FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Image processing module for Camellia identification

Francisco Eça da Silva Próspero Luís

Mestrado Integrado em Engenharia Eletrotécnica e de Computadores

Supervisor: Luís António Pereira de Meneses Corte-Real

Second Supervisor: Teresa Cristina de Sousa Azevedo Terroso

July 6, 2018

Abstract

Plant recognition is an arduous task normally performed by specialists; it is time-consuming and very complex. For non-experts this is a frustrating process, therefore a tool which solves this problem is of great importance not merely for protecting biodiversity but also to motivate enthusiasts and researchers in these fields of studies.

Nowadays, there is a big array of techniques and technologies, such as, mobile devices, digital cameras, image processing, pattern recognition algorithms and databases that can be used by everyone for any specific task. The aim of this work is to combine some of the previously stated techniques and available data to achieve the best possible ratio of matching camellia flower shapes. Some of the stated technologies and algorithms will be described in the literature review.

The workflow is described following the timeline of development. The first phase was to collect a dataset to work on since there was no existing camellia species dataset. From the collected dataset, different sub-datasets were derived following increasing difficulty levels for shape identification. Some segmentation methods were developed: autonomous and user-aided. The next phase was to use the previously segmented images to extract features using algorithms such as SIFT and HOG, for later classification and detection with an SVM classifier. Another approach was to train a model with YOLO to classify the different camellia shapes.

In conclusion, it was possible to correctly identify with good accuracy the camellia's shape using HOG and color features as well as with YOLO's trained model with segmented frontal flowers. Using the last stated model, it was also possible to classify the shapes with the original unsegmented frontal flower images, nonetheless with lower accuracy.

Acknowledgments

I would first like to thank my thesis advisor Prof. Luís António Pereira de Meneses Corte-Real and second advisor Prof. Teresa Cristina de Sousa Azevedo Terroso of the Faculty of Engineering at the University of Porto. The door to Prof. Corte-Real and Prof. Terroso office was always open whenever I ran into a trouble spot or had a question about my research or writing. They consistently allowed this paper to be my own work but steered me in the right direction whenever they thought I needed it.

Finally, I must express my very profound gratitude to my parents, grandparents and to my girlfriend for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

Francisco Eça da Silva Próspero Luís

“Every problem has in it the seeds of its own solution.
If you don’t have any problems, you don’t get any seeds.”

Norman Vincent Peale

Contents

List of Figures

List of Tables

Abreviaturas e Símbolos

2D	Two Dimensions
AR	Aspect Ratio
CDS	Center Distance Sequence
CV	Computer Vision
DSLR	Digital Dingle-Lens Reflex
EnS	Entropy Sequence
GPS	Global Positioning System
GPU	Graphics Procesing Unit
HOG	Histogram of Oriented Gradients
HSV	Hue Saturation Value
ICL	Intelligent Computing Laboratory
ICM	Intersecting Cortical Model
IDSC	Inner-Distance Shape Context
k-NN	k-Nearest Neighbors
LBP	Local Binary Patterns
MARCH	Multiscale-arch-height
ML	Machine Learning
NN	Neural Network
ORM	Object Relational Mapping
PDF	Probability Density Function
PNN	Probabilistic Neural Network
QR	Quick Response
RGB	Red Green Blue
SC	Shape Context
SIFT	Scale-Invariant Feature Transform
SMSD	Simple and Morphological Shape Descriptors
SVM	Support Vector Machines
YOLO	You Only Look Once

Chapter 1

Introduction

This research looks forward to better understand what are the best approaches to recognize shapes in camellia species flowers from a set of pictures. It will take into consideration state-of-the-art approaches used in other projects so that the final goal can be met with the best possible accuracy.

1.1 Context

People are getting more conscious of the environment and the life that surrounds us. Global warming and related disasters are creating an unstable world for living creatures, which makes it important to catalog and analyze data related to this. This way, we can prevent future hazards and save species from extinction. Regarding plants, the main issue of cataloging all its species is their huge variety and the similarity between them. For instance, there are more than 400.000 plant species around the world [9] and there are still many that haven't been cataloged yet. Furthermore, many people are interested in studying this subject. However, for non-experts identifying and learning about a certain species requires a lot of research time; for example, to determine what type of plant is being analyzed, first it is needed to check what is the most peculiar part of the plant - leaves, flowers, bark or, branches. After, the features are evaluated and compared with a table or list made by experts. By the end of this process, the species can be identified.

All the above reasons make flora classification very troublesome. Since there are so many different species, the problem was split into several smaller ones, identifying a specific species. Each species have distinct characteristics that define them, if there is one identification module for each species, they can all be joined and create a general plant classifier. To tackle this problem, this work focuses on the development of a shape classifier for the camellia species.

1.2 Motivation

Not having the possibility to quickly identify a specific species may be a problem in a big range of activities such as Education, Botanic, Agriculture, Research, hobbyists and so on. Developing a tool that could solve this issue is of big interest for everyone taking part in activities regarding this subject. However, there are many adversities concerning the automatic identification of plants. Each plant has its own characteristics such as if it is a tree, a bush or grass; if it has fruits, flowers or leaves. Each case has its own particularities like shape, texture, and color [8].

There are already some mobile and web applications that try to implement this feature, such as Pl@ntnet [10], which is the best-known application, Plantix [11] or PictureThis [12]. Each of these applications uses different techniques for the identification process. However, depending on the picture and plant some get better results than others. Each application has a different focus; some are more inclined for crops, others for general plants. Some only work with species of certain world regions, while others are more global. Taking this into account, a module to identify among camellia shapes could be useful for using similar methods for other plants as well as implementing it in an existing plant classifier. For this reason this research focuses on the identification of the camellia species shape.

1.3 Objectives

Having a shape classifier for the camellia species would speed up the process of identification of both known and unknown species because manual identification is laborious and needs a lot of expertise. The widespread usage of it would allow users of any background to be able to learn and study these species, anywhere at any time. Also, it would make cataloging easier for experts, which would speed up important research projects. In this research, different identification techniques will be studied and tested to better understand the feasibility of such classifier. Image processing and machine learning algorithms [13, 14, 15] will be analyzed with different datasets to conclude which ones could be the best approach to achieve the goal of this study. User interaction will also be considered to enhance the whole process and obtain better results.

1.4 Contribution

The main contributions of the developed work to the scientific community are listed below:

- A dataset of camellia images with around 750 pictures taken with different equipment of 11 species;
- Ground truth labeling of the acquired images with a specialist;
- Proposed segmentation methods for flower segmentation;
- A trained model for camellia shape classification.

1.5 Document Structure

This document contains five chapters, being introduction the first one. In the next chapter, different technologies, computer vision techniques, machine learning algorithms, classification means, and other methods will be studied and explained so that it facilitates the decision process of which techniques to use to achieve the goal. Chapter 3 follows the workflow of the development of each method. The fourth chapter is where the obtained results are showned and discussed. The last chapter is the conclusion of this work as well as the possible future development.

Chapter 2

Literature review

Different approaches have been developed throughout the years that try to distinguish and group images of the same theme. With current algorithms and techniques some important applications have been developed, such as, traffic sign recognition for cars, license plate readers, Quick Response (QR) code readers, factory quality tests of the product for size and shape, and so on. But when the group of images to analyze is of a considerable size and their characteristics divergent, for example, identification of obstacles in marine environments, identification of wild animals as well as recognition of plant species, new applications need to be developed.

The way images are acquired is of extreme importance because they are the foundations of the identification process. In this chapter some techniques, algorithms, datasets, and existent applications will be explored.

2.1 Taxonomic rank

Biologists use taxonomic rank to organize all living beings in related hierarchical groups. The International Code of Nomenclature for algae, fungi, and plants (also known as Melbourne Code), is the set of rules that defines the botanical ranks [16]. The most specific rank is the one that categorizes a single species, above it more general ranks exist that relate some common characteristics of different species until the most general:

- Domain is the most general of all; it consists of 3 groups, Archaea, Bacteria, and Eukarya. The first two are single-celled organisms whose cells have no nucleus;
- Kingdom is composed of 5 groups, this is the rank that separates plants from animals as well as others;
- Phylum more known in botany as division. This is the first separation in similar characteristics;
- Class is normally determined by a taxonomist, in this group there is no hard rule in how to define a class;

- Order as in the group above, there is no hard rule for this rank, normally suffixes are used to explain a group of species;
- Family, taxonomists also define this one as no definite rule can be applied. In botany suffixes like -aceae are used;
- Genus rank has three main rules, all descendants of an ancestral taxon are grouped together, it should not be expanded needlessly and distinctness. This is the first of the binomial nomenclature which defines a species with only to ranks;
- Species is the basic unit of the taxonomic rank, it defines each different species. Being the last in the binomial nomenclature.

2.2 Computer Vision

Computer vision is the field of study that uses computers to try to replicate the human visual system. Using an image as an input, it analyses and interprets all the necessary information to get a decision or conclusion as an output [17].

The objective of Computer Vision techniques in this work is the automation of repetitive and laborious tasks that can use visual features to determine the next stage. Therefore, it is possible to distinguish among different kinds of plants depending on their unique visual characteristics.

Some research has already been done on this subject. The main features that define a plant are shape, texture, color, margin, and vein structure of its leaves, flowers, and fruits. The studies that had better results used a combination of shape and texture analysis of leaves and color of flowers [8] .

Some methods, techniques and descriptors will be explained in the next sub-sections.

2.2.1 Simple and morphological shape descriptors (SMSD)

There are a lot of features that can be extracted from the leaves, flowers, and fruits, as shown in 2.1.

Multiple measurements can be made on an image to extract their main characteristics. In the following table (Table 2.1) simple and morphological shape descriptors used in researches on this subject are listed. In the same species of plants, although similar, the leaf's shape, texture, margin, vein and color are slightly different among them. These descriptors alone are not enough for accurate plant recognition of a big dataset, but they are important to reduce the number of possible species [1]. With small datasets of distinct plants they should give good results.

Table 2.1: SMSD descriptors an their short definitions. Adapted from [8].

Diameter	Longest distance between two points of the same shape
Major axis length	Line segment that connects the stem and the tip of the leaf

Minor axis length	Maximum width perpendicular to major axis of the leaf
Area	Area of the leaf
Perimeter	Perimeter of the leaf
Centroid	Geometric center
Aspect Ratio	Major axis divided by minor axis
Roundness	Difference between the leaf and the circle that surrounds it
Compactness (Shape complexity)/ Rectangularity	Ratio of perimeter and area. How rectangular a shape is
Eccentricity	Ratio of the distance between the foci of the ellipse and its major axis length
Narrow Factor	Ratio of the diameter and major axis
Perimeter ratio of diameter	Ratio between the perimeter and diameter
Perimeter ratio of major axis length and minor axis length	Ratio between the perimeter and major axis
Convex hull	Ratio of the perimeter and the sum of the major and minor axis length
Perimeter convexity	Smallest region that contains the leaf's region and is convex
Area convexity	Ratio of convex perimeter and perimeter
Area ratio of convexity	Difference of the convex hull area and the leaf's area
Sphericity	Ratio of the radius of the inside bounding box's circle and the radius of the outside bounding box circle
Equivalent diameter	Diameter of a circle with the same area of the leaf
Ellipse variance	Represents the mapping error of a shape to fit an ellipse with same covariance matrix as the shape
Smooth factor	Ratio between organ's area smoothed by a 5x5 rectangular averaging filter and one smoothed by a 2x2 rectangular averaging
Leaf width factor	The shape is sliced perpendicular to the major axis in several strips. A ratio of each slice and the length of the leaf
Area width factor	The shape is sliced perpendicular to the major axis in several strips. A ratio between each slice's area and the leaf area
Porosity	Counting of the holes inside the shape

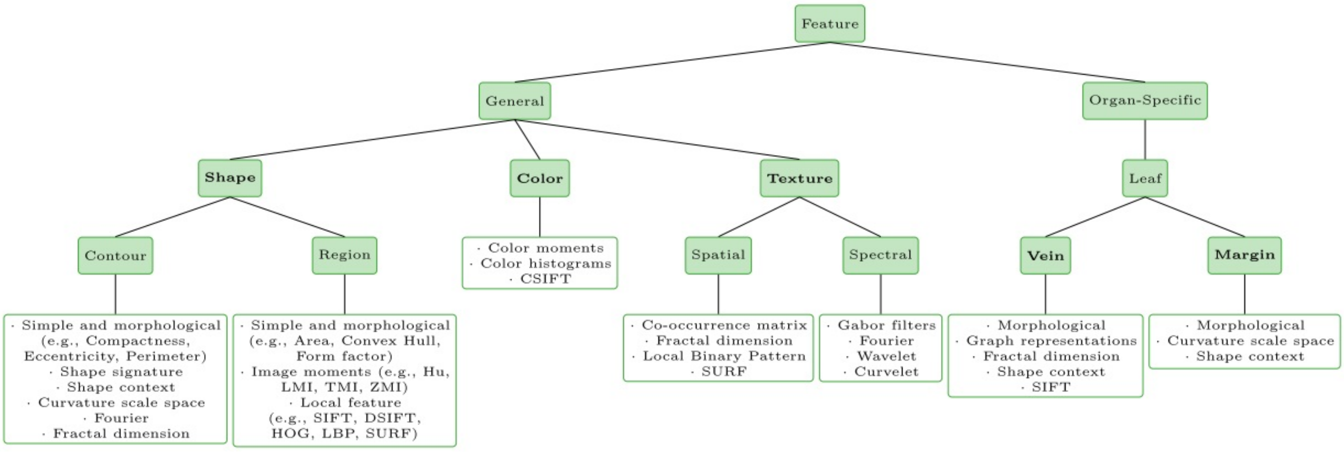


Figure 2.1: Plant features. From [1]

2.2.2 Center Distance Sequence (CDS)

This method of leaf identification lies on the contour of the leaf. Using only the centroid of the leaf and one of the SMSD described above, CDS measures the Euclidean distance between that point and the leaf's contour, This means that the shape is described with an array of distances from the centroid to the edge. Hence, the shape is defined with those values.

$$D(x,y) = \sqrt{(x-x_c)^2 + (y-y_c)^2} \tag{2.1}$$

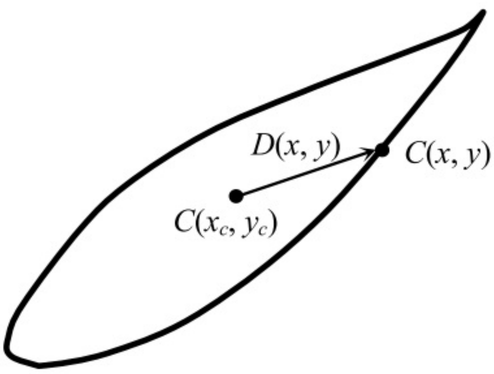


Figure 2.2: Center Distance. From [2].

$$CDS = D(x_i, y_i) \,|\, 0 < i < M-1 \tag{2.2}$$

D(x,y) is the distance. Xc is the x position of the center. Yc is the y position of the center. C(x,y) is the point on the contour.

2.2.3 Inner-Distance Shape Context (IDSC)

This shape descriptor is based on the shape descriptor Shape Context (SC), where at each point of the leaf contour the distance and the angles of all other points is quantized creating a 2D histogram [18]. Using SC some difficulties to identify the same leaf appeared if the relative pose of the petiole and the blade is different in two images. To solve this problem, IDSC replaces the Euclidean distance and relative angles with "inner-distance" and "inner-angles", which makes it rotation and articulation invariant [19]. The 2D histogram generated by this method showed better results. Nonetheless, leaves with similar global shape and different local details tend to be misclassified.

2.2.4 Scale-Invariant Feature Transform (SIFT)

To identify objects in an image it is needed to collect unique features of it. But to detect them, something else is needed, because in a different image lighting conditions, angle of the image and proximity to the subject may differ. The Scale-Invariant Feature Transform (SIFT) algorithm is used to detect and describe local features in an image while invariant to translation and rotation of the object in the image. This algorithm results in a collection of key points, as shown in 2.3. that can be used to match the same object in other image. This algorithm is used in applications such

Figure 2.3: SIFT keypoints in an image.

as object recognition, video tracking, gesture recognition and many others.

2.2.5 Local Binary Patterns (LBP)

LBP is an algorithm used for texture classification. To generate the LBP feature vector first the window needs to be divided into cells (16x16 for example), then for each pixel of that cell (256 pixels in this case), compare it with its neighbors (N8 connectivity). If the neighboring pixel is greater then the center one a "0" is written, otherwise "1". This will give a binary number with 8 digits, which can be converted to decimal. Then the histogram of each examined window can be created with the decimal number obtained earlier, this will show how many times that pattern occurs. Finally, all the cells are concatenated creating a feature vector[20].

2.2.6 Entropy Sequence (Ens)

If we have a group of images as an output from a single image it is possible to create a feature vector with an entropy sequence. This sequence has the advantages of invariance to scale, translation and rotation. The entropy sequence (EnS), proposed by Ma [21], is defined as:

$$EnS[n] = -p_1[n]\log 2p_1[n] - p_0[n]\log 2p_0[n] \tag{2.3}$$

where p1[n] and p0[n] represent the probability when $Yij[n] = 1$ and $Yij[n] = 0$ in the output $Y[n]$ separately.

The crucial part of this method is its uniqueness to every image because it is dependent on the image texture [21].

2.2.7 Histogram of Oriented Gradients (HOG)

The Histogram of oriented gradients is a scale-invariant feature descriptor which uses gradient changes in equal sized cells of the image using overlapping local contrast normalization for better results. This descriptor gives detailed information about abrupt changes in the image which indicates the presence of an object or features of it. This is an important shape descriptor since it shows a pattern that can represent an object and its details as shown in 2.4.

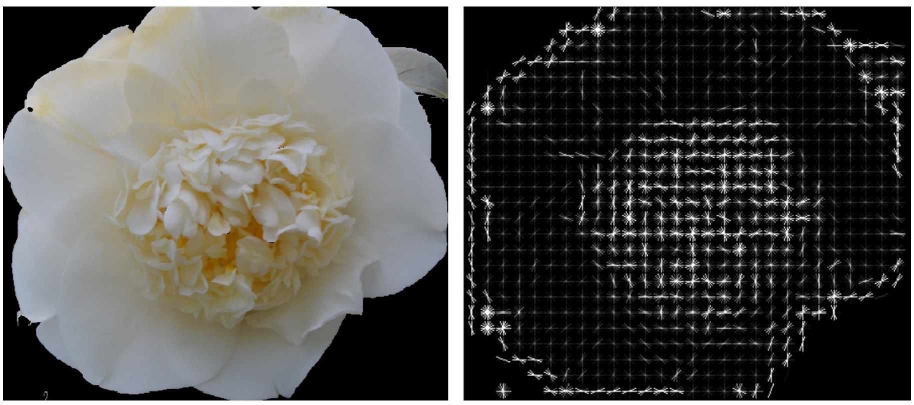


Figure 2.4: Original image on the left. HOG features on the right.

It is easy the interpret the visual result of such descriptor.

2.3 Machine Learning

Machine learning is a field of computer science that gives computers the ability to learn without being explicitly programmed [22]. This is particularly important for autonomous categorization systems since it can learn from well-defined examples given as inputs (datasets) and therefore predict to which group does a new item belongs to. As the results are strongly related to the quality of the datasets, different ones need to be tested as well as distinct combinations of hidden layers and neurons.

Some studies using these techniques have been performed in this subject, with unprecedented results.

2.3.1 Support Vector Machines (SVM)

Support vector machines are based in creating boundaries between a set of different objects. In Figure 2.5, a complex structure is shown, so classifying it is not linear. This is where SVM come helpful since they can extract features to create that kind of non-linear separation line. Through a

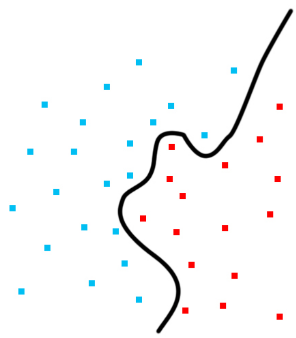


Figure 2.5: Separation into two groups

set of mathematical formulas this method can obtain small errors when the margin is well defined [2].

2.3.2 k-Nearest Neighbors (k-NN)

The k-Nearest Neighbors k-NN is one of the first and most widely used classification algorithms[8]. Nowadays, it is still used since it is simple and showed to be competitive with more advanced algorithms [1].

The classifier compares an unknown sample by comparing it to the K nearest neighbors among a set of known samples. The first thing to compute is the distance between the unknown sample and the training ones. Various techniques can be used to calculate that distance, such as, Euclidean distance. To conclude the process, the distances are sorted according to the smaller distance among samples. This way the k nearest samples are determined and a class is assigned to the unknown one [23].

2.3.3 Probabilistic Neural Network (PNN)

Probabilistic neural network is a feedforward neural network that can be used wither for classifier or pattern recognition. A Probability Distribution function (PDF) of each class is approximated by a Parzen window and a non-parametric function. Then the Bayes' rule is employed to each class PDF, picking the one with higher probability as the new input. This method minimizes the mis-classification probability [24].

2.3.4 Fuzzy k-Nearest Neighbors (Fuzzy k-NN)

Fuzzy k-Nearest Neighbors was developed in 1985 by [23], and it usually improves significantly the performance in classification problems. It uses membership functions to various classes. The following equation provides class membership to a sample according to the K-NN training sample's distance.

$$u_i(x) = \frac{\sum_{j=1}^{k} u_i(x^j)(\|x - x^j\|^{\frac{(-2)}{(m-1)}})}{\sum_{j=1}^{k}(\|x - x^j\|^{\frac{(-2)}{(m-1)}})}\ ,\ i = 1,...,c \tag{2.4}$$

with m the fuzzy strength parameter where 1<m<2, it determines how distance is weighted. K is the number of nearest neighbors $Ui(x)$ is the membership of test sample x to class i $||x - x(j)||$ is the distance between test sample and nearest training samples

Various methods can be used for measuring distance like in the k-NN method. The membership function can be crisp, where it is either 1 or 0, but it can also be "fuzzier" having a probability of taking part in other class. After processing all the information, the sample is assigned to a class with highest membership [23].

2.3.5 Intersecting Cortical Model (ICM)

The Intersecting Cortical Model (ICM) is a simplified model of the Pulse-Coupled Neural Network (PCNN). This method's objective is to reduce the processing power and time, while keepzing the effectiveness. It is based in elements of biological models. This model has two greater advantages, it reduces the computational cost and it has a good performance. The model is shown in 2.6.

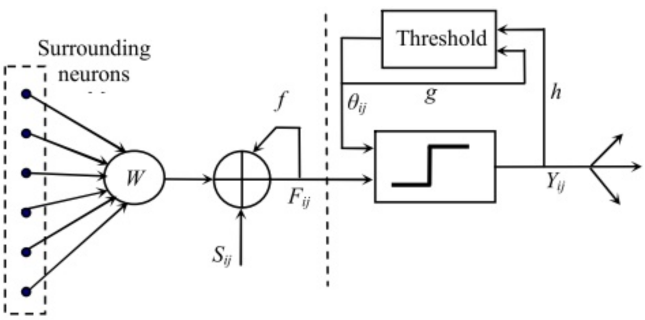


Figure 2.6: ICM representation. From [3].

The model is mathematically described by 3 equations:

$$F_{ij}[n+1] = fF_{ij}[n] + S_{ij} + W\{Y\}_{ij} \tag{2.5}$$

$$Y_{ij}[n+1] = \{1\ ,if\ F_{ij}[n+1] > \Theta_{ij}[n]0,\ Otherwise \tag{2.6}$$

$$\Theta_{ij}[n+1] = g\Theta_{ij}[n] + hY_{ij}[n+1] \tag{2.7}$$

Sij is the input image (scale to 1.0 maximum value) Yij is the firing state of the neuron Y is the output, binary image f, g, h are scalars n is the iteration number W is the function describing the connections between neurons The parameters are usually set by experience.

Since this model only consists in 3 equations and two neuron oscillators (neuron potential and threshold), upon simulation it can generate a sequence of spikes where features can be extracted from. This method can be very helpful to extract key segments for image identification [2].

2.3.6 You Only Look Once v3 (YOLO)

YOLO [25] is a state-of-the-art system that detects objects in images or videos. It can also be used in real-time.

This method applies only one neural network to the whole image at once. Then the image is devided into regions where bounding boxes are predicted in accordance to the propability of the trained classes, Figure 2.7.

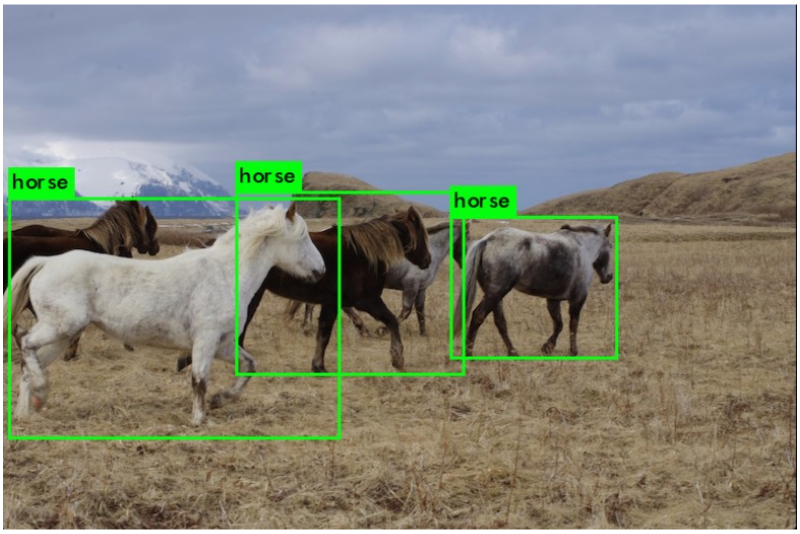


Figure 2.7: An example of a labeled image using YOLO v3. From [4].

The system can be trained for different datasets. Firstly, each training image need to have a twin file with the same name that countains the bounding box coordinates of the objects to detect and the corresponding class. After, a model can be trained for any number of iterations. A graph with the error is shown each iteration. When the error is low, the smaller the better, the training process can be stopped and the last generated weights for the neural network should be used.

The final step is to test the model in images different than the ones used for training. Bounding boxes will be shown with labels around the detected objects.

2.4 Identification Keys

Using identification keys can be of extreme importance in the development of the application since it can decrease the number of species to compare with just some input from the user. This would speed up substantially the identification process because the group of plants to compare would be smaller and well defined.

2.4.1 Single-access Keys

Single-access key is a method of identification based in tree graphs. They can be dichotomous or polytomous.

A dichotomous key is a binary tree, where each node is a question and the branches are binary answers. By following a certain path down the tree, a result will be given according to all previous information. Biologists and botanists usually use these keys to identify species since this is a straightforward process, 2.8. They are easy to use and can lead to good results in a short time. Sometimes, a polytomous key is used. The difference between both is that this one is not binary, so each node can have any number of branches.

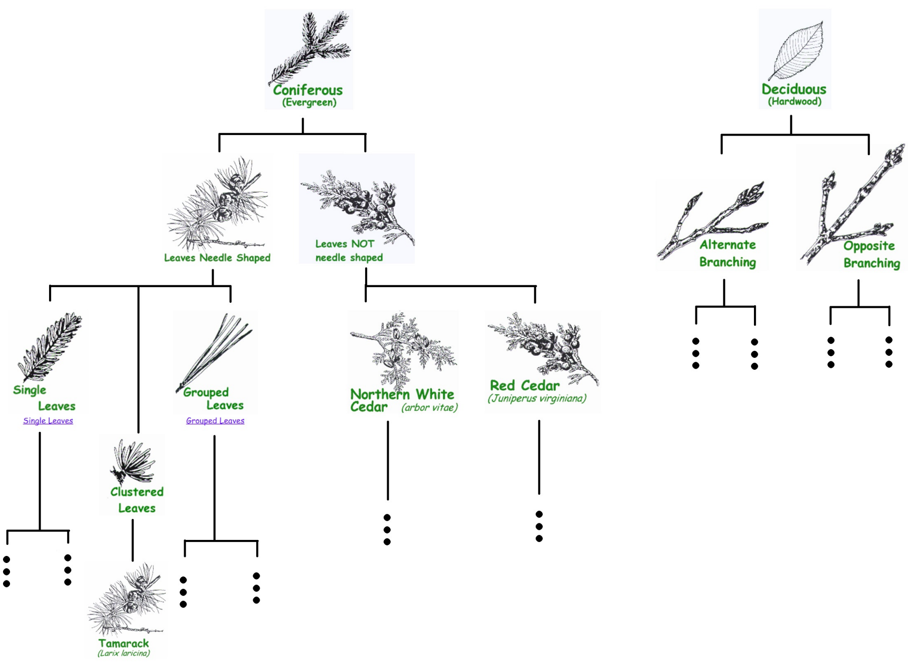


Figure 2.8: An example of a single-access key Adapted [5].

2.4.2 Multi-access Keys

A multi-access key overcomes the problem of the single-access key which requires a fixes sequence as identification steps. This method looks forward to generalizing how to describe a species, for example, if we are trying to identify a plant in winter, it is probably different than in the summer, using the simpler more restrict method of single-access key, it would be impossible to do it. That is when this approach gets really convenient since the user can adapt the key to a particular organism making it possible to use in any situation for that specific purpose. Normally, these keys are computer aided since they require some processing power.

2.5 Mobile and Web applications

Mobile and Web applications are becoming more important for the development of a classification tool since big datasets are needed to get more accurate results even if they insert some noise [26]. The problem of this kind of user interaction is the correctness (or lack of it) of the data introduced. So, if this problem could be solved, datasets would be more complete, and the recognition process would get unprecedented results and maybe a global plant identifier could be implemented with greater accuracy.

There are some mobile and web applications already developed such as Pl@ntnet [10], which is the best-known application, Plantix [11], Garden Answers or PictureThis [12], but there are still flaws that make this research important in this field. First, the main problem is the correctness of user data. As the recognition process takes place, it shows to the user plants that the software finds to be like the one uploaded, but for non-experts, the results shown are difficult to compare. Also, some applications are focused on specific regions of the globe or plant species.

2.5.1 Pl@ntnet

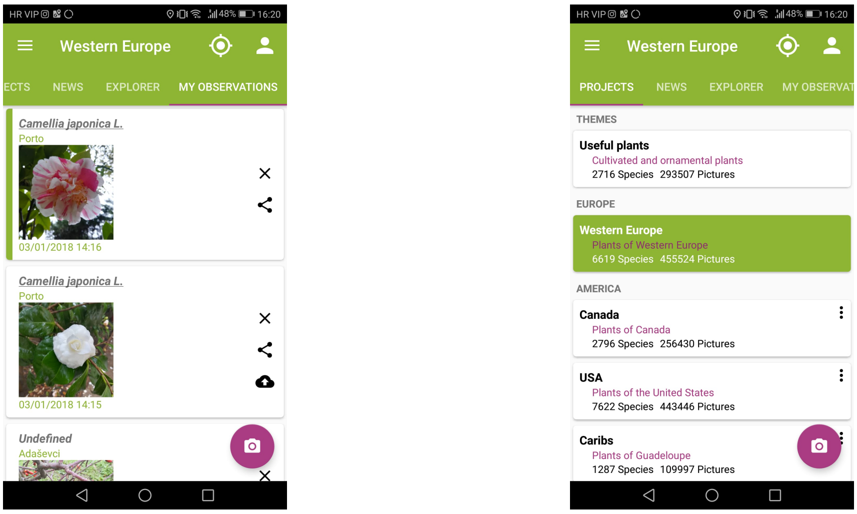
Pl@ntnet is the most widely used application on plant recognition since it has partnerships with some research laboratories. It has a big community contributing to the system and helping each other when there is no match with the automatic categorization system. Since it is a mobile application it can be used wherever in the world as long as there is internet connection. Pictures can be uploaded later as well. Its interface is shown in 2.9a.

This mobile application compares user pictures to a private botanical database, it is not metioned which one is it, [10] showing the most similar results and the final selection is made by the user. This mobile application is supported by user donations. Table 2.2 shows advantages and disadvantages of this mobile application.

Table 2.2: Summary of Pl@ntnet advantages and disadvantages

Advantages	**Disadvantages**

Most widely used application for this purpose.	When some matches are found there is no way to compare the original image with the match side by side
Big community and partnerships with research laboratories	There is no information about diseases or pests
Before the recognition phase the user needs to indicate what part of the plant is in the photo	There is no input from the user other than the picture
There is a tab where it is possible to see the new submissions to the application	It is constrained to a few regions of the world
If one user thinks the plant is mis-classified, he can suggest other species name	
There is a voting system to evaluate the picture quality	
There is no need for payment to use the application, but donations are accepted	
There is a possibility to see all the pictures taken from a specific species	

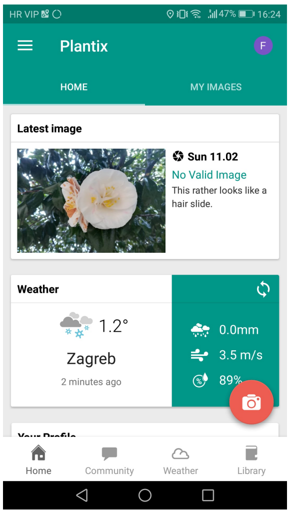


(a) Identification module (b) Tab of projects where this application is used

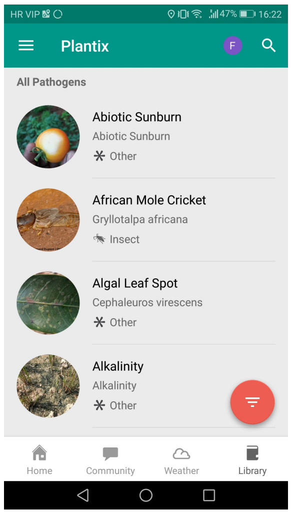
Figure 2.9: Screenshots of Pl@ntnet mobile application.

2.5.2 Plantix

This mobile application is more focused on diseases and pests [11] of crops than in the identification of the plant itself. It is made specially for farmers and gardeners to help them take care of their plants. After the recognition process solutions to take care of the crops are presented to the user. Its interface is shown in 2.10a. Table 2.3 shows advantages and disadvantages of this mobile application.



(a) Identification module



(b) Tab of uploaded images

Figure 2.10: Screenshots of Plantix mobile application.

Table 2.3: Summary of Plantix advantages and disadvantages

Advantages	Disadvantages
Help on how to use the application	Low number of possible plants to identify
Community interaction	Information about part of the plant in the picture could be asked to the user
Focuses on diseases and pests	
Possibility to indicate the crop to search just for the disease	

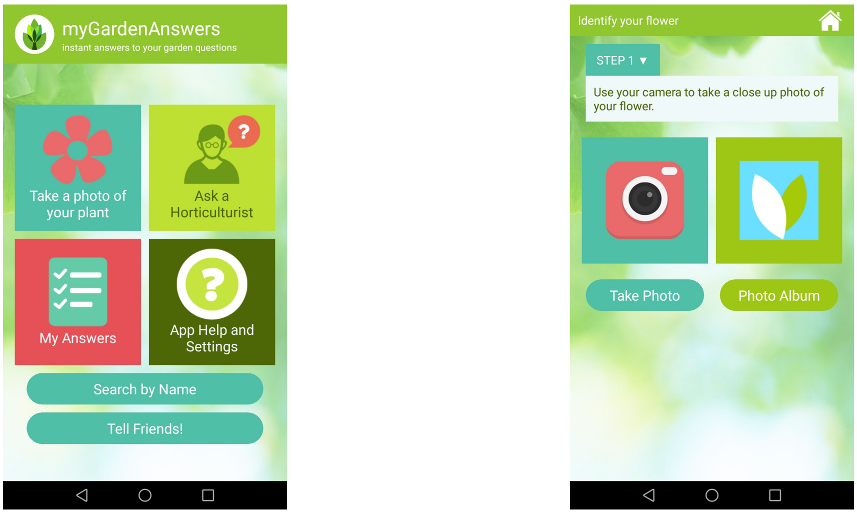
2.5.3 Garden Answers

Garden answers is focused mainly in common outdoor and indoor plants. It is useful for a hobbyist that wants to learn more about his garden. Using a picture as input it gives the plants' name and some details. The interesting part of this mobile application is the information about possible diseases and infections that can affect the plant in question. Its interface is shown in 2.11a. Table 2.4 shows advantages and disadvantages of this mobile application.

Table 2.4: Summary of GardenAnswers advantages and disadvantages

Advantages	Disadvantages
It has details about approximately 20000 plant species	The recognition module matches the input with plants that are very different

There is a lot of information about each plant	Payment is needed to ask the expert for help on the identification



(a) Garden Answers menu (b) Garden Answers camera module

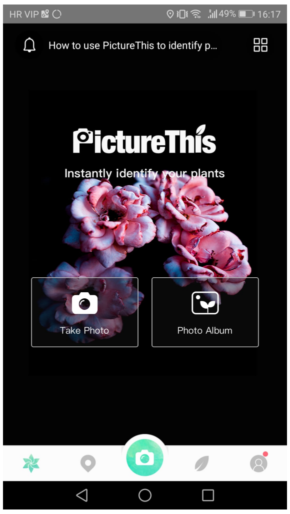
Figure 2.11: Screenshots of GradenAnswers mobile application.

2.5.4 PictureThis

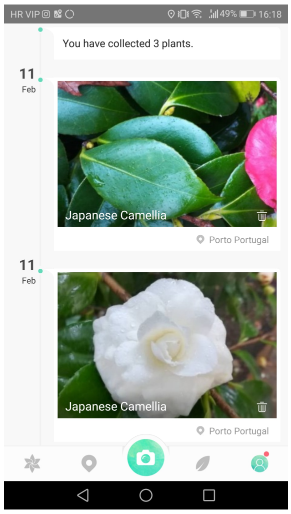
The focus of this application is identification of plants and creating a map with their position, so that everyone can go and see that species in the real world [12]. Its interface is shown in 2.12a. Table 2.5 shows advantages and disadvantages of this mobile application.

Table 2.5: Summary of PictureThis advantages and disadvantages

Advantages	**Disadvantages**
Pictures of matching plants are shown side by side with the original	There is no information about diseases or pests
There is a lot of information about each plant such as description, cultivation, uses, meaning, facts, and poems	It just can identify 4000 species
Every user can report or comment on the picture	There is no input from the user other than the picture
There is a tab to see newly uploaded pictures	
The is a tab for help on using the application	
The pictures uploaded by users are shown in the map	
Accuracy of 84 percent	



(a) PictureThis menu



(b) PictureThis identification module

Figure 2.12: Screenshots of PictureThis mobile application.

2.6 Datasets

Choosing the right datasets is one of the most important parts of the process since it will define the effectiveness of the machine learning algorithms and classifiers, if some plants are not well categorized it will insert noise in the whole system making it perform poorly. Furthermore, datasets are also crucial to test the effectiveness and validation of the computer vision algorithms.

There are some plant datasets available online, but first, it is important to choose what kind of flora to focus on: specific country's plants, global or only certain species. The most used datasets will be described.

- ImageCLEF dataset is the most widely used because of its size, correct data, and diversity of species. It consists in 10000 different species from Western Europe, North America, also plant species from all around the world as well as endangered species [9];
- The Swedish leaf dataset contains 15 different tree species with 75 leaf pictures each, all the trees are from Sweden [8];
- The leafsnap dataset with 185 tree species from the Northeastern United States, containing both high-quality lab images and images taken with phones [8];
- Flavia dataset contains 32 species with a range of 50 to 77 images per species. Those leaves were sampled on the campus of the Nanjing University and the Sun Yat-Sen arboreum, Nanking China [1];
- ICL dataset contains leaves of 220 plant species with a range of 26 to 1078 images per plant. It has two subsets (50 species each), one of them has 50 easily distinguishable species by humans and the other contains 50 with more similar species among them. The leaves were collected at Hefei Botanical Garden in Hefei, China [1];

- Oxford Flower 17 dataset contains 17 flower species that are unique in visual appearance in their natural habitat [17];
- Oxford Flower 102 dataset contains flowers of 102 flower classes that exist in the United Kingdom [1].

Using the Swedish leaf dataset a combination of shape and texture analysis got an accuracy of 99.25% using the DS-LBP descriptor with a classifier Fuzzy k-NN [3]. The second best only evaluated the shape with MARCH descriptor and had an accuracy of 97.33% [3]. With the Inteligent Computing Laboratory (ICL) dataset (220 species) the best result was 99.48% using only shape features with I-IDSC descriptor and 1-NN classifier. In this one the combination of shape and texture was not tested. Using the FLAVIA dataset, using shape and texture with Ens and CDS descriptors and SVM classifier got the best results with an accuracy of 97.80%. In LifeCLEF 2017 a team of researchers got a ratio of 92% using a dataset created by specialists as well as a noisy dataset collected from the internet [26]. These results were better than the ones that only used the trusted dataset, which means that using pictures captured from people all over the world can help the algorithm to get better results. When there are more images for each species from different angles and plants it helps the algorithm to be less biased in some cases, that is why it achieved better results when evaluating different pictures than the ones used to train the network [26].

In [14], using a dataset of 24 medical plants, a team of researchers could get a ratio of 90.1% using a random forest classifier using a 10-fold cross-validation technique [14].

2.7 Final Considerations

In this chapter different methodologies, techniques, algorithms and datasets used in similar projects were studied and described so that in the planning phase a path can be choosen according to the best performing algorithms.

Having in mind all the previous topics it is needed to decide what would be the best combination of methods and/or datasets to achieve the desired goal. For this matter, some considerations need to be taken into account, such as, the computational power available, the possibility of using different techniques together, the existing software and the time constraint for the completion of the project.

Chapter 3

Workflow

In this chapter, the development workflow of this project is described, followed by the procedures used in each step. It is organized in a linear form explaining how and why the decisions were made in each case, starting from the first phase of acquiring the necessary images, finishing with classification phase a classifier, as shown in Figure 3.1.

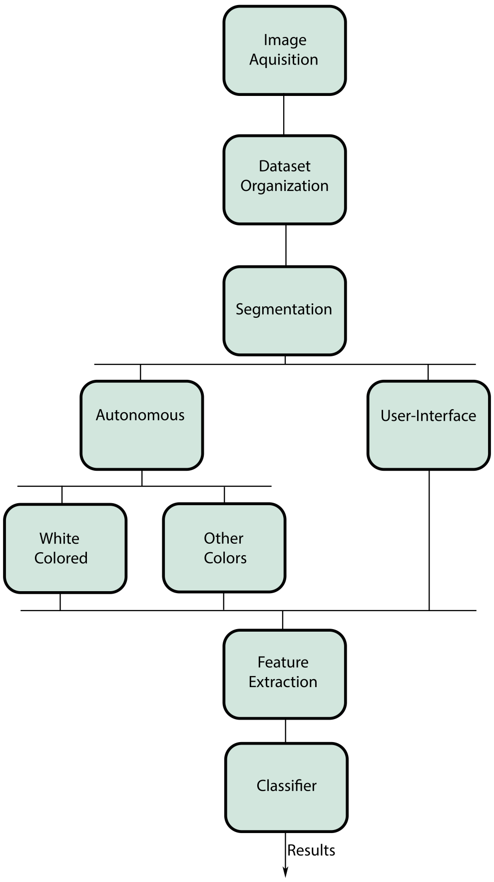


Figure 3.1: Workplan Scheme.

3.1 Dataset Aquisiton

As referred on the previous chapters a controlled and well-categorized dataset is crucial for the success of any image identification module.

Following the proposed workflow, the first part of the project was to acquire and organize a useful dataset. Studying the existing species of camellia, it was found that there are more than 3000 species of this family[27], where some are natural while others are hybrids. To identify them all a big dataset is required because each species needs to have a considerable amount of images in different environments to cover all possible situations, the problem is that there are no existing camellia species datasets available. So, a new dataset needed to be created.

Different species bloom in various times of the year. Since this research did not have large temporal flexibility, it was decided to select some species that were blooming during the acquisition phase, from February to June. To help on the camellia species identification, we resort to an expert, Mr. Assunção, owner of a camellia nursery in Portugal which has more than 1300 different species of camellia [28]. According to Maria Ferreira in [6], the first phase to identify these species is to look at its shape, which can be separated into 6 different types, as shown in Figure 3.2.

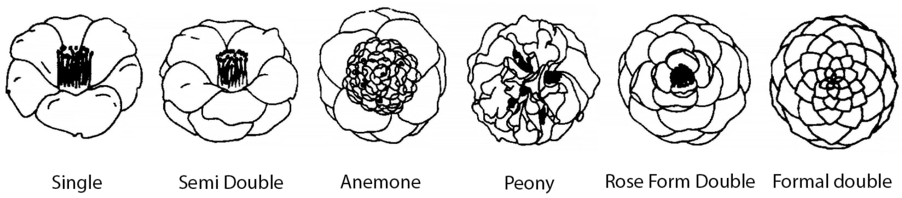


Figure 3.2: Different camellia shapes, from [6].

The definition of each type is the following:

- Single - Single layer of petals. No more than eight;
- Semi-Double - Two or more layers of petals with stamens;
- Anemone - One or more outer layers of petals, sorrounding a central mass of stamens and petaloids;
- Peony - A mass of a mixture of petals and petaloids which can have some stamens showing;
- Rose Form Double - Layers of petals closed with stamens showing;
- Formal Double - Many rows of petals imbricated, never showing stamens.

Starting by identifying the flower's shape makes further identification easier since after understanding its shape the number of possibilities decreases considerably. So, it was decided to collect at least 1 species of each type to have data of every shape. However, there are other important features such as color and size of the flower (Figure 3.3), as well as the brightness and saturation of the leaves. Even if their shape is different, some species can have the same coloration and size.

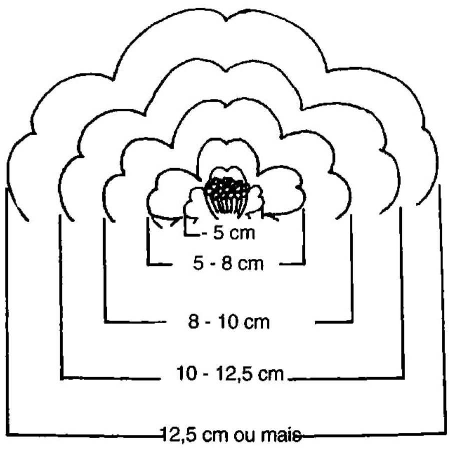


Figure 3.3: Intervals of flower size in camellia species. < 5 cm miniature, 5-8 cm small, 8-10 cm medium, 10-12.5cm big, > 12.5cm very big, From [6].

Taking all the previous constraints into consideration, it was decided to create a dataset that contains species from every shape.

To be systematic in the acquisition process the pictures were taken in a defined order for each chosen species:

- one picture of a flower and a leaf with a ruler for size comparison;
- five frontal different flower pictures without highlights;
- five frontal different flower pictures with highlights and shadows;
- two with some angle to three different flowers;
- five of different leaves without highlights;
- five of different leaves with highlights and shadows;
- three with various flowers in the same picture;
- various pictures with another camera without any specific order.

Images were captured with a Nikon D5100 digital single-lens reflex (DSLR) camera and a regular mobile phone camera. Two cameras were used to produce a richer and more robust dataset containing photos with different characteristics. Furthermore, the user probably has a different equipment, so keeping the dataset broad is important.

All the pictures were taken in a structured manner. It consists in 11 species, Figure 3.4. There are 2 species of each shape, except one, since it was the only species of its shape blooming in the acquisition phase.

Figure 3.4: One image sample of the 11 camellia species present in the acquired image dataset.

Since the focus of this work is mainly to classify the shape of each flower, a first sub-dataset - hereinafter referred to as Frontal flowers dataset - was separated from the main one. It consists of all 133 frontal flower images collected. This is the simplest, but the most important sub-dataset since the flower shape is more clear when facing the camera lens than in images where it is tilted. Some frontal flower image examples are shown in Figure 3.5. If the accuracy is high with this subset of images, the application is ready to be used with just one constraint: the picture needs to be taken with the flower facing the lens.

A second sub-dataset - hereinafter referred to as not frontal flowers dataset - consists of all the single flower pictures: frontal, angled, highlights and shadows. This one has a greater level of difficulty than the first one because the same shape of a flower may have different features when facing other directions and under different light conditions. The objective behind creating this sub-dataset was to make the plant image acquisition process easier for the user, sometimes it is hard to take a perfect frontal picture of the flower when in the field. It consists of 432 images,

Figure 3.5: Four image examples of dataset D1: frontal flower images.

some examples are shown in Figure 3.6.

Figure 3.6: Four image examples of dataset D2: not frontal flowers

A third sub-dataset - hereinafter referred to as leaf dataset-, of circa 250 images, consists of the leaves of each species. There is a slight difference in brightness and color among species' leaves. Some examples are shown in Figure 3.7.

Figure 3.7: Four image examples of dataset D3: leaf images

The last dataset - hereinafter referred to as Multiple flowers dataset - consists of all the taken pictures plus some collected images taken by other people. The level of difficulty is very high because it has all types of images: multiple flowers, tilted, out-of-focus, highlights. This would be the final dataset to be tested to understand how robust is the developed identification module. It has more than 750 images, some examples are shown in Figure 3.8.

Figure 3.8: Four image examples of dataset D4: Multiple flowers.

All these datasets were created with a crescent level of difficulty for identification, nevertheless, the focus will be on the first one since it is crucial to find the shape in a first phase. It can take more effort for the user to get the image, but it would imply less effort than using identification keys. This last dataset is built for a more general approach to this problem.

3.2 Flower Segmentation

To extract flower features it is needed to isolate it from the background. In this section, the segmentation process is described.

Looking at a flower, the human eye can easily distinguish the flower of a plant, mainly because of the distinct shape of it and the color difference between it and the leaves.

Two distinct sets of methods were developed: autonomous and using some user inputs.

3.2.1 Autonomous Segmentation

The camellia flowers' colors range from completely white with yellow stamen to dark red, meaning that both blue and green colors are not present.

The first approach - hereinafter referred to as Method A - was to create a color interval that represented the presence of a flower. The intervals were based in the Hue Saturation Value (HSV) color space, more precisely in the Hue channel.

Observing Figure 3.9 it is possible to understand the importance of the HSV color space to identify colors in images. Being Hue the channel that better represents the color separation, it was used to find the flower in the image. It is considered by many to be the best for color analysis in images[29][14][30][31].

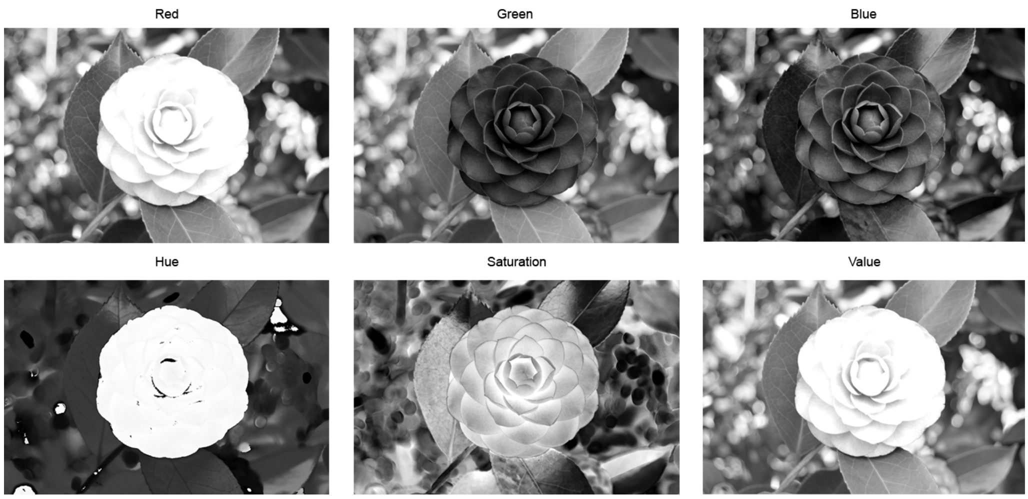


Figure 3.9: The top images are the individual Red Green Blue (RGB) channels. The bottom images are the individual HSV channels

The V channel was discarded; after all, the difference in highlights should be eliminated to avoid losing information of the flower. It was chosen as an interval of [165^o, 15^o] on the H channel, corresponding to the color red, Figure 3.10. The mask contained an S interval of [50-255] to represent different saturation of the color red.

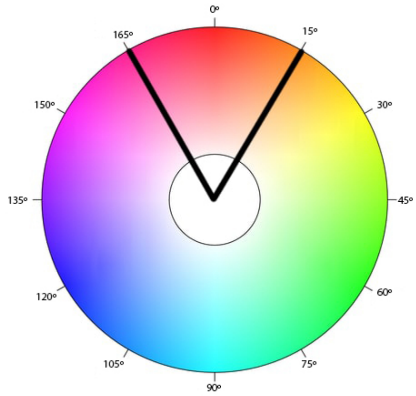


Figure 3.10: Selection of red region of the Hue values.

These values contained some colors found on the dataset's flowers and it was an initial segmentation method to prove the effectiveness of color-based segmentation. Finally, the original image was masked with the defined intervals segmenting the flowers, Figure 3.11.

Figure 3.11: Three examples of the autonomous segmentation method A: on the left the original image. On the right, the masked image.

The second test was to use a threshold technique which could be more broadly applied to a wider range of cases. After observing the different flowers in various color representations, it was clear that the green channel was very important because in most cases the background consists of leaves, which are green.

Once again, the V channel was discarded, and the image was again converted to RGB. This process removes the lightening differences of the original image. After, the Otsu thresholding technique [32] was applied to the green channel of this image. This method consists in separating the image pixels into two classes, foreground and background, by calculating an optimal threshold level so that the inter-class variance is maximal. Finally, the biggest contour found on the thresholded image is selected. The segmentation process is illustrated in Figure 3.12.

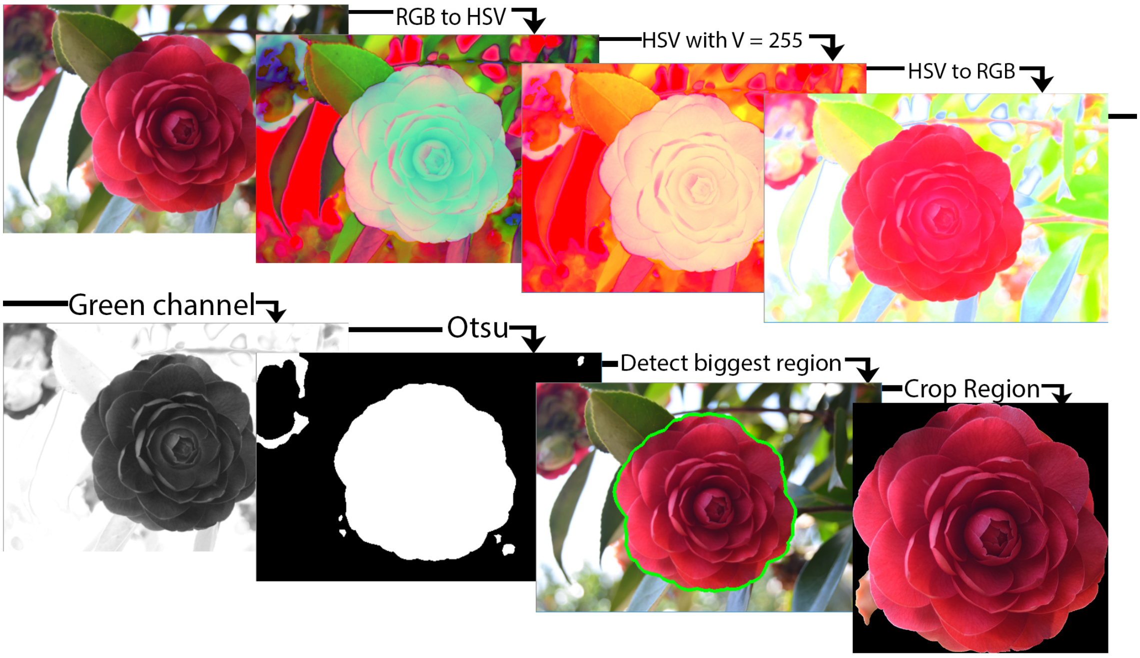


Figure 3.12: Scheme representing all steps involved in the autonomous segmentation method B.1.

To segment white flowers another approach was taken - hereinafter referred to as Method B.2. It was observed that using Otsu threshold directly in the green channel of the original image worked for the white flowers Figure 3.13, reducing the required steps of the previously described method B.1.

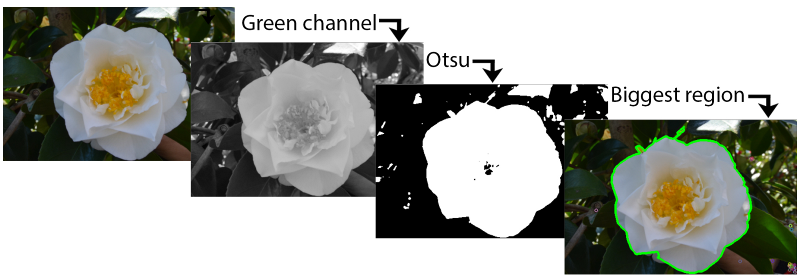


Figure 3.13: Scheme representing all steps involved in the autonomous segmentation method B.2.

With the flowers correctly segmented both methods B.1 and B.2 are merged to autonomously segment the whole dataset - hereinafter referred to as Method B. To choose between B.1 and B.2 a prior hue histogram analyzes is done to check for the presence of the color red, Figure 3.14.

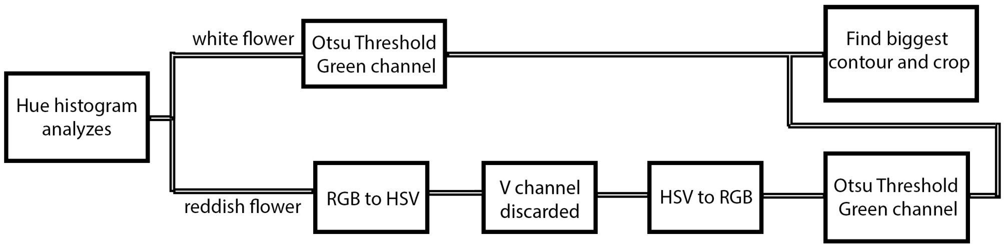


Figure 3.14: Autonomous segmentation algorithm scheme.

3.2.2 User-interface segmentation

To create a more robust and reliable method to prevent wrongly segmented images, a user-aided algorithm was developed - hereinafter referred to as Method C. Using the GrabCut [33] algorithm a user can select the region of interest to be segmented in the image. The GrabCut algorithm determines a threshold that most accurately segments the area inside the rectangle.

The original image is shown to the user for him to select the flower region. After running the GrabCut algorithm, the segmented image is shown again so that the user can approve it if correctly identified or select a new region if not. When the image gets accepted the biggest contour is detected, which indicates the flower. The image is then cropped so that the segmented flower fits perfectly in the image with a black background. Only then it is stored in an image file for posterior feature extraction, Figures 3.15 and 3.16.

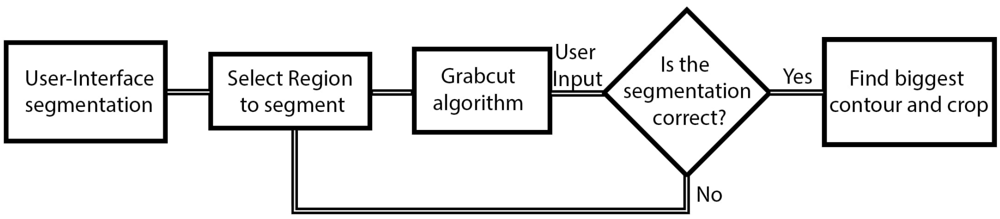


Figure 3.15: User-Interface segmentation algorithm scheme.

The user interaction methodology could minimize the errors due to poor segmentation since it is based on human interaction that can, most of the times, easily detect flowers on images. Also, it won't be as time-consuming as a pure manual segmentation process because the only thing to do is to select a region.

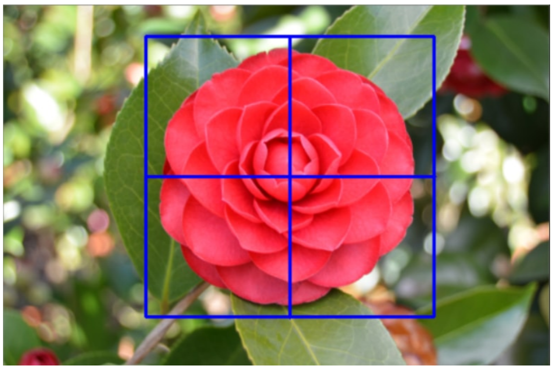


Figure 3.16: On the left image, user selects region of the flower. On the right image, result of the method C - user-interface segmentation algorithm

3.3 Feature Extraction

Humans easily recognize objects around them because there are specific features that our brains connect to them. This process takes place with experience from the day we are born, we learn and strengthen our visual recognition knowledge every day as we see the world. To mimic this process, Computer Vision tries to collect information through algorithms that run in real-world images.

3.3.1 SIFT features

In the literature review chapter, some algorithms and methods were studied in this matter. To check which were the features that best described these flowers some of them were tested. Since the most relevant aspect of flowers is its shape, first the SIFT algorithm was tested.

The SIFT extracts local features - key points - from the image, which are invariant to rotation and translation. The resulting key points are important because they describe the object using only some relevant points, Figure 3.17. With the features, it is possible to find the object in other images. To find these features the original image was used as input to the SIFT algorithm with a mask containing only the flower contour obtained in the segmentation step described previously.

Figure 3.17: Examples of SIFT key points. The green region indicates where the SIFT key points must be extracted from. The points indicate the extracted SIFT key points.

3.3.2 HOG features

Looking at each individual species it is possible to observe distinct features in the petal arrangement: some have petals all over the flower, while others have them only on the edge. To extract these features the studied HOG algorithm was chosen because it detects abrupt changes in the image which in this case could indicate changes between petals and stamen.

The HOG method considers that the shape and texture of objects can be described by the distribution of the intensity gradients or in the direction of the object's border. Firstly, the segmented flower image is converted to grayscale since the HOG algorithm works in a normalized color space to avoid errors caused by different channels. The feature vector and the HOG image are then returned. Both consist of small-sized blocks containing the oriented gradient of that region. Figure 3.18 shows the image representation of the extracted HOG features.

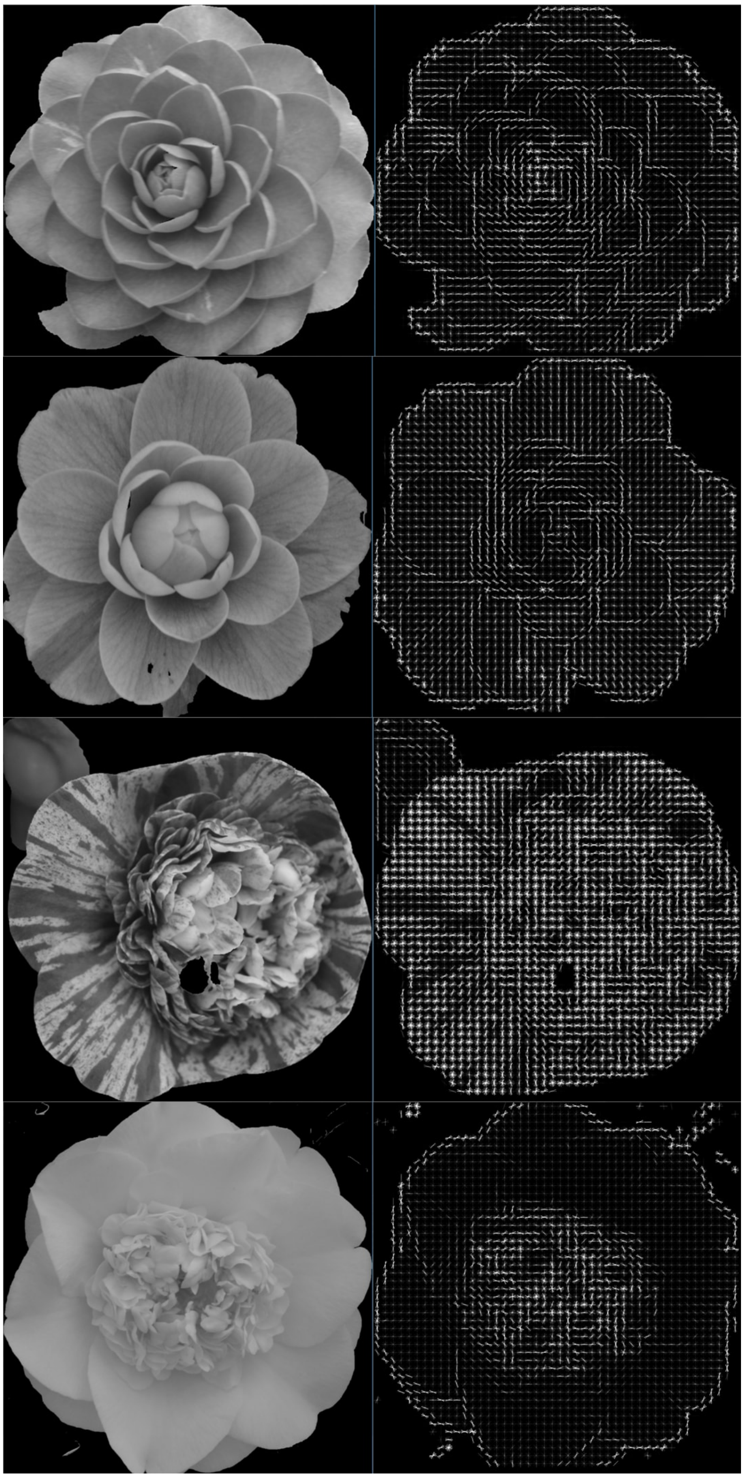


Figure 3.18: On the left, original images after grayscaling. On the right, obtained results of the HOG algorithm.

3.3.3 Color features

Another important aspect of the dataset is the color variation within species. Some are very distinct while others are very similar. Considering the narrow interval of possible colors for each species a simple color feature extraction was implemented. Looking at the histogram of the Hue channel of the HSV color space of the segmented image a spike surges on the most present color or colors, Figure 3.19.

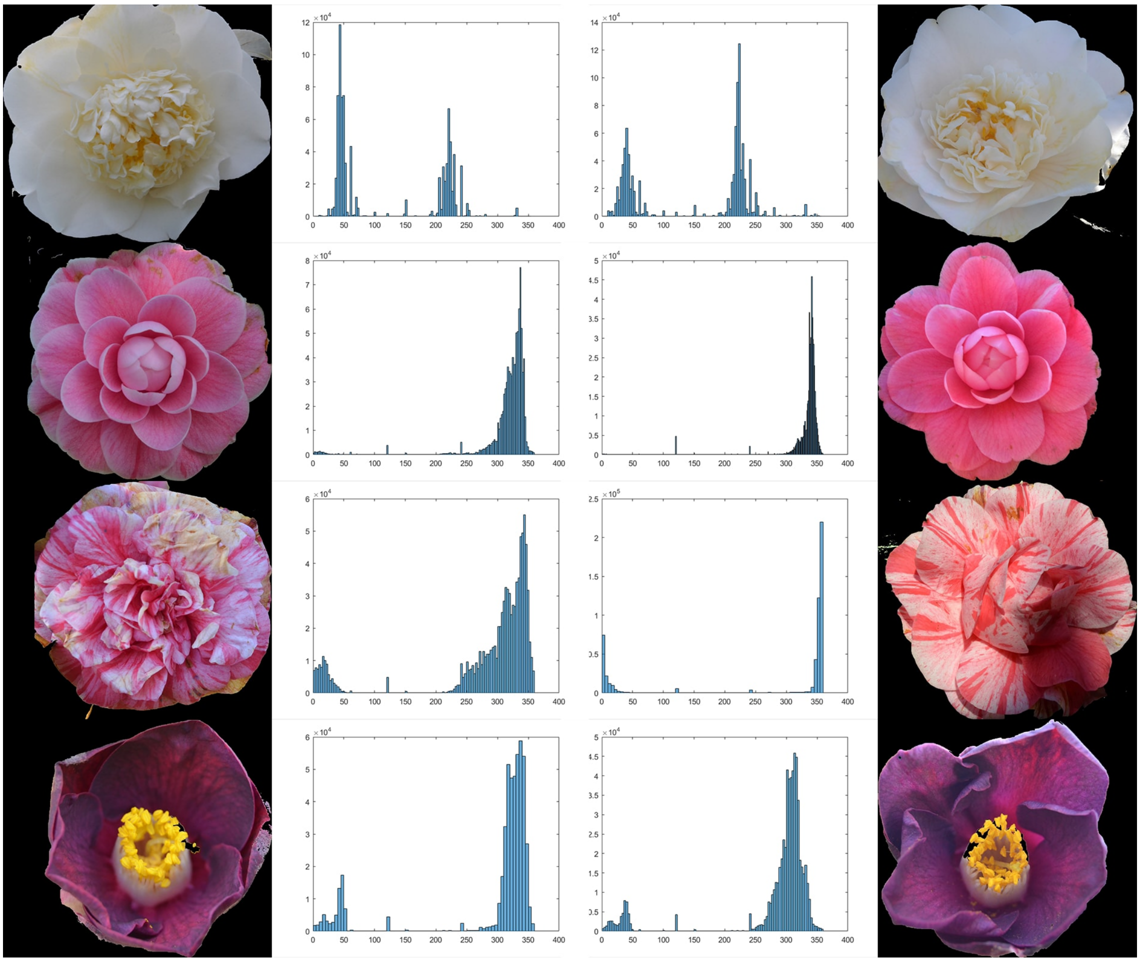


Figure 3.19: Two examples of Hue channel histogram of four different camellia species.

Since the segmented flower image has a black background with just the flower on the foreground this method creates a color signature for each species. If the flower has one main color, only one spike will appear; if the flower has two main colors, two spikes will appear, and so on.

3.4 Classifier and detector

After collecting the features that describe the desired object it is needed to compare them to another image. With the features collected, the classifier will validate the uniqueness of them.

3.4.1 Brute force

The first method tested was the brute-force[34] method which accepts key points and tries to find the exact same key points in the new image. In the end, it returns the corresponding features of the object, Figure 3.20.

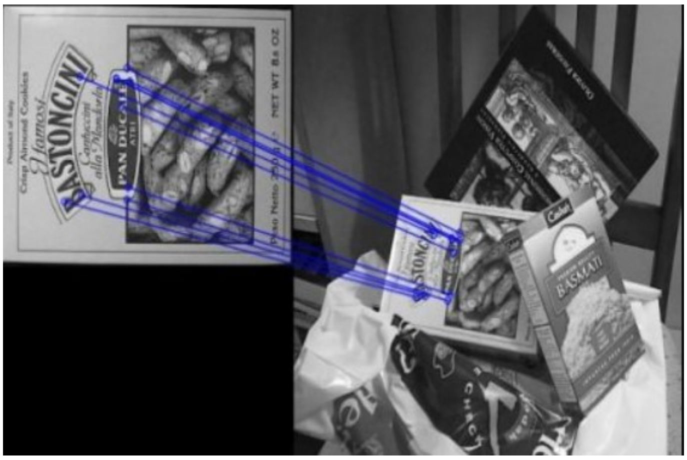


Figure 3.20: brute-force matching example. From [7].

This algorithm needs a template image for the comparison process, which is used to find the object in all the other images.

3.4.2 SVM

As studied in the Chapter 2 2.3.1, one of the most used and with better performance classifier is the SVM. This algorithm receives as input a group of training feature vectors of each image with a corresponding label. Then, giving the HOG and color feature vectors as parameters for training, it creates a model which separates the feature vectors with the provided labels.

Since the SVM classifier creates a separation among classes to be detected, several tests were performed with an increasing number of shapes to differ from. This test was made to observe what would happen to the accuracy of the created model.

All features extracted in the last step were tested for comparison reasons. They were used alone or joined together with other features.

3.4.3 YOLO

Another approach for shape recognition was to train an already existent object detector algorithm to distinguish among different shapes. Using the same segmented front flower dataset a detector model was created using YOLO, explained in section 2.3.6. A set of images was used as training data for the model, the training process took 6700 iterations and a total of 12 hours using 1GB of the Graphics Processing Unit (GPU). The training process was stopped when the error stagnated. After, different datasets were tested with this model for comparison reasons.

3.5 Final Considerations

In this chapter, the workflow of this work was explained and described. From the data acquisition to the classification, several methods were implemented in every step so that a more general approach could be done. In the next chapter, the results of the previously described methods will be shown and analyzed to understand the usability of each developed algorithm.

Chapter 4

Results and Discussion

In this chapter, the obtained results will be discussed and analyzed for every method used in each workflow phase. In each step, several approaches were taken to achieve the goal. When the results weren't what was expected some changes needed to be made or even new methods needed to be tested. All those outcomes will be discussed as well.

4.1 Flower segmentation

In section 3.2 several segmentation methods were described: The first uses a mask for segmentation - method A - the second uses different color spaces for autonomous segmentation - method B - while the last is user-aided - method C. In the next subsections the obtained results for each segmentation method will be presented and analyzed.

4.1.1 Autonomous Segmentation

- Method A

This method uses a hue interval [165^o,15^o] to separate the flowers from the background. Figure 4.1 shows some of the obtained results using this approach.

In the first row of Figure 4.1, the image is well segmented, since the flower color fits the defined interval.

In the second row, the segmented region is not clear, since the flower has colors (bright pink) that do not correspond to the defined interval for the hue channel values.

In the third row, the masked region isn't correct; once again the interval does not include the necessary colors present in these species (bright pink).

In the fourth row, the flower isn't segmented because it does not work for white flowers. Also, the contour given by this method has some errors when there are highlights in the flower, the results were not satisfactory to segment the whole dataset since it only works in very specific cases: reddish flowers.

Figure 4.1: Four result samples of the flower segmentation using method A. On the left, original image. On the right, obtained result.

- Method B

Figure 4.2, show the white flower segmentation results using the flower segmentation method B.1.

Figure 4.2: Two result samples of the flower segmentation using method B.1. On the left, original image. On the right, obtained result.

This method worked for all frontal white flowers acquired. Normally the background is green, so it has high contrast with the white flower, making this a very accurate approach.

In Figure 4.3 the results of the flower segmentation method B.2 are shown. The following 3 original images are the same used in method A for a better result comparison.

Figure 4.3: Three result samples of the flower segmentation using method B.2. On the left, original image. On the right, obtained result.

In the first row of Figure 4.3, as in method A the segmentation of camellia flowers with this color range is clear and smooth.

However, in the second and third examples, one can notice that with method B.1 the camellia flowers are well separated from the background even if they have stripes with different colors or colors different than red.

In the image on the right of the third row of Figure 4.3 there is a blob caused by other flowers of the same species in the background. This was expected since it separates everything that has the same color as the flower.

With the implemented segmentation method B the contours were more precise and smoother when compared with method A. Merging methods B.1 and B.2 it was possible to segment the frontal flower dataset cropping the image based on the flower contour, Figure 4.4.

Figure 4.4: Four results of the segmentation method B in flower with different colors.

4.1.2 User Segmentation

- Method C

To identify a plant, a user needs to input an image. Taking this into account having some input from the user to correctly segment the image is desirable. A method with feedback was implemented. The flower's region is selected by the user as well as the acceptance of the segmentation, Figure 4.5.

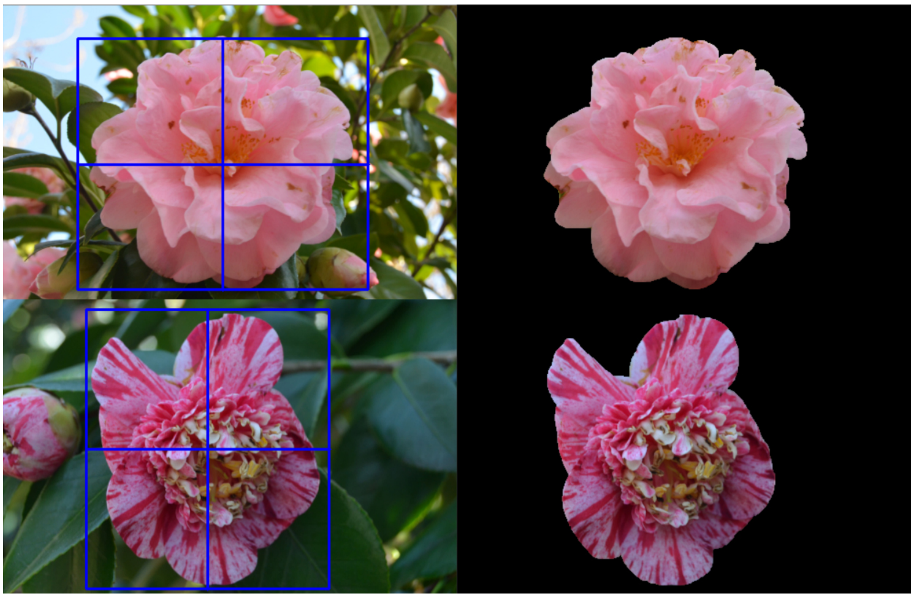


Figure 4.5: Two result samples of the flower segmentation using method C. On the left, original image. On the right, obtained result.

This approach gave good results and the fact that the segmentation can be corrected is a plus for error avoidance of the segmentation. If an error is found by the visual inspection of a user in the segmented image a new selection can be made, Figure 4.6.

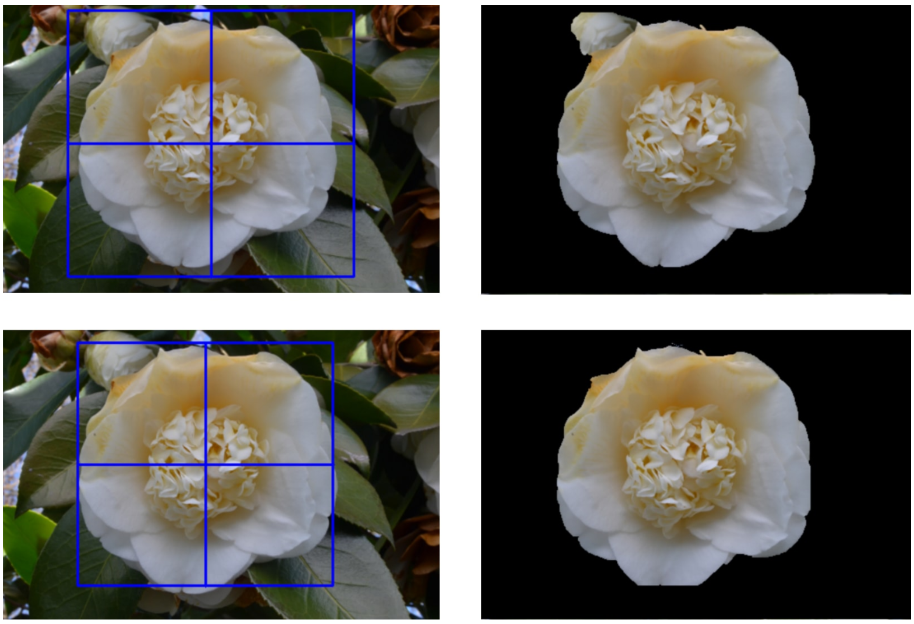


Figure 4.6: On the top left, the user selects the region of the flower. On the top right, the result of segmentation method C is shown. Since the biggest flower present in the image is not well segmented, the user can chose to select another region. On the bottom left, the user selects the new region. On the bottom right, the flower is well segmented so the user informs the algorithm to keep this one.

4.2 Classifier Results

With the extracted features of the segmented frontal flower dataset, the following results were obtained.

4.2.1 SIFT with Brute Force Matching

One typical result of matching SIFT key points with a template image to find the corresponding species is shown in Figures 4.7.

Figure 4.7: Brute-force matching with a red flower.

This method did not perform correctly in any tested image because it tries to find the exact same object in the new image. Since every flower is different from each other, even in the same species or camellia's shape type, it could not be correctly identified with the implemented Brute Force matching method.

4.2.2 HOG shape features with SVM Classifier

Having the HOG features of each image with the corresponding labels, a training dataset consisting of the segmented frontal flowers was fed to an SVM classifier. The results shown below are in form of a graph that represents the accuracy of the algorithm to predict the test images. The most relevant result is the shape accuracy since it represents a more general group than the species accuracy which just represents a small percentage of all existent camellia species, as explained in the previous chapter.

The SVM classifier creates a separation line between different classes. To understand how accurate this separation is, tests were made with an increasing number of classes. Firstly, this method was tested with just 2 different shapes (4 different camellia species), Figure 4.8. In this

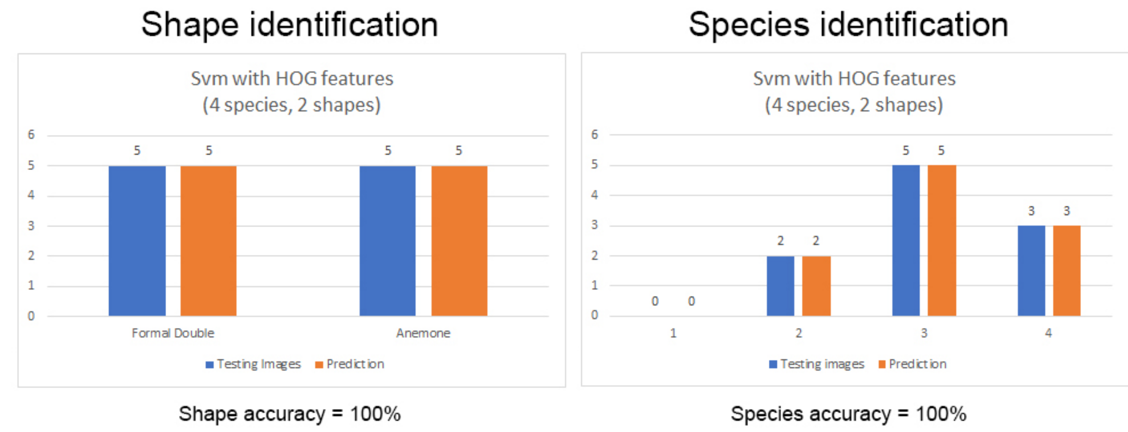


Figure 4.8: SVM results with HOG features of 2 shapes (4 camellia species). On the left the predictions of the shape are compared with the test images. On the right the predictions of the species are compared with the test images

case, the accuracy was 100%, which means that the separation line is very clear between both shapes, as well as species.

Next, one more shape was added to the classification test (and consequently, two more species). The results are shown below in Figure 4.9. With an accuracy of 92.9% the separation among shapes and species is still clear, this time with some errors. Adding 1 more shape (2 more species),

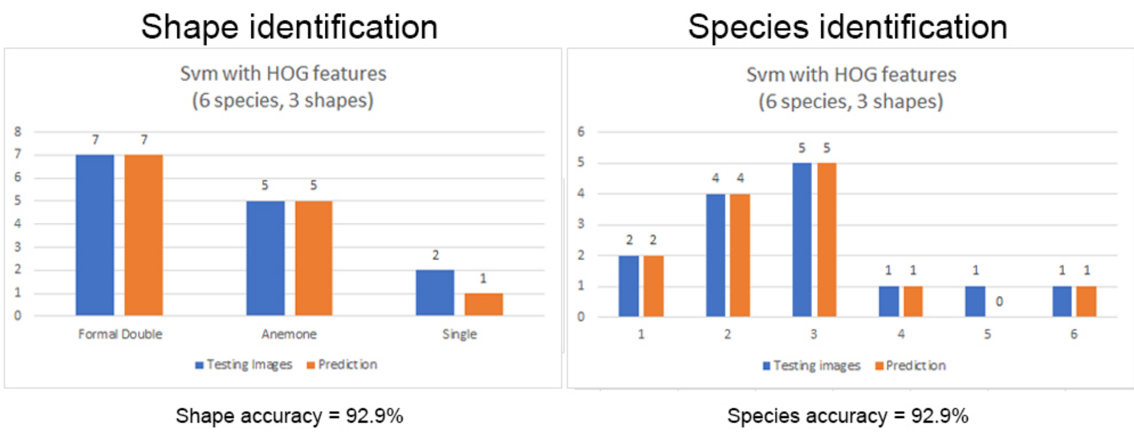


Figure 4.9: SVM results with HOG features of 3 shapes (6 camellia species). On the left the predictions of the shape are compared with the test images. On the right the predictions of the species are compared with the test images

the outcome is shown in Figure 4.10. As expected, the more classes to distinguish from, the worst the accuracy gets. It starts to be difficult to create lines that separate the used classes.

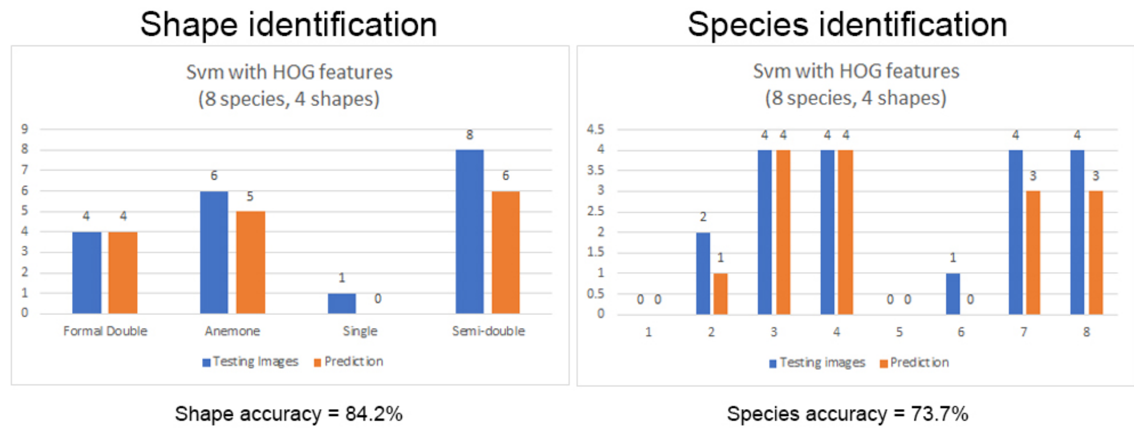


Figure 4.10: SVM results with HOG features of 4 shapes (8 camellia species). On the left the predictions of the shape are compared with the test images. On the right the predictions of the species are compared with the test images

For the last test, the remaining 2 shapes were added (3 species), Figure 4.11. Previously, the results were satisfactory when the number of species was low. However, when the number of species increased the accuracy dropped. This is because the features used to define the species' classes were not relevant enough to distinguish among all.

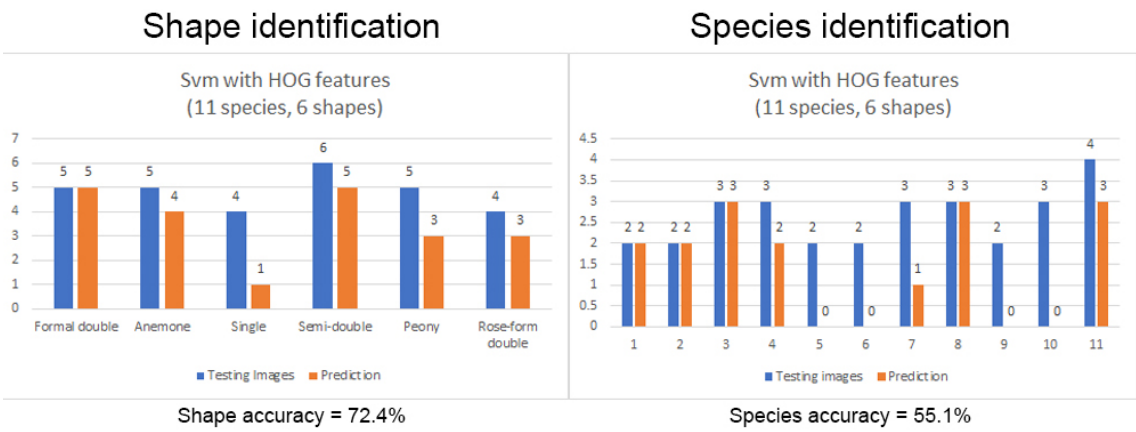


Figure 4.11: SVM results with HOG features of 6 shapes (11 camellia species). On the left the predictions of the shape are compared with the test images. On the right the predictions of the species are compared with the test images

4.2.3 Histogram color features with SVM Classifier

The next test combination was to use color features in the SVM classifier to understand how important the color was in the recognition process.

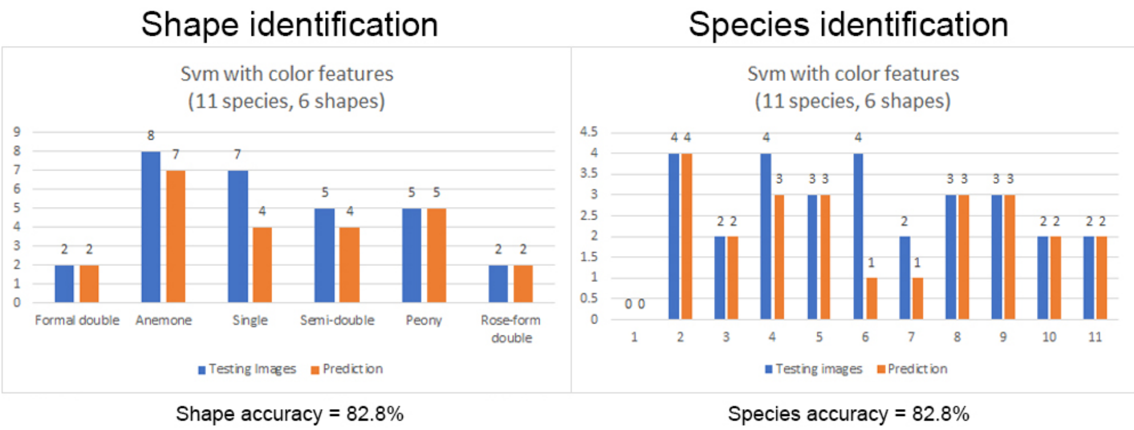


Figure 4.12: SVM results with histogram features of 6 shapes (11 camellia species). On the left the predictions of the shape are compared with the test images. On the right the predictions of the species are compared with the test images.

As it can be seen in Figure 4.12, it had great performance with all the species since they all differ in color even if just slightly.

4.2.4 HOG shape features and Histogram color features with SVM Classifier

The last combination test was to use both color and shape features to see if the accuracy would rise. Results, for all shapes and camellia species, can be found in Figure 4.13.

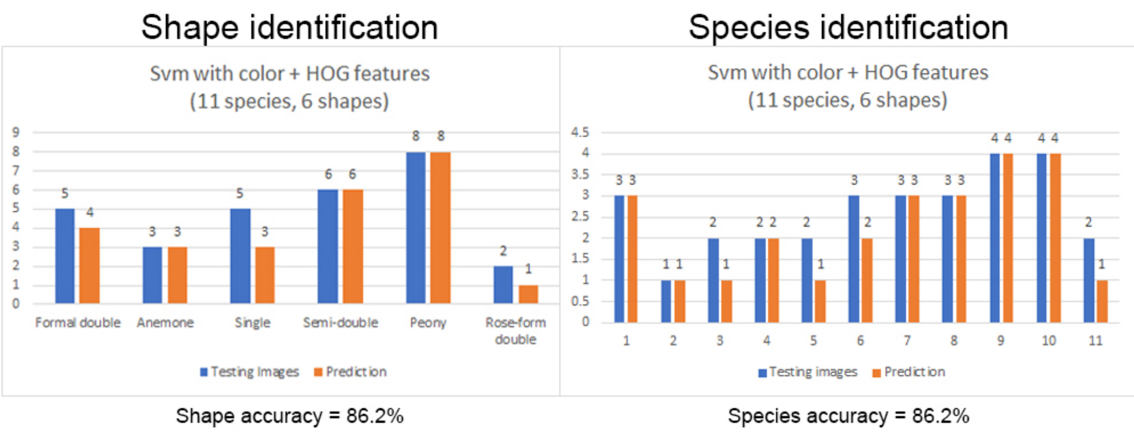


Figure 4.13: SVM results with histogram + HOG features of 6 shapes (11 camellia species). On the left the predictions of the shape are compared with the test images. On the right the predictions of the species are compared with the test images

Comparing with using the only color features, the performance increased, not much but the results were satisfactory for the existing dataset.

4.2.5 SVM results summary

In Table 4.1, all the previously obtained accuracy values with the SVM classifier are shown for comparison.

Table 4.1: Summary of the results obtained using SVM as classifier.

Nr of shape	Shape Accuracy	Species Accuracy
SVM results with HOG features		
2 shapes	100%	100%
3 shapes	92.9%	92.9%
4 shapes	84.2%	73.7%
all shapes	72.4%	55.1%
SVM results with color features		
all shapes	82.8%	82.8%
SVM results with HOG features + color features		
all shapes	86.2%	86.2%

Observing this table it is clear to see a decrease in performance with the addition of new classes using the same features. However, when other relevant features are added the results get better.

The results were satisfactory since the shape recognition is very high. With this, it was possible to identify the shape of most of the camellia species' flower with a good accuracy, using an image of its flower.

4.3 YOLO

With the trained model of the segmented dataset, some different tests have been done. The first was just with frontal flower images already segmented, as shown in Figure 4.14.

Figure 4.14: Segmented images from the frontal flower dataset classified with the YOLO trained model.

This test had good results getting 100% accuracy with the test dataset of the segmented frontal flower dataset. Only one image got two possibilities of classification; however, the one with the highest percentage was the correct one, Figure 4.15. Considering that there was only one flower in the image, the class with higher probability was taken as the prediction.

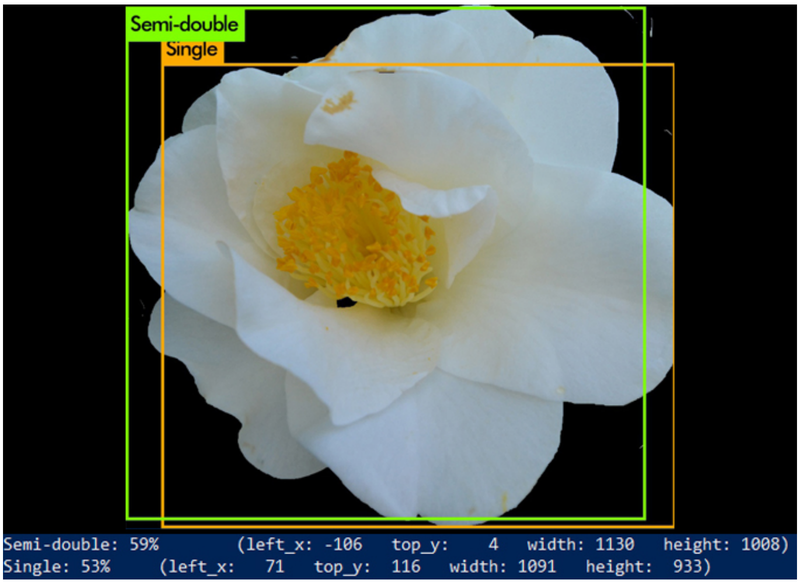


Figure 4.15: Only case where two classes were attributed. However, the percentage of the correct shape is the biggest.

The next test was using the same trained model with images that contained the frontal flower but without a segmented background. This test intended to understand how viable it was to use a trained model with segmented flowers to identify flowers in original images without segmenting them, Figure 4.16.

Figure 4.16: Original images classified with the YOLO trained model.

With an accuracy of 91.9% for all shapes on the unsegmented frontal flower dataset, the results were worse than with the segmented images but still with good performance. The test images are taken with a DSLR and mobile phone which makes this model a pretty good shape detector in a real environment. Nevertheless, some errors occurred, such as misclassification or no shape found, Figure 4.17. The model can either incorrectly identify the shape or not recognize it at all. In both cases, the images had an incorrect exposure as well as several highlighting differences. This means that to be correctly identified, the photo to be classified needs to be correctly exposed.

Figure 4.17: On the left, the shape prediction is incorrect. On the right, the model does not recognize any shape.

The next step was to test the same model with not frontal flowers dataset. The accuracy, in this case, dropped to 65.9% which still was pretty good knowing that the model was trained with segmented flowers images facing the lens. Some correct results are shown in Figure 4.18, while incorrect results are shown in Figure 4.19.

Figure 4.18: YOLO model correct predictions, with flowers that were not facing the lens.

Figure 4.19: YOLO model incorrect predictions, with flowers that were not facing the lens.

The last test used the model to predict the shape when there are multiple flowers in the same image, Figure 4.20. In this case, the accuracy was pretty low, only reaching 25.0%.

Figure 4.20: Shapes of the multiple flowers dataset classified with yolo trained model.

The summary of the results are shown in Table 4.2.

Table 4.2: YOLO model results with different test images.

YOLO results	
Dataset	Shape accuracy
Flower front segmented	100%
Flower front original	91.9%
Flower multiple angles	65.9%
Multiple flowers per image	25.0%

To sum up, the trained model had great accuracy when the test images were segmented. In this case, the implemented segmentation methods could be applied prior to the identification phase. This had great results and was a viable option to use as an identification module. In cases where the flowers are not segmented it could still recognize their shape with good accuracy, as long as the flower was facing the lens. If the user could not take a clear picture of the flower it was always possible to take a tilted image, knowing that the performance in the case was lower. However, the big constraint was, there could not be more than 1 flower in the image since this model did not work well in this situation.

4.4 Final Considerations

In this chapter, the obtained results of each phase are shown and discussed. They are compared with other used methods and the performance of each algorithm is evaluated. In the next chapter, conclusions will be discussed of all the developed work and outcomes. The possible future work is also considered.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

Plant recognition is a difficult task for most people, but also for experts and enthusiasts, it can take some research time to differ among similar species. For this reason, this work focuses on identifying autonomously the shape of every camellia plant species, based on its flower.

The first step was to collect and organize a relevant dataset to work on containing several flower and leaf images of different camellia species. After, it was decided to segment the dataset to extract key features that could describe the image with the best accuracy possible. Two flower segmentation methods with good results were created based on CV techniques. One autonomous to segment a big amount of images at once, and one that needs user assistance. The latter was created because this work aims to identify one picture at a time to permit the user to access the data as soon as possible. Another advantage of this user-aided method is the human feedback on how good the segmentation is, which helps to avoid classification errors from a bad segmentation.

With a correctly segmented dataset, labels were attached so that classifiers could be applied. Different features were extracted from the segmented images. These features were then tested to see which ones work best on the identification process. The best results were obtained using a YOLO trained model that identifies the frontal flower segmented dataset with 100% accuracy. The same model was tested with unprocessed frontal flower images and the results dropped to 91.1%. Good results were also obtained using HOG and color features with SVM classifier, 86.2% for shape accuracy. This result is comparable to the YOLO model that had 100% accuracy since the same segmented dataset was used for training and testing in both tests.

Shape classification was chosen instead of species because the acquired dataset did not contain a relevant amount of different species. However, it contains enough data to detect a very relevant step in the camellia species identification, its shape.

In conclusion, the defined objectives were achieved with good results for the main group of images. The classification process could be enhanced in terms of possible images to identify, after all, for the implemented algorithms to have better accuracy, the acquired image must be in accordance with two constraints. Them being, the flower needs to be facing the camera and there can only be one flower present in each image. These can make the acquisition process more difficult for the user, however, the probability of being correctly identified is substantially increased.

5.2 Work to be done

Despite the satisfactory results there still exists space for improvements. The main difficulty for the development of this work was the inexistence of a camellia species dataset, this meant that a new dataset needed to be created.

Since this is a time-bounded project it was impossible to collect images from the majority of the camellia species' plants, since each species blossom in a specific time of the year. Another constraint was the big amount of existent species, a team of contributors from different parts of the globe would be needed to collect enough images, also some species are only found in specific regions. Furthermore, the collected species needed to be validated by a specialist to prevent labeling errors and provide a rich and accurate dataset for camellia species.

With a complete labeled dataset, the best methods could be applied to recognize each species. The developed workflow could be used to split the species into different sub-datasets because the shape is already recognized. Having the images separated by their shape a new model could be trained for specific species of the same shape.

Finally, if this module could correctly identify the camellia species it could be implemented in a mobile application for this matter. Likewise, it could be used as a submodule for a more general plant identification application that identifies the plant as camellia. The same methods could be tested in other flowers than camellias to validate this approach in several other plants.

References

[1] Jana Wäldchen and Patrick Mäder. *Plant Species Identification Using Computer Vision Techniques: A Systematic Literature Review*, volume 0. Springer Netherlands, 2017. doi:10.1007/s11831-016-9206-z.

[2] Zhaobin Wang, Xiaoguang Sun, Yide Ma, Hongjuan Zhang, Yurun Ma, Weiying Xie, and Yaonan Zhang. Plant recognition based on intersecting cortical model. *Proceedings of the International Joint Conference on Neural Networks*, (2013):975–980, 2014. doi:10.1109/IJCNN.2014.6889656.

[3] Xuan Wang, Junhua Liang, and Fangxia Guo. Feature extraction algorithm based on dual-scale decomposition and local binary descriptors for plant leaf recognition. *Digital Signal Processing: A Review Journal*, 34:101–107, 2014. URL: http://dx.doi.org/10.1016/j.dsp.2014.08.005, doi:10.1016/j.dsp.2014.08.005.

[4] 92071fa83889a5294ba352cccd89a0431298a32e @ pjreddie.com, 2018. URL: https://pjreddie.com/darknet/yolo/.

[5] index @ eekwi.org, 2018. URL: http://eekwi.org/veg/treekey/.

[6] Maria Celina Veiga Ferreira. *O Mundo da Camélia*. V. Ferreira, 2000.

[7] 92071fa83889a5294ba352cccd89a0431298a32e @ pjreddie.com, 2018. URL: https://pjreddie.com/darknet/yolo/.

[8] J Wäldchen and P Mäder. Plant Species Identification Using Computer Vision Techniques: A Systematic Literature Review. *Archives of Computational Methods in Engineering*, pages 1–37, 2017. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85008498761{&}doi=10.1007{%}2Fs11831-016-9206-z{&}partnerID=40{&}md5=154d64147f701729f955b3ac19237942, doi:10.1007/s11831-016-9206-z.

[9] Alexis Joly, Hervé Goëau, Hervé Glotin, Concetto Spampinato, Pierre Bonnet, Willem Pier Vellinga, Jean Christophe Lombardo, Robert Planqué, Simone Palazzo, and Henning Müller. LifeCLEF 2017 lab overview: Multimedia Species identification challenges. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10456 LNCS(i):255–274, 2017. doi:10.1007/978-3-319-65813-1_24.

[10] index @ identify.plantnet-project.org, 2018. URL: https://identify.plantnet-project.org/.

[11] index @ plantix.net, 2018. URL: https://plantix.net/.

[12] index @ www.picturethisai.com, 2018. URL: https://www.picturethisai.com/.

[13] Mostafa Mehdipour Ghazi, Berrin Yanikoglu, and Erchan Aptoula. Plant identification using deep neural networks via optimization of transfer learning parameters. *Neurocomputing*, 235:228–235, 2017. URL: http://dx.doi.org/10.1016/j.neucom.2017.01.018, doi:10.1016/j.neucom.2017.01.018.

[14] Adams Begue, Venitha Kowlessur, Upasana Singh, Fawzi Mahomoodally, and Sameerchand Pudaruth. Automatic Recognition of Medicinal Plants using Machine Learning Techniques. *IJACSA) International Journal of Advanced Computer Science and Applications*, 8(4):166–175, 2017. URL: www.ijacsa.thesai.org.

[15] Jose Carranza-Rojas, Herve Goeau, Pierre Bonnet, Erick Mata-Montero, and Alexis Joly. Going deeper in the automated identification of Herbarium specimens. *BMC Evolutionary Biology*, 17(1):181, 2017. URL: http://bmcevolbiol.biomedcentral.com/articles/10.1186/s12862-017-1014-z, doi:10.1186/s12862-017-1014-z.

[16] J Mcneill, F R Barrie, W R Buck, V Demoulin, W Greuter, D L Hawksworth, P S Herendeen, S Knapp, K Marhold, J Prado, W F Prud Homme Van Reine, G F Smith, J H Wiersema, and N J Turland. International Code of Nomenclature for algae, fungi, and plants (Melbourne Code). Regnum Vegetabile 154. *Koeltz Scientific Books*, pages 1–140, 2012. arXiv:3, doi:10.1111/j.1365-2699.2010.02341.x.

[17] T. Huang. Computer Vision: Evolution And Promise. *19th CERN School of Computing*, pages 21–25, 1996. doi:10.5170/CERN-1996-008.21.

[18] Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, 2002. arXiv:arXiv:1011.1669v3, doi:10.1109/34.993558.

[19] Cong Zhao, Sharon S.F. Chan, Wai Kuen Cham, and L. M. Chu. Plant identification using leaf shapes - A pattern counting approach. *Pattern Recognition*, 48(10):3203–3215, 2015. URL: http://dx.doi.org/10.1016/j.patcog.2015.04.004, doi:10.1016/j.patcog.2015.04.004.

[20] Timo Ojala, Matti Pietikäinen, and David Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 29(1):51–59, 1996. doi:10.1016/0031-3203(95)00067-4.

[21] Y Ma, R Dai, L Li, and L Wei. Image segmentation of embryonic plant cell using pulse-coupled neural networks . *Chinese Science Bulletin*, pages 169–173, 2002. doi:10.1360/02tb9040.

[22] A. L. Samuel. Some Studies in Machine Learning Using the Game of Checkers. *IBM Journal of Research and Development*, 3(3):210–229, 1959. URL: http://ieeexplore.ieee.org/document/5392560/, doi:10.1147/rd.33.0210.

[23] James M. Keller, Michael R. Gray, and James A. Givens. A fuzzy K-nearest neighbor algorithm. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-15(4):580–585, 1985. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6313426, doi:10.1109/TSMC.1985.6313426.

[24] Yasha Zeinali and Brett A. Story. Competitive probabilistic neural network. *Integrated Computer-Aided Engineering*, 24(2):105–118, 2017. doi:10.3233/ICA-170540.

[25] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv*, 2018.

[26] John R Koza, Forrest H Bennett III, David Andre, and Martin A Keane. Automated design of both the topology and sizing of analog electrical circuits using genetic programming. In *Artificial Intelligence in Design '96*, pages 151–170, 1996.

[27] Camellia @ www.britannica.com, 2018. URL: https://www.britannica.com/plant/Camellia.

[28] index @ cameliasflavius.com, 2018. URL: http://cameliasflavius.com/.

[29] Maria Elena Nilsback and Andrew Zisserman. A visual vocabulary for flower classification. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2:1447–1454, 2006. doi:10.1109/CVPR.2006.42.

[30] Soon Won Hong and Lynn Choi. Automatic recognition of flowers through color and edge based contour detection. *2012 3rd International Conference on Image Processing Theory, Tools and Applications, IPTA 2012*, pages 141–146, 2012. doi:10.1109/IPTA.2012.6469535.

[31] Siu Yeung Cho and Peh Ti Lim. A novel virus infection clustering for flower images identification. *Proceedings - International Conference on Pattern Recognition*, 2:1038–1041, 2006. doi:10.1109/ICPR.2006.144.

[32] Nobuyuki Otsu. Threshold selection method from gray level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-9(1):62–66, 1979. URL: https://www.scopus.com/record/display.uri?eid=2-s2.0-0018306059{&}origin=inward{&}txGid=b0cd16e2e7bd0be890360eee58c9bfb3.

[33] py_grabcut @ docs.opencv.org, 2018. URL: https://docs.opencv.org/3.0-beta/doc/py{_}tutorials/py{_}imgproc/py{_}grabcut/py{_}grabcut.html.

[34] complexity-of-brute-force-search-5R6q3 @ www.coursera.org, 2018. URL: https://www.coursera.org/lecture/ml-clustering-and-retrieval/complexity-of-brute-force-search-5R6q3.