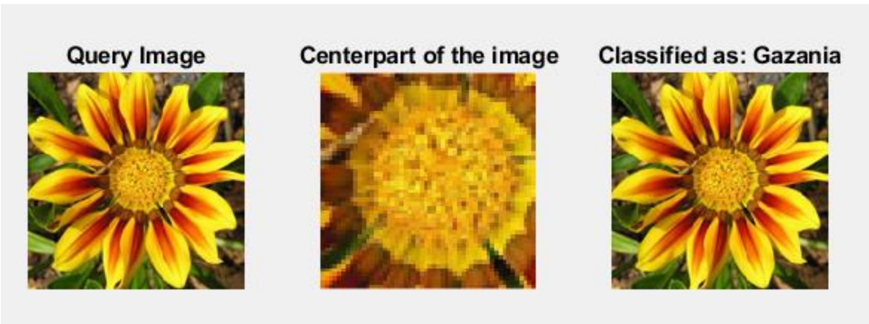


Fig. 4.48 Identification of Sunflower (Class11)

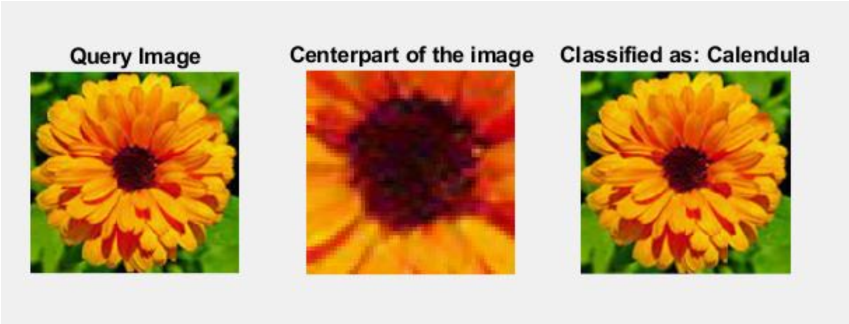


Fig. 4.49 Identification of Sunflower (Class12)

Fig. 4.50 Identification of Unknown Flower

The experimental results of the above mentioned three experiments are compared. From the analysis it is observed that when features are extracted only from centre part of the image, all of the 6 different prediction models showed better performance to recognize flower and most of the models correctly predicted the class of the query image. Hence the there was less confusion to determine the highly predicted class of the query image. But whenever the features are extracted from the entire flower image, many of the classification model could not predict the correct class of the query image. Therefore, there was a confusion to determine the highly predicted class and some cases the system could not predict the correct class of the query image. Some of the results are shown in **Table. 4.15(a)**, **Table. 4.15(b)**, **Table. 4.16(a)** and **Table. 4.16(b)**. Comparative analysis of the predicted class from both considering full image and centre part of image are shown in **Fig. 4.52** and **Fig. 4.55**.

Table. 4.15(a) Identification of Morning Glory using centre part of image

Classification Model	Predicted Class
Classification Model1	Morning Glory (7)
Classification Model2	Morning Glory (7)
Classification Model3	Morning Glory (7)
Classification Model4	Morning Glory (7)
Classification Model5	Poppy (3)
Classification Model6	Morning Glory (7)

Fig. 4.51 Identification of Morning Glory using centre part in hybrid model

When the features are extracted only from the centre oart of the flower image, the proposed system is able to recognize the Morning glory flower correctly as shown in **Fig. 4.51** .

Table. 4.15(b) Identification of Morning Glory using entire flower image

Classification Model	Predicted Class
Classification Model1	Rose (2)
Classification Model2	Calendula (12)
Classification Model3	Hibiscus (1)
Classification Model4	Hibiscus (1)
Classification Model5	Merigold (8)
Classification Model6	Morning Glory (7)

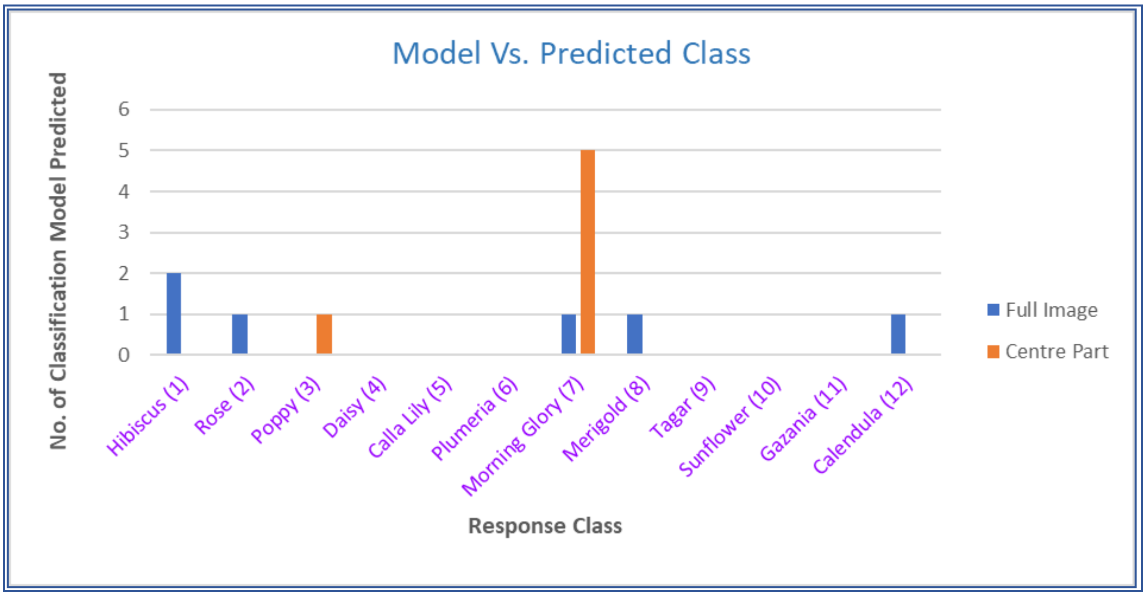


Fig. 4.52 Classification Model Vs Predicted Class for Morning Glory

Table. 4.16(a) Identification of Gazania using entire flower image

Classification Model	Predicted Class
Classification Model1	Calendula (12)
Classification Model2	Calendula (12)
Classification Model3	Gazania (11)
Classification Model4	Calendula (12)
Classification Model5	Merigold (8)
Classification Model6	Gazania (11)

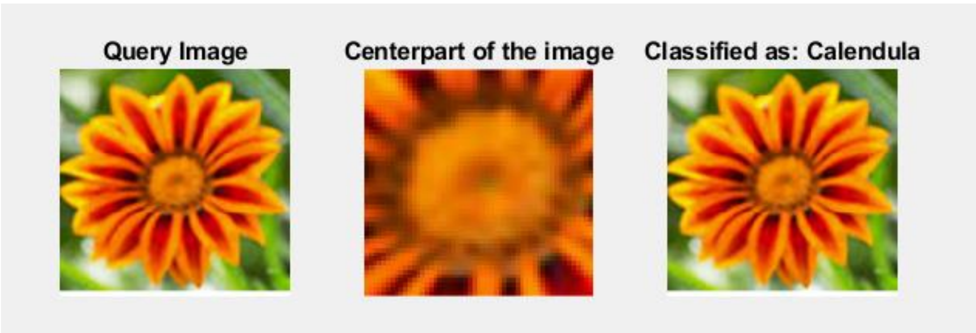


Fig. 4.53 Identification of Gazania using full image

When considered full image, Gazania is classified as Calendula and there is huge confusion in different prediction model and most of the prediction models classified the flower as Calendula as shown in **Fig. 4.53**.

Table. 4.16(b) Identification of Gazania using centre part of image

Classification Model	Predicted Class
Classification Model1	Gazania (11)
Classification Model2	Gazania (11)
Classification Model3	Morning Glory (7)
Classification Model4	Gazania (11)
Classification Model5	Gazania (11)
Classification Model6	Poppy (3)

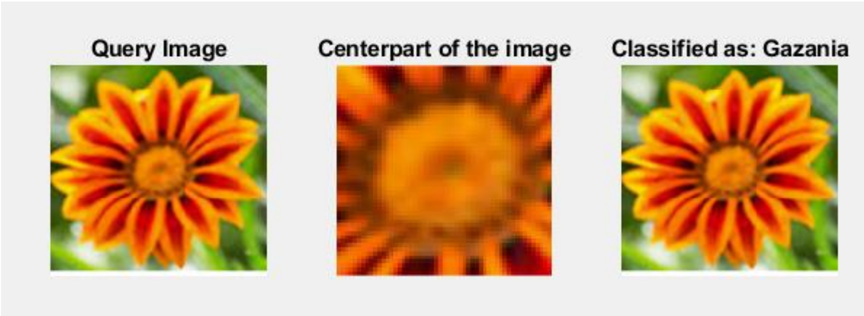


Fig. 4.54 Identification of Gazania using centre part in hybrid model

When considered only centre part of the image, all of the 6 different prediction models correctly classified the flower as Gazania and the hybrid model is able to determine the highly predicted class correctly and the system is able to recognize the flower correctly as shown in **Fig. 4.54**.

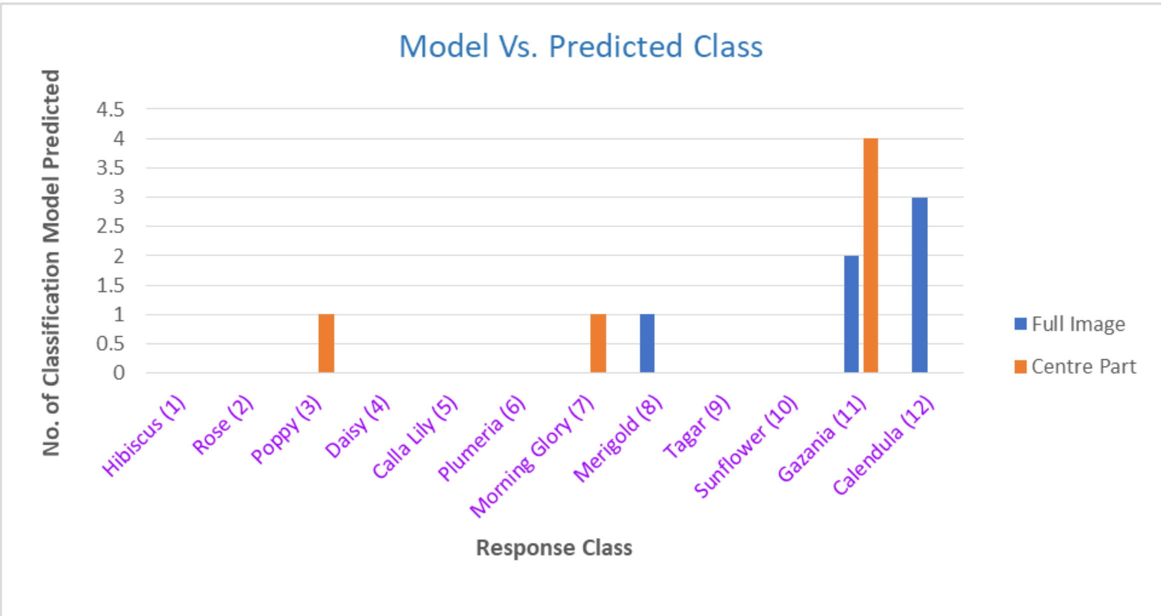


Fig. 4.55 Classification Model Vs Predicted Class for Gazania

Chapter 5

COMPARATIVE ANALYSIS

In order to test the validity of the hybrid model proposed in this work, relevant verification experiments are carried out. Results of each experiment are validated against 12 different species of flowers. In this section results from different tests conducted are analysed and compared against other models suggested by **Lodh and Parekh [11]**, **Dhar (2019) [17]**, **Tiay *et al.* (2014) [9],** and **Shaparia *et al.* (2017) [12].**

5.1. Comparison with Lodh and Parekh [11]

The performance of the proposed model is compared against the performance of the model suggested by **Lodh and Parekh [11]** and the result showed that the proposed model is more accurate to recognize flower. The comparison of the results are shown in **Table. 5.1**, **Table. 5.2** and **Table. 5.3.**

Table. 5.1 Comparison of Accuracy with **Lodh and Parekh [11]**

Classification Model	Hibiscus (1)	Rose (2)	Poppy (3)	Daisy (4)	Calla Lily (5)	Plumeria (6)	Morning Glory (7)	Merigold (8)	Tagar (9)	Sunflower (10)	Gazania (11)	Calendula (12)	Overall Accuracy
Proposed Model	100	89	78	89	89	100	89	100	100	89	78	78	**89.9**
Lodh and Parekh [11]	76	71	67	71	71	73	73	93	84	73	82	56	74.3

From **Table. 5.1** it is observed that the model suggested by **Lodh and Parekh [11]** shows 74.3% accuracy whereas the proposed model shows 89.9% accuracy. The confusion matrix of both models are shown in **Table. 5.2** and **Table. 5.3.**

Table. 5.2 Confusion matrix of proposed system

True Class

100	0	0	0	0	0	0	0	0	0	0	0
11	89	0	0	0	0	0	0	0	0	0	0
0	0	78	0	0	22	0	0	0	0	0	0
11	0	0	89	0	0	11	0	0	11	11	0
0	0	0	0	89	0	0	0	11	0	0	0
0	0	0	0	0	100	0	0	0	0	0	0
0	0	0	0	11	0	89	0	11	0	0	0
0	0	0	0	0	0	0	100	0	0	0	0
0	0	0	0	0	0	0	0	100	0	0	0
0	0	0	11	0	0	0	0	0	89	0	0
0	0	0	11	0	0	11	0	0	0	78	0
0	0	11	0	0	0	0	11	0	0	0	78

Predicted Class

Table. 5.3 Confusion matrix of system suggested by **Lodh and Parekh [11]**

True Class

89	0	0	0	0	0	11	0	0	0	0	0
11	78	0	0	0	11	0	0	0	0	0	0
0	0	56	11	0	0	0	0	0	0	0	33
0	0	11	56	0	0	11	0	0	11	11	0
0	0	0	0	78	11	0	0	11	0	0	0
0	0	0	0	0	89	0	0	0	0	0	11
0	0	0	0	11	0	78	0	11	0	0	0
0	0	0	0	0	0	0	100	0	0	0	0
0	0	0	0	22	0	11	0	67	0	0	0
0	0	0	22	0	0	0	0	0	67	11	0
11	0	0	0	0	0	0	0	0	11	67	11
0	0	33	0	0	0	0	11	0	0	0	56

Predicted Class

From the confusion matrix shown in **Table. 5.2** and **Table. 5.3**, it is observed that the proposed model is able to resolve the misclassification issues in Class1, Class6, and Class9. It is also observed that the Proposed model shows better performance accuracy of classification for all other class for which the model is trained. The model proposed by **Lodh and Parekh [11]**, requires better segmentation for proper classification and if proper segmentation is not done, the system shows erroneous result. But the proposed model removes the dependency of segmentation as it focusses on the Centre part of the image rather than the entire image thus eliminating the requirement of segmentation.

5.2. Comparison with Shaparia *et al.* (2017) [12]

The performance of the proposed model is compared against the performance of the model suggested by **Shaparia *et al.* (2017) [12]** and the result showed that the proposed model is more accurate to recognize flower. The comparison of the results are shown in **Table. 5.4**, **Table. 5.5** and **Table. 5.6**.

Table. 5.4 Comparison of Accuracy with **Shaparia *et al.* (2017) [12]**

Classification Model	Hibiscus (1)	Rose (2)	Poppy (3)	Daisy (4)	Calla Lily (5)	Plumeria (6)	Morning Glory (7)	Merigold (8)	Tagar (9)	Sunflower (10)	Gazania (11)	Calendula (12)	Overall Accuracy
Proposed Model	100	89	78	89	89	100	89	100	100	89	78	78	**89.9**
Shaparia et al.[12]	53	67	60	62	73	80	69	89	96	71	91	60	72.6

From **Table. 5.4** it is observed that the model suggested by **Shaparia *et al.* (2017) [12]** shows 72.6% accuracy whereas the proposed model shows 89.9% accuracy. The confusion matrix of both models are shown in **Table. 5.5** and **Table. 5.6.**

Table. 5.5 Confusion matrix of proposed system

True Class

100	0	0	0	0	0	0	0	0	0	0	0
11	89	0	0	0	0	0	0	0	0	0	0
0	0	78	0	0	22	0	0	0	0	0	0
11	0	0	89	0	0	11	0	0	11	11	0
0	0	0	0	89	0	0	0	11	0	0	0
0	0	0	0	0	100	0	0	0	0	0	0
0	0	0	0	11	0	89	0	11	0	0	0
0	0	0	0	0	0	0	100	0	0	0	0
0	0	0	0	0	0	0	0	100	0	0	0
0	0	0	11	0	0	0	0	0	89	0	0
0	0	0	11	0	0	11	0	0	0	78	0
0	0	11	0	0	0	0	11	0	0	0	78

Predicted Class

Table. 5.6 Confusion matrix of proposed system suggested by **Shaparia *et al.* (2017) [12]**

True Class

33	33	0	33	0	0	0	0	0	0	0	0
11	44	0	33	0	11	0	0	0	0	0	0
0	0	56	33	0	0	0	11	0	0	0	0
22	22	11	22	0	0	11	0	0	11	0	0
0	0	0	0	78	11	0	0	11	0	0	0
0	0	0	11	0	89	0	0	0	0	0	0
0	0	0	0	11	0	78	0	11	0	0	0
0	0	0	0	0	0	0	89	0	0	0	11
0	0	0	11	11	0	11	0	67	0	0	0
0	0	0	22	0	0	0	0	0	67	11	0
11	0	0	11	0	0	0	0	0	11	44	22
0	0	33	0	0	0	0	11	0	0	0	56

Predicted Class

From the above two confusion matrixes shown in **Table. 5.5** and **Table. 5.6**, it is observed that the proposed model is able to resolve the misclassification issues in Class1, Class6, Class8, and Class9. It is also observed that the Proposed model shows better performance accuracy of classification for all other class for which the model is trained. The proposed model also removes the dependency of segmentation as it focusses on the Centre part of the image rather than the entire image thus eliminating the requirement of segmentation.

5.3. Comparison with Tiay *et al.* (2014) [9]

In the model suggested by **Tiay *et al.* (2014) [9]**, the colour and edge properties need to be determined using Hu's seven-moment technique and histogram. Then K-nearest neighbour (KNN) classifier is used to classify the flower of unknown input image. When the model is validated against 12 different classes of flower it shows poor accuracy whenever there are similarities in petal colours. The performance accuracy is shown 43% whereas the proposed model is able to overcome the challenges and showed satisfactory result with 89.9% accuracy. The model proposed by **Tiay *et al.* (2014) [9]**, requires better segmentation for proper classification and if proper segmentation is not done, the system shows erroneous result. But the proposed model does not require any segmentation. The comparison of the results are shown in **Table. 5.7**, **Table. 5.8** and **Table. 5.9**.

Table. 5.7 Comparison of Accuracy with **Tiay *et al.* (2014) [9]**

Classification Model	Hibiscus (1)	Rose (2)	Poppy (3)	Daisy (4)	Calla Lily (5)	Plumeria (6)	Morning Glory (7)	Merigold (8)	Tagar (9)	Sunflower (10)	Gazania (11)	Calendula (12)	Overall Accuracy
Proposed Model	100	89	78	89	89	100	89	100	100	89	78	78	**89.9**
Tiay et al. [9]	27	22	44	38	27	33	36	89	82	22	69	27	43

From **Table. 5.7** it is observed that the model suggested by **Tiay *et al.* (2014) [9]** shows 43% accuracy whereas the proposed model shows 89.9% accuracy. The confusion matrix of both models are shown in **Table. 5.8** and **Table. 5.9.**

Table. 5.8 Confusion matrix of proposed system

True Class

100	0	0	0	0	0	0	0	0	0	0	0
11	89	0	0	0	0	0	0	0	0	0	0
0	0	78	0	0	22	0	0	0	0	0	0
11	0	0	89	0	0	11	0	0	11	11	0
0	0	0	0	89	0	0	0	11	0	0	0
0	0	0	0	0	100	0	0	0	0	0	0
0	0	0	0	11	0	89	0	11	0	0	0
0	0	0	0	0	0	0	100	0	0	0	0
0	0	0	0	0	0	0	0	100	0	0	0
0	0	0	11	0	0	0	0	0	89	0	0
0	0	0	11	0	0	11	0	0	0	78	0
0	0	11	0	0	0	0	11	0	0	0	78

Predicted Class

Table. 5.9 Confusion matrix of proposed system suggested by **Tiay *et al.* (2014) [9]**

True Class

27	11	2	9	2	2	9	0	16	4	4	3
7	22	9	7	0	0	4	13	20	4	9	4
2	4	44	2	0	0	7	4	2	2	9	16
11	11	2	38	0	0	4	4	11	2	2	13
4	4	4	2	27	16	2	11	16	0	0	13
2	2	7	9	9	33	4	4	9	4	2	13
7	2	13	4	4	2	36	0	11	9	2	9
0	0	2	0	0	0	0	89	4	2	2	0
4	0	0	2	0	0	0	11	82	0	0	0
9	4	9	4	2	4	7	18	2	22	7	11
2	0	7	7	0	0	0	2	2	7	69	4
18	4	13	9	2	4	2	2	4	11	2	27

Predicted Class

From the above two confusion matrixes shown in **Table. 5.8** and **Table. 5.9**, it is observed that the proposed model is able to resolve the misclassification issues in Class1, Class6, Class8, and Class9. It is also observed that the Proposed model shows better performance accuracy of classification for all other class for which the model is trained.

5.4. Comparison with Dhar (2019) [17]

Comparison with **Dhar (2019) [17]** and Proposed Method on 12 different species of flowers. Threshold has been determined by the below approach. For each sample of the training dataset, 10 strongest points are selected and percentage match is calculated against each class. Threshold has been determined by the maximum matched percentage obtained from interclass. Threshold is determined as 99.9%. Based on the threshold we could only correctly identify 25% of the testing flower images. When the SURF feature is combined with colour, texture, GIST features the hybrid model was able to correctly identify 89.9% of the testing flower images. The comparison of the results are shown in **Table. 5.10**, **Table. 5.11** and **Table. 5.12**.

Table. 5.10 Comparison of Accuracy with **Dhar (2019) [17]**

Classification Model	Hibiscus (1)	Rose (2)	Poppy (3)	Daisy (4)	Calla Lily (5)	Plumeria (6)	Morning Glory (7)	Merigold (8)	Tagar (9)	Sunflower (10)	Gazania (11)	Calendula (12)	Overall Accuracy
Proposed Model	100	89	78	89	89	100	89	100	100	89	78	78	**89.9**
Dhar [17]	0.00	0.00	0.93	0.00	0.93	0.93	1.85	8.33	1.85	1.85	1.85	0	**18.52**

From **Table. 5.10** it is observed that the model suggested by **Dhar (2019) [17]** shows 18.52% accuracy whereas the proposed model shows 89.9% accuracy. The confusion matrix of both models are shown in **Table. 5.11** and **Table. 5.12.**

Table. 5.11 Confusion matrix of proposed system

True Class

100	0	0	0	0	0	0	0	0	0	0	0
11	89	0	0	0	0	0	0	0	0	0	0
0	0	78	0	0	22	0	0	0	0	0	0
11	0	0	89	0	0	11	0	0	11	11	0
0	0	0	0	89	0	0	0	11	0	0	0
0	0	0	0	0	100	0	0	0	0	0	0
0	0	0	0	11	0	89	0	11	0	0	0
0	0	0	0	0	0	0	100	0	0	0	0
0	0	0	0	0	0	0	0	100	0	0	0
0	0	0	11	0	0	0	0	0	89	0	0
0	0	0	11	0	0	11	0	0	0	78	0
0	0	11	0	0	0	0	11	0	0	0	78

Predicted Class

Table. 5.12 Confusion matrix of proposed system suggested by **Dhar (2019) [17]**

True Class

11	11	11	11	33	0	0	0	0	11	11	0
22	22	11	0	22	0	11	0	0	0	11	0
0	0	100	0	0	0	0	0	0	0	0	0
0	0	22	44	11	0	0	0	0	0	11	11
0	0	11	0	78	11	0	0	0	0	0	0
0	11	11	0	0	78	0	0	0	0	0	0
0	0	0	0	0	0	100	0	0	0	0	0
0	0	0	0	0	0	0	100	0	0	0	0
11	0	0	0	0	0	11	0	78	0	0	0
11	0	11	0	0	0	0	0	0	78	0	0
0	11	0	0	0	0	0	0	0	0	89	0
11	0	0	0	0	0	0	0	0	0	0	89

Predicted Class

From the above two confusion matrixes shown in **Table. 5.11** and **Table. 5.12**, it is observed that the proposed model is able to resolve the misclassification issues among all of the flower class in the scope of present work. It is also observed that the Proposed model shows better performance accuracy of classification for all other class for which the model is trained.

From the above mentioned 4 comparative analysis it is clear that the proposed model shows better performance and scored better accuracy for all 4 models suggested by **Lodh and Parekh [11]**, **Dhar (2019) [17]**, **Tiay *et al.* (2014) [9],** and **Shaparia *et al.* (2017) [12].** It also overcome the challenges to identify the ROI properly using segmentation.

Chapter 6

CONCLUSIONS AND FUTURE SCOPES

6.1. Conclusion

Flower is the most attractive part of a plant and by identifying the flower we can also identify the plant. Hence a good number of research is done in the computer vision-based flower recognition system. But due to the complexity of the environment, the diversity within the flower category and the interclass similarity among different flower categories the accuracy of flower classification has always been unsatisfactory. There are lots of challenges to design robust and reliable system to classify large number of flower categories. Aiming at the challenges, this thesis work proposed an improved model for flower classification which has shown better performance compared to other models designed in past. The proposed approach has shown flower can be classified in better way, if we consider the centre part of the flowers instead of considering the entire flower image as there are unique feature in each flower species even though there are dissimilarities among the different flowers in the same flower species. Also, it is observed that hybrid model with combination of different classification models performs better compared to any specific classification model.

The contributions of the thesis work are mentioned below:

- Problem has been identified based on literature survey carried in the field of flower recognition along with classification techniques.

- An improved model has been proposed to overcome the issues and challenges faced by the other researchers for flower recognition. It has been tried to develop the model in such way that any flower specie can easily with 11 different classification models to have holistic view of accuracy score among all eleven models.

- Initially experimentation and testing has been done on full image of the flowers consisting 12 species with different combined features and sample flowers are taken at different weather conditions. Different parameters for the proposed model have been analysed to check the accuracy and reliability of the system.

- To Remove the bias from the system 5-fold cross validation has been performed to determine the overall accuracy of the proposed system

- In the proposed work, based on the filed survey, a novel technique to identify the flower species by considering only the centre part of the flower has been experimented and analysed. In this case, segmentation is not required and images are obtained by cropping from the full flower.

- It has been observed and concluded that accuracy of the proposed model is much better with minimum feature considering central part of the flower instead of full image.

- Finally, in improved hybrid models, consisting 11 classifiers have been designed to identify the unknown flower for better prediction and reliability.

- The improved flower classification model which is a combination of different classification models and considers the benefits of different approaches which showed satisfactory results and scores 89.9% accuracy on 12 classes of flower for which the model is trained.

- A comparison among the existing models (4) have been done with proposed improved model to check the accuracy using the same data sets. All the cases, it has been observed that improved model shows better accuracy.

6.2. Future Scope:

The future scope related to this study is discussed below.

- In the present work, the model is trained with only 12 different flower species. In future the model can be trained to classify large number of flowers species.

- The model can be integrated with any mobile application and can be utilized to recognize flower by capturing images using phone.

- Currently the model is designed to handle only top view of the flower. System can be extended to handle other views also to make it more robust and reliable.

- Shape and edge feature can be added to the system to make the system more reliable.

- As of now hardcoded threshold is specified to identify unknown flower images, Adaptive threshold needs to be implemented for better classification.

- More studies on different flower species will aid in this process.

References

[1] Das, M., Manmatha, R., Riseman, E.: 'Indexing flower patent images using domain knowledge', *IEEE Intell. Syst. Appl.*, 1999, **14**, (5), pp. 24– 33.

[2] Saitoh, T., Kaneko, T.: 'Automatic recognition of wild flowers', *IEEE Explore: System and Computer in Japan*, 2003, **34**, pp.10-17.

[3] Nilsback, M. Zisserman, A.: 'Automated Flower Classification over a Large Number of Classes', *Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, 2008, pp. 722-729. Bhubaneswar, India doi: 10.1109/ICVGIP.2008.47.

[4] Guru, D. S., Sharath, Y. H., Manjunath, S.: 'Texture Features and KNN in Classification of Flower Images', *Recent Trends in Image Processing and Pattern Recognition (RTIPPR)*, 2010, pp.21-29. Published By Foundation of Computer Science.

[5] Mohd-Ekhsan, H., Hamid, J.N., Ramle, R., Ismail, M. H.: 'Classification of Flower Images Based on Colour and Texture Features Using Neural Network', *IEEE International Conference on Intelligent Network and Computing* (ICINC 2010), 2010, pp. 361-364. Kuala Lumpur, Malaysia.

[6] Hsu, T., Lee, C., Chen, L.: 'An interactive flower image recognition system', *Springer, Multimedia Tools and Applications*, 2011, **53**(1) 53–73.

[7] Hong, S. W., Choi, L.: 'Automatic recognition of flowers through colour and edge based contour detection', *3rd International Conference on Image Processing Theory, Tools and Applications (IPTA)*, 2012, Istanbul, Turkey, pp. 141-146

[8] Mukane, S. M., Kendule, J. A.: 'Flower Classification Using Neural Network Based Image Processing, *IOSR Journal of Electronics and Communication Engineering* 2013, pp.80-85.

[9] Tiay, T., Benyaphaichit, P., Riyamongkol, P.: 'Flower recognition system based on image processing', *Third ICT International* (ICT-ISPC-2014), Nakhon Pathom, Thailand, 2014 pp. 99-102.

[10] Ben Mabrouk, A., Najjar A., Zagrouba, E.: 'Image Flower Recognition based on a New Method for Color Feature Extraction', *In Proceedings of the 9th International Conference on Computer Vision Theory and Applications (VISAPP-2014)*, Lisbon, Portugal, pp. 201-206 ISBN: 978-989-758-004-8 DOI: 10.5220/0004636302010206

[11] Lodh, A. Parekh, R.: 'Flower recognition system based on colour and GIST features', *IEEE Devices for Integrated Circuit (DevIC)*, 2017, Kalyani, India, pp.790-794.

[12] Shaparia , R. H., Patel N. M., Shah, Z. H.: 'Flower Classification using Texture and Colour Features', *International Conference on Research and Innovations in Science*, Engineering &Technology, 2017, Gujarat, India, **2**, pp.113–118..

[13] Arafah, M. Moghli, Q. A.: 'Efficient Image Recognition Technique Using Invariant Moments and Principle Component Analysis', *Journal of Data Analysis and Information Processing*, 2017, **5**, pp.1-10.

[14] Lee, H.H., Hong, K.S.: 'Automatic recognition of flower species in the natural environment', *Image and Vision Computing,* **61**, pp.98–114.

[15] Hiary, H. Saadeh, H., Saadeh, M., Yaqub, M.: 'Flower classification using deep convolutional neural networks', *IET Comput. Vis.,* 2018, **12 (6)**, pp. 855-862.

[16] Cibuk, M., Budak, U., Guo, Y., M. Ince, C. Sengur, A.: 'Efficient deep features selections and classification for flower species recognition', Measurement, 2019, **137**, pp. 7–13.

[17] Dhar, P.: 'A New Flower Classification System Using LBP and SURF Features', *Int. Journal of Image, Graphics and Signal Processing*, 2019, **5**, pp.13-20.

[18] Yangyang, Y., Xiang, F.: 'A Flower Image Classification Algorithm Based on Saliency Map and PCANet', *Journal of Communication and Computer*, 2019, **15**, pp. 14-24 doi: 10.17265/1548-7709/2019.01.002.

[19] Bae, K., Park, J., Lee, J., Lee, Y., Lim, C.: 'Flower classification with modified multimodal convolutional neural networks', *Expert Systems With Applications,2020,* **159**, pp. 11345.

[20] Cao, S. Song, B.: 'Visual attentional-driven deep learning method for flower recognition, *Mathematical Biosciences and Engineering,* 2021, **18**, pp.1981-1991. doi: 10.3934/mbe.2021103

[21] Bozkurt, F.: 'A Study on CNN Based Transfer Learning for Recognition of Flower Species', *European Journal of Science and Technology*, 2021, **32**, pp.883-890.

[22] Lv, R., Li, Z., Zuo, J., Liu, J.: 'Flower Classification and Recognition Based on Significance Test and Transfer Learning', *IEEE International*

Conference on Consumer Electronics and Computer Engineering (ICCECE 2021), 2021, Guangzhou, China, DOI: 10.1109/ICCECE51280.2021.934246

[23] Togˇaçar, M., Ergen, B., Cömert, Z.: Classification of flower species by using features extracted from the intersection of feature selection methods in convolutional neural network models', Elsevier: *Measurement,* 2020, **158**, pp. 107703

[24] Kaur, R., Jain A., Kumar, S.: 'Optimization classification of sunflower recognition through machine learning', Elsevier: *Materials Today: Proceedings,* **51**, pp.207–211.

[25] Zhang, M., Su, H., Wen, J.: 'Classification of flower image based on attention mechanism and multi-loss attention network', Elsevier: *Computer Communications, 2022,* **179**, pp. 307–317.

[26] Oliva, Torralba, A.: 'Modeling the shape of the scene: a holistic representation of the spatial envelope, *International Journal of Computer Vision*, 2001 **42(3)**, pp.145-175.

APPENDIX -A

A.1 Sample Code for the Implementation

A.1.1 Sample code to Get Centre of RGB Image

```
clc;clear;clearvars;format compact;
fontSize=10;
srcTrainDir = '..\CropedTrainImages\';
srcTrainFiles=dir(fullfile(srcTrainDir,'*.png'));
totalTrainFiles = length(srcTrainFiles);
trainImgs = cell(totalTrainFiles,1);
trainImgsNm = cell(totalTrainFiles,1);
trainCenterPartOfImgs = cell(totalTrainFiles,1);

%Delete all files in the destination folder
myFolder = '..\CenterOfTrainImages';
filePattern = fullfile(myFolder, '*.png'); % Change to whatever pattern you need.
theFiles = dir(filePattern);
for centImg = 1 : length(theFiles)
  baseFileName = theFiles(centImg).name;
  fullFileName = fullfile(myFolder, baseFileName);
  fprintf(1, 'Now deleting %s\n', fullFileName);
  delete(fullFileName);
end

%Step 1: Reading the traing dataset
for c=1:totalTrainFiles
    B= fullfile(srcTrainDir,srcTrainFiles(c).name);
    trainImgs{c}=imread(B);
    trainImgsNm{c}=B;
end

% pre-processing
% resize the images
for cnt=1:totalTrainFiles
    [r, c, ~] = size(trainImgs{cnt});
    aspectRatio = r/c;
    r = 150;
    c = r/aspectRatio;
    trainImgs{cnt}= imresize(trainImgs{cnt},[r,c]);  %DB
end

% Process the Images
for  j=1:totalTrainFiles
     rgbImage = (trainImgs{j});
% Get the dimensions of the image.  numberOfColorBands should be = 3.
[rows,columns,numberOfColorBands] = size(rgbImage);
% Get the rows and columns to split at taking care to handle odd-size dimensions:
col1 = 1;
col2 = floor(columns/3);
col3 = col2 + 1;
col4 = col3 + 1;
```

```
row1 = 1;
row2 = floor(rows/3);
row3 = row2+1;

% Now crop
upperLeft = imcrop(rgbImage, [col1 row1 col2 row2]);
upperMid = imcrop(rgbImage, [col2 row1 col3 row2]);
upperRight = imcrop(rgbImage, [col3 row1 columns - col3 row2]);
MidMid = imcrop(rgbImage, [col2 row2 col3 row3]);
lowerLeft = imcrop(rgbImage, [col1 row3 col2 row2]);
lowerRight = imcrop(rgbImage, [col3 row3 columns - col2 rows - row2]);


% Save the center part of the images in desired location
ImageFolder = '..\CenterOfTrainImages';
B1= fullfile(srcTrainDir,srcTrainFiles(j).name);
imgName = replace( B1 , 'CropedTrainImages' , 'CenterOfTrainImages' ) ;
fprintf(1, 'Now copying %s\n', imgName);
imwrite(MidMid,imgName) ;
trainCenterPartOfImgs{j} = MidMid;
end

save '..\Code\GetCenterPartOfTrainImgs.mat';
```

A.1.2. Sample Code for SURF feature extraction

```
clc;
format compact;
load ('CroppedTrainingImg.mat');
nc = length(CroppedTrainingImg);

for surfCnt=1:nc
   image = CroppedTrainingImg{surfCnt};
   gray = rgb2gray(image);
   bw = imbinarize(gray, 0.3);
   fPoints = detectSURFFeatures(gray);
   strongestPoints = fPoints.selectStrongest(10);
   [imageFeatures, validPoints] = extractFeatures(gray,strongestPoints);
   TrainSurfFeatureDB{surfCnt,:} = imageFeatures;
end

save '..\Code\TrainSurfFeatureDB.mat';
```

A.1.3 Sample Code for Colour, Texture, GIST feature extraction

```
InputImage = imresize(InputImage, [300 NaN]);

% Get Color moment of the image
% Extract RGB Channel
R=InputImage(:,:,1); % For Red Channel
G=InputImage(:,:,2); % For Green Channel
B=InputImage(:,:,3); % For Blue Channel
```

```
% Extract Statistical features
% 1] MEAN
meanR=mean2(R);
meanG=mean2(G);
meanB=mean2(B);

% 2] Standard Deviation
stdR=std2(R);
stdG=std2(G);
stdB=std2(B);

% Convert to grayscale if image is RGB
if ndims(InputImage) == 3
  Grayimg = rgb2gray(InputImage);
end

% Create the Gray Level Cooccurance Matrices (GLCMs)
glcms = graycomatrix(Grayimg);

% Derive Statistics from GLCM
stats = graycoprops(glcms,'Contrast Correlation Energy Homogeneity');
Contrast = stats.Contrast;
Correlation = stats.Correlation;
Energy = stats.Energy;
Homogeneity = stats.Homogeneity;

% Calculate Hues Seven Invarient Moments
image = double(InputImage);
if ~exist('mask','var')
   mask = ones(size(image,1),size(image,2)); %if mask is not defined select the whole image
end
% computation of central moments up to order 3
for i=1:1:4
   for j=1:1:4
      nu(i,j) = Centr_Moment(image, mask,i-1,j-1);
   end
end
% computation of scale invariant moments using central moments of up to order 3
eta = zeros(3,3);
for i=1:1:4
   for j=1:1:4
      if i+j >= 4
         eta(i,j) = (double(nu(i,j))/(double(nu(1,1)).^(double((i+j)/2)))); %scale invariant moment matrix
      end
   end
end

%Calculation of various invariant Hu's moments
HuesSevenInvarientMoments(1) = eta(3,1) + eta(1,3);

HuesSevenInvarientMoments(2) = (eta(3,1) - eta(1,3))^2 + (4*eta(2,2)^2);

HuesSevenInvarientMoments(3) = (eta(4,1) - 3*eta(2,3))^2 + (3*eta(3,2) - eta(1,4))^2;

HuesSevenInvarientMoments(4) = (eta(4,1) + eta(2,3))^2 + (eta(3,1) + eta(1,4))^2;

HuesSevenInvarientMoments(5) = (eta(4,1) - 3*eta(2,3))*(eta(4,1) + eta(2,3))*((eta(4,1) + eta(2,3))^2 -
3*((eta(3,2) + eta(1,4))^2)) + (3*(eta(3,2) - eta(1,4)))*(eta(3,2) + eta(1,4))*(3*(eta(4,1) + eta(2,3))^2 -
(eta(3,2) + eta(1,4))^2);
```

```
HuesSevenInvarientMoments(6) = (eta(3,1) - eta(1,3))*((eta(4,1)+eta(2,3))^2 - (eta(3,2)+ eta(1,4))^2)
+ 4*eta(2,2)*((eta(4,1) + eta(2,3))*(eta(3,2) + eta(1,4)));

HuesSevenInvarientMoments(7) = (3*eta(3,2) - eta(1,4))*(eta(4,1) + eta(2,3))*((eta(4,1) + eta(2,3))^2 -
3*(eta(3,2)-eta(1,4))^2) - (eta(4,1) - 3*eta(2,3))*(eta(3,2) + eta(1,4))*(3*(eta(4,1) + eta(2,3))^2 -
(eta(3,2) + eta(1,4))^2);


% Derive statistics from GIST
% GIST Parameters:
clear param
param.orientationsPerScale = [8 8 8 8]; % number of orientations per scale
param.numberBlocks = 4;
param.fc_prefilt = 4;
% Computing gist:
gist1 = LMgist(InputImage, '', param);


F1 = [Contrast,Correlation,Energy,Homogeneity,meanR,meanG,meanB,stdR,stdG,stdB];
F2 = gist1;
F3 = [Contrast,Correlation,Energy,Homogeneity,meanR,meanG,meanB,stdR,stdG,stdB,gist1];
F4 = [meanR,meanG,meanB,stdR,stdG,stdB,gist1];
F5=HuesSevenInvarientMoments;


% Function to calculate the central moment of interested image region

function cen_mmt = Centr_Moment(image,mask,p,q)
if ~exist('mask','var')
   mask = ones(size(image,1),size(image,2)); %if mask is not specified, select the whole image
end
image = double(image);
% moments necessary to compute components of centroid
m10 = moment(image,mask,1,0);
m01 = moment(image,mask,0,1);
m00 = moment(image,mask,0,0);
% components of centroid
x_cen = floor(m10/m00);
y_cen = floor(m01/m00);
cen_mmt =0;
for i=1:1:size(mask,1)
   for j=1:1:size(mask,2)
      if mask(i,j) == 1
         % calculating central moment
         cen_mmt = cen_mmt + (double(image(i,j))*((i-x_cen)^p)*((j-y_cen)^q));
      end
   end
end
end


% Function to calculate any ordinary moment of the input image region
 function m = moment(image,mask,p,q)
if ~exist('mask','var')
   mask = ones(size(image,1),size(image,2));   %if mask is not specified, select the whole image
end
image = double(image);
m=0;
for i=1:1:size(mask,1)
   for j=1:1:size(mask,2)
```

```
        if mask(i,j) == 1
            m = m + (double((image(i,j))*(i^p)*(j^q))) ;
        end
    end
end
end

% Function to calculate the scale-invariant moment of interested image region
function eta = SI_Moment(image, mask)
image = double(image);
if ~exist('mask','var')
    mask = ones(size(image,1),size(image,2)); %if mask is not defined select the whole image
end
% computation of central moments up to order 3
for i=1:1:4
    for j=1:1:4
        nu(i,j) = Centr_Moment(image, mask,i-1,j-1);
    end
end
% computation of scale invariant moments using central moments of up to order 3
eta = zeros(3,3);
for i=1:1:4
    for j=1:1:4
        if i+j >= 4
            eta(i,j) = (double(nu(i,j))/(double(nu(1,1)).^(double((i+j)/2))));
        end
    end
end
end
```

A.1.4 Sample code for Feature DB creation

```
% Processing Training Set
NT = numel(trainCenterPartOfImgs);
NF1= 10; %no. of features for each images in model1
NF2= 512; %no. of features for each images in model2
NF3= 522; %no. of features for each images in model3
NF4= 518; %no. of features for each images in model4
NF5= 7; %no. of features for each images in model5

TrainingFeatureDB1=zeros(NT,NF1);   %feature vector of color moments and texture
TrainingFeatureDB2=zeros(NT,NF2);   %feature vector of Gist
TrainingFeatureDB3=zeros(NT,NF3);   %feature vector of color moments, texture and Gist
TrainingFeatureDB4=zeros(NT,NF4);   %feature vector of color moments and Gist
TrainingFeatureDB5=zeros(NT,NF5);   %feature vector of Hues Sene Invariant Moments

 fprintf('Processing Training Set\n');

fprintf('Extract the feature and Create Training Feature DB\n');
for cnt=1:NT
    InputImage=(trainCenterPartOfImgs{cnt});
    disp (cnt);
    FeatureExtraction;
    TrainingFeatureDB1(cnt,:)=F1;
    TrainingFeatureDB2(cnt,:)=F2;
    TrainingFeatureDB3(cnt,:)=F3;
    TrainingFeatureDB4(cnt,:)=F4;
    TrainingFeatureDB5(cnt,:)=F5;
```

```
    close;
end

save '..\Code\TrainingFeatureDB.mat';
disp('Traing Feature DB creation completed successfully.\n')
```

A.1.5 Sample code to Create different Classification models

```
load ('..\Code\TrainingFeatureDB.mat');
load ('..\Code\TraingDataLabel.mat');
colors = 'rgbcmyk'; markers = 'hsdp';
ClassificationModel1=fitcecoc(TrainingFeatureDB1, TraingDataLabel);
ClassificationModel2=fitcecoc(TrainingFeatureDB2, TraingDataLabel);
ClassificationModel3=fitcecoc(TrainingFeatureDB3, TraingDataLabel);
ClassificationModel4=fitcecoc(TrainingFeatureDB4, TraingDataLabel);
ClassificationModel5=fitcecoc(TrainingFeatureDB5, TraingDataLabel);
ClassificationModel6 =
fitcknn(TrainingFeatureDB1,TraingDataLabel,'NumNeighbors',5,'Standardize',1);
ClassificationModel7 =
fitcknn(TrainingFeatureDB2,TraingDataLabel,'NumNeighbors',5,'Standardize',1);
ClassificationModel8 =
fitcknn(TrainingFeatureDB3,TraingDataLabel,'NumNeighbors',5,'Standardize',1);
ClassificationModel9 =
fitcknn(TrainingFeatureDB4,TraingDataLabel,'NumNeighbors',5,'Standardize',1);
ClassificationModel10 =
fitcknn(TrainingFeatureDB5,TraingDataLabel,'NumNeighbors',5,'Standardize',1);


save '..\Code\ClassificationModel.mat';
disp('Classification Model creation completed successfully.\n')
```

A.1.6 Sample code to test different classification models

```
load ('TestingFeatureDB.mat');
load ('ClassificationModel.mat');
load ('TestDataLabel.mat');

[PredictedLabels1, Score1] = predict(ClassificationModel1, TestingFeatureDB1);
[ConfusionMatrix1, Label1] = confusionmat(TestDataLabel, PredictedLabels1);
accuracy1 = sum(PredictedLabels1 == TestDataLabel)/numel(TestDataLabel);

[PredictedLabels2, Score2] = predict(ClassificationModel2, TestingFeatureDB2);
[ConfusionMatrix2, Label2] = confusionmat(TestDataLabel, PredictedLabels2);
accuracy2 = sum(PredictedLabels2 == TestDataLabel)/numel(TestDataLabel);

[PredictedLabels3, Score3] = predict(ClassificationModel3, TestingFeatureDB3);
[ConfusionMatrix3, Label3] = confusionmat(TestDataLabel, PredictedLabels3);
accuracy3 = sum(PredictedLabels3 == TestDataLabel)/numel(TestDataLabel);

[PredictedLabels4, Score4] = predict(ClassificationModel4, TestingFeatureDB4);
[ConfusionMatrix4, Label4] = confusionmat(TestDataLabel, PredictedLabels4);
accuracy4 = sum(PredictedLabels4 == TestDataLabel)/numel(TestDataLabel);

[PredictedLabels5, Score5] = predict(ClassificationModel5, TestingFeatureDB5);
```

```
[ConfusionMatrix5, Label5] = confusionmat(TestDataLabel, PredictedLabels5);
accuracy5 = sum(PredictedLabels5 == TestDataLabel)/numel(TestDataLabel);

[PredictedLabels6, Score6] = predict(ClassificationModel6, TestingFeatureDB1);
[ConfusionMatrix6, Label6] = confusionmat(TestDataLabel, PredictedLabels6);
accuracy6 = sum(PredictedLabels6 == TestDataLabel)/numel(TestDataLabel);

[PredictedLabels7, Score7] = predict(ClassificationModel7, TestingFeatureDB2);
[ConfusionMatrix7, Label7] = confusionmat(TestDataLabel, PredictedLabels7);
accuracy7 = sum(PredictedLabels7 == TestDataLabel)/numel(TestDataLabel);

[PredictedLabels8, Score8] = predict(ClassificationModel8, TestingFeatureDB3);
[ConfusionMatrix8, Label8] = confusionmat(TestDataLabel, PredictedLabels8);
accuracy8 = sum(PredictedLabels8 == TestDataLabel)/numel(TestDataLabel);

[PredictedLabels9, Score9] = predict(ClassificationModel9, TestingFeatureDB4);
[ConfusionMatrix9, Label9] = confusionmat(TestDataLabel, PredictedLabels9);
accuracy9 = sum(PredictedLabels9 == TestDataLabel)/numel(TestDataLabel);

[PredictedLabels10, Score10] = predict(ClassificationModel10, TestingFeatureDB5);
[ConfusionMatrix10, Label10] = confusionmat(TestDataLabel, PredictedLabels10);
accuracy10 = sum(PredictedLabels10 == TestDataLabel)/numel(TestDataLabel);

save '..\Code\TestModelForFlowerRecognition.mat'
```

A.1.7 Sample code to recognize flower

```
clear;  clc; format compact;
fprintf('RECOGNIZE FLOWER\n');
load 'TrainSurfFeatureDB.mat';
load ('ClassificationModel.mat');
load ('CroppedTrainingImg.mat');


% Delete all images from temp folder.
myFolder = '..\Temp';
filePattern = fullfile(myFolder, '*.png'); % Change to whatever pattern you need.
theFiles = dir(filePattern);
for k = 1 : length(theFiles)
  baseFileName = theFiles(k).name;
  fullFileName = fullfile(myFolder, baseFileName);
  fprintf(1, 'Now deleting %s\n', fullFileName);
  delete(fullFileName);
end

disp('Please select a flower to be recognized.\n')

% get user input image file
startingFolder = '..\ValidationDat\';
if ~isfolder(startingFolder)
    startingFolder = '..\ValidationData\';
end

% Get the name of the file that the user wants to use.
defaultFileName = fullfile(startingFolder, '*.*');
[baseFileName, folder] = uigetfile(defaultFileName, 'Select a file');
if baseFileName == 0
    return;
```

```
end

% Copy the selected image to temp folder.

fullFileName = fullfile(folder, baseFileName);
QueryInputImg = cell(1,1);
QueryInputImg{1} = imread(fullFileName);
figure (1),imshow(QueryInputImg{1});title('Query Image');
copyfile(fullFileName,'..\Temp\')


srcTempDir = ('..\Temp\'); %Directory for Testing data set
srcTempFiles=dir(fullfile(srcTempDir,'*.png'));
totalTempFiles = length(srcTempFiles);
global sFlower;


% Processing Input Image
NT = totalTempFiles; % Total number of testing files
NF1= 10; %no. of features for each images in model1
NF2= 512; %no. of features for each images in model2
NF3= 522; %no. of features for each images in model3
NF4= 518; %no. of features for each images in model4
NF5= 7; %no. of features for each images in model5

TestingFeatureDB1=zeros(NT,NF1);
TestingFeatureDB2=zeros(NT,NF2);
TestingFeatureDB3=zeros(NT,NF3);
TestingFeatureDB4=zeros(NT,NF4);
TestingFeatureDB5=zeros(NT,NF5);

%Read the RGB image
rgbQueryImage=QueryInputImg{1};
% Resize Input Image
rgbQueryImage = imresize(rgbQueryImage,[300 NaN]);
figure (2),imshow(rgbQueryImage);

fprintf('Processing Image Set\n');

% Get the center part of the query Image
[r, c, ~] = size(rgbQueryImage);
aspectRatio = r/c;
r = 150;
c = r/aspectRatio;
rgbQueryImage= imresize(rgbQueryImage,[r,c]);
figure (3),imshow(rgbQueryImage);
% Get the dimensions of the image.  numberOfColorBands should be = 3.
[rows,columns,numberOfColorBands] = size(rgbQueryImage);
% Get the rows and columns to split at taking care to handle odd-size dimensions:
col1 = 1;
col2 = floor(columns/3);
col3 = col2 + 1;
col4 = col3 + 1;
row1 = 1;
row2 = floor(rows/3);
row3 = row2+1;

% Now crop
upperLeft = imcrop(rgbQueryImage, [col1 row1 col2 row2]);
upperMid = imcrop(rgbQueryImage, [col2 row1 col3 row2]);
```

```
upperRight = imcrop(rgbQueryImage, [col3 row1 columns - col3 row2]);
MidMid = imcrop(rgbQueryImage, [col2 row2 col3 row3]);
lowerLeft = imcrop(rgbQueryImage, [col1 row3 col2 row2]);
lowerRight = imcrop(rgbQueryImage, [col3 row3 columns - col2 rows - row2]);
rgbQueryImageCentralPart = MidMid;
figure (4),imshow(rgbQueryImageCentralPart);
fprintf('Center part of query image is extracted. \n');

%Step 1 fetch SURF feature

%Give path for Testing data Folder
figure (5),imshow(rgbQueryImage);
grayImage = rgb2gray(rgbQueryImage);
fPoints = detectSURFFeatures(grayImage);
 strongestPoints = fPoints.selectStrongest(10);
 [imageFeatures, validPoints] = extractFeatures(grayImage,strongestPoints);
 A{1,:} = imageFeatures;

for intCount=1:length(TrainSurfFeatureDB)
        [matchedPairs, matchMetric] = matchFeatures(TrainSurfFeatureDB{intCount},A{1});
        percentMatch = (1-mean(matchMetric))*100;
        PM{intCount} = percentMatch;
End

MaxAccuracyClass1 = (max(cell2mat(PM))); %100
disp(MaxAccuracyClass1);

if  isnan(MaxAccuracyClass1)
    PredictedLabels11 = "UnKnown";
else
    index = find(cell2mat(PM)== MaxAccuracyClass1);
    load ('TraingDataLabel.mat');
    PredictedLabels11 = TraingDataLabel(index);
    PredictedLabels11 = mode( PredictedLabels11 );
    disp PredictedLabels11;
    disp TraingDataLabel(index);
end

% Extract other features from the image
 InputImage = rgbQueryImageCentralPart;
figure (6),imshow(InputImage);
FeatureExtraction;
FeatureDB1(1,:)=F1;
FeatureDB2(1,:)=F2;
FeatureDB3(1,:)=F3;
FeatureDB4(1,:)=F4;
FeatureDB5(1,:)=F5;

disp('Feature extraction completed successfully.\n')

disp('Predict the flower.\n')
%USING SVM
[PredictedLabels1, Score1] = predict(ClassificationModel1, FeatureDB1);
[PredictedLabels2, Score2] = predict(ClassificationModel2, FeatureDB2);
[PredictedLabels3, Score3] = predict(ClassificationModel3, FeatureDB3);
[PredictedLabels4, Score4] = predict(ClassificationModel4, FeatureDB4);
[PredictedLabels5, Score5] = predict(ClassificationModel5, FeatureDB5);
```

```
%USING KNN
[PredictedLabels6, Score6] = predict(ClassificationModel6, FeatureDB1);
[PredictedLabels7, Score7] = predict(ClassificationModel7, FeatureDB2);
[PredictedLabels8, Score8] = predict(ClassificationModel8, FeatureDB3);
[PredictedLabels9, Score9] = predict(ClassificationModel9, FeatureDB4);
[PredictedLabels10, Score10] = predict(ClassificationModel10, FeatureDB5);

%Save all prediction in a matrix to determine the highly predicted class.
PredictedLabel=zeros(11);
PredictedLabel(1)=PredictedLabels1;
PredictedLabel(2)=PredictedLabels2;
PredictedLabel(3)=PredictedLabels3;
PredictedLabel(4)=PredictedLabels4;
PredictedLabel(5)=PredictedLabels11;
PredictedLabel(6)=PredictedLabels6;
PredictedLabel(7)=PredictedLabels7;
PredictedLabel(8)=PredictedLabels8;
PredictedLabel(9)=PredictedLabels9;
PredictedLabel(10)=PredictedLabels10;
PredictedLabel(11)=PredictedLabels11;

 HighlyPredictedLabel = mode( PredictedLabel ); %returns the sample mode of PredictedLabel , which
is the most frequently occurring value in PredictedLabel .
disp ('HighlyPredictedLabel: ');
disp (HighlyPredictedLabel(1));
iPredLebel = HighlyPredictedLabel(1);

if iPredLebel == 1
    sFlower = 'Hibiscus';
    disp ('Hibiscus');
    figure (7),imshow(rgbQueryImage);title('Classified as: Hibiscus');
elseif iPredLebel == 2
    sFlower = 'Rose';
     disp ('Rose');
    figure (7),imshow(rgbQueryImage);title('Classified as: Rose');
elseif iPredLebel == 3
    sFlower = 'Poppy';
     disp ('Poppy');
    figure (7),imshow(rgbQueryImage);title('Classified as: Poppy');
elseif iPredLebel == 4
    sFlower = 'Daisy';
     disp ('Daisy');
    figure (7),imshow(rgbQueryImage);title('Classified as: Daisy');
elseif iPredLebel == 5
    sFlower = 'Calla Lily';
    disp ('Calla Lily');
    figure (7),imshow(rgbQueryImage);title('Classified as: Calla Lily');
elseif iPredLebel == 6
    sFlower = 'Plumeria';
    disp ('Plumeria');
    figure (7),imshow(rgbQueryImage);title('Classified as: Plumeria');
elseif iPredLebel == 7
    sFlower = 'Morning Glory';
    disp ('Morning Glory');
    figure (7),imshow(rgbQueryImage);title('Classified as: Morning Glory');
elseif iPredLebel == 8
    sFlower = 'Marigold';
    disp ('Marigold');
    figure (7),imshow(rgbQueryImage);title('Classified as: Marigold');
elseif iPredLebel == 9
```

```
    sFlower = 'Tagar';
    disp ('Tagar');
    figure (7),imshow(rgbQueryImage);title('Classified as: Tagar');
elseif iPredLebel == 10
    sFlower = 'Sunflower';
    disp ('Sunflower');
    figure (7),imshow(rgbQueryImage);title('Classified as: Sunflower');
    elseif iPredLebel == 11
    sFlower = 'Gazania';
    disp ('Gazania');
    figure (7),imshow(rgbQueryImage);title('Classified as: Gazania');
elseif iPredLebel == 12
    sFlower = 'Calendula';
    disp ('Calendula');
    figure (7),imshow(rgbQueryImage);title('Classified as: Calendula');
else
    sFlower = 'Unknown';
    disp ('Unknown');
    figure (7),imshow(rgbQueryImage);title('Classified as: Unknown');
end

figure (8)
subplot(1,3,1),imshow(rgbQueryImage);title('Query Image');
subplot(1,3,2),imshow(rgbQueryImageCentralPart);title('Centerpart of the image');
subplot(1,3,3),imshow(rgbQueryImage);title("Classified as: " + sFlower);

save '..\Code\ValidationFeatureDB.mat';
```