Technical University of Crete

School Of Electrical & Computer Engineering

DIPLOMA THESIS

Detection Of Visual Grape Leaf Characteristics For Automated Ampelography Type

Committee:

Prof. Michalis Zervakis (Supervisor)

Prof. Aggelos Bletsas

Theodora Pitsoli, Hellenic Agricultural Organisation "DEMETER"

Author:

Aikaterini Tsellou

July 24, 2020

Abstract

Ampelography is the branch of viticulture that studies the description, distinction, classification and evaluation of grapevine varieties. In the modern era of varietal wines, correct identification of different grapevine varieties is necessary as it can have a substantial financial impact on the wine industry. The development in digital photography and image processing tools offers enhanced capabilities for ampelography by providing automated and more accurate methods to discriminate leaves, replacing the classic technique. In this thesis, we prove that machine learning algorithms are able to classify efficiently different kinds of grape leaves in an automated way. The proposed approach consists of the following phases: segmentation, feature extraction, feature selection and classification. In the segmentation phase the leaf is separated from its background. Then, in the feature extraction phase, the segmented leaf image is analyzed in order to extract shape and contour features. After extracting the features, we select an optimal subset of them in order to perform the classification in the next phase. Finally, the results are classified using 3 different algorithms: Naïve Bayes, Decision Tree, SVM with linear kernel and quadratic kernel. Evaluating the classification results, it should be noted that the automatic extraction of morphological data and machine learning modelling proved to be rapid and accurate methods for cultivar classification.

Περίληψη

Η Αμπελογραφία είναι ο κλάδος της αμπελουργίας, που έχει ως αντικείμενο την περιγραφή, διάκριση, ταξινόμηση και αξιολόγηση των ποικιλιών της αμπέλου. Στη σύγχρονη εποχή, είναι απαραίτητη η σωστή αναγνώριση των διαφορετικών ποικιλιών αμπέλου, καθώς μπορεί να έχει σημαντική οικονομική επίπτωση στην οινοβιομηχανία. Η ανάπτυξη των εργαλείων ψηφιακής φωτογραφίας και επεξεργασίας εικόνας προσφέρει βελτιωμένες δυνατότητες για αμπελογραφική αναγνώριση παρέχοντας αυτοματοποιημένες και πιο ακριβείς μεθόδους για τη διάκριση των φύλλων, αντικαθιστώντας τις κλασσικές μεθόδους. Σε αυτή τη διατριβή, αποδεικνύουμε ότι οι αλγόριθμοι μηχανικής μάθησης είναι σε θέση να ταξινομήσουν αποτελεσματικά διαφορετικά είδη φύλλων της αμπέλου με αυτοματοποιημένο τρόπο. Η προτεινόμενη προσέγγιση αποτελείται από τις ακόλουθες φάσεις: τμηματοποίηση, εξαγωγή χαρακτηριστικών, επιλογή χαρακτηριστικών και ταξινόμηση. Στη φάση της τμηματοποίησης το φύλλο διαχωρίζεται από την υπόλοιπη εικόνα. Στη συνέχεια, στη φάση εξαγωγής χαρακτηριστικών, το φύλλο, αφού έχει περάσει απο τη διαδικασία της τμηματοποίησης, αναλύεται με σκοπό την εξαγωγή χαρακτηριστικών που αφορούν το σχήμα καθώς και το περίγραμμα. Μετά την εξαγωγή των χαρακτηριστικών, επιλέγουμε ένα βέλτιστο υποσύνολο αυτών για να πραγματοποιήσουμε την ταξινόμηση στην επόμενη φάση. Τέλος, τα αποτελέσματα ταξινομούνται χρησιμοποιώντας 3 διαφορετικούς αλγορίθμους: Naïve Bayes, Decision Tree, SVM με 2 είδη πυρήνα, γραμμικό και τετραγωνικό. Αξιολογώντας τα αποτελέσματα της ταξινόμησης, θα πρέπει να σημειωθεί ότι η αυτόματη εξαγωγή μορφολογικών δεδομένων και η μοντελοποίηση με τη βοήθεια της μηχανικής μάθησης αποτελούν τόσο γρήγορες όσο και ακριβείς μεθόδους για την ταξινόμηση των ποικιλιών.

Acknowledgements

This thesis becomes a reality with the kind support of many individuals. I would like to thank all of them.

First, I would like to thank Professor Michalis Zervakis for the chance to work on this thesis and for his guidance at every step along in the road.

My sincere thanks also goes to Mrs. Konstantia Moirogiorgou for guiding me and providing consultation.

I would also like to thank the rest of my thesis committee: Prof. Aggelos Bletsas and Mrs. Theodora Pitsoli.

My special gratitude goes to Hellenic Agricultural Organisation ”DEMETER” for providing information as well as the data needed for this thesis.

And finally, last but by no means least, I would like to thank my family for supporting me all these years and my friends for always believing in me.

Contents

1 Introduction **19**

1.1 Problem Statement 19

1.2 Previous & Related Work 20

1.2.1 Grape Vine Leaves 20

1.2.2 Leaf Pattern Recognition 21

1.3 Purpose, Objectives & Innovation 22

1.3.1 Purpose 22

1.3.2 Objectives 22

1.3.3 Innovation 22

1.4 Thesis Outline 23

2 Background **25**

2.1 Ampelographic Background 25

2.1.1 The concept of Ampelography 25

2.1.2 Classification systems 25

2.1.3 Ampelographic description 27

2.2 Technical Background 29

2.2.1 Segmentation 29

2.2.2 Supervised & Unsupervised Learning 31

2.2.3 Classification 32

2.2.4 Feature Selection 33

3 Proposed Methodology **35**

3.1 Segmentation 35

3.1.1 Visible Spectral Indexes 35

3.2 Feature Extraction . . . 37
3.2.1 Shape Features . . . 37
3.2.2 Vein Extraction . . . 41
3.3 Feature Selection . . . 42
3.3.1 Fisher's Score . . . 43
3.3.2 Support Vector Machine - Recursive Feature Elimination . . . 45
3.4 Classification . . . 46
3.4.1 Naïve Bayes Classifier . . . 46
3.4.2 Decision Tree Classifier . . . 47
3.4.3 Support Vector Machine (SVM) . . . 48
3.5 Evaluation . . . 50

4 Results & Discussion **53**
4.1 Experiment Dataset . . . 53
4.2 Segmentation Results . . . 54
4.3 Feature Extraction Results . . . 55
4.4 Feature Selection Results . . . 56
4.4.1 Fisher's Score . . . 56
4.4.2 Support Vector Machine - Recursive Feature Elimination . . . 57
4.5 Classification Results . . . 58
4.5.1 Experiment 1 . . . 59
4.5.2 Experiment 2 . . . 60
4.5.3 Experiment 3 . . . 61
4.6 Discussion . . . 62
4.6.1 Evaluation of the extracted feature set . . . 62
4.6.2 Comparison of Fisher's Score and SVM-RFE . . . 62
4.6.3 Comparison of classifiers and their capabilities . . . 63

5 Conclusion & Future Work **65**

Appendices **67**

A Supplementary material **69**
A.1 Segmentation results . . . 69

Bibliography **71**

List of Figures

1.1 Leaf Recognition Steps 21

2.1 Growing Tips 27
2.2 Mature Leaf Parts 28
2.3 Different leaf blade shapes 28
2.4 Edge Based Segmentation Results 29
2.5 Otsu's Threshold Method Results 30
2.6 K-Means Segmentation 31
2.7 Feature Selection Methods 33

3.1 Flow Diagram of Segmentation Process 36
3.2 Leaf's Perimeter & Bounding Box Visualization 37
3.3 Leaves and their Convex Hulls 39
3.4 Contour points and Centroid of the leaf 40
3.5 Dispersion Feature 41
3.6 Vein Extraction Steps 42
3.7 Self-Organizing Map Scheme 43
3.8 Decision Tree Form 47

4.1 Sample images from our dataset 53
4.2 Combination of EG & ER binarized images. 54
4.3 Segmented leaf samples along with their binary images 54
4.4 Indicative Feature Extraction Results 55
4.5 Sample Clusters 56
4.6 LDA Visualization: Sample Clusters 57
4.7 Classification Results: 15 Features 59
4.8 Classification Results: 13 Features 60

4.9 Classification Results: 10 Features 61

List of Tables

3.1 Confusion Matrix 51

4.1 Fisher's Discriminant Ratio 56
4.2 Feature's Occurring Rate 58
4.3 SVM-RFE Ranking 58
4.4 Evaluation: 15 Features 59
4.5 Evaluation: 13 Features 60
4.6 Evaluation: 10 Features 61

List of Abbreviations

EG ..... Excess Green Index

ER ..... Excess Red Index

CCD ..... Contour-Centroid Distance

SOM ..... Self-Organizing Map

BMU ..... Best Matching Unit

FDR ..... Fisher's Discriminant Ratio

LDA ..... Linear Discriminant Analysis

ANN ..... Artificial Neural Network

SVM ..... Support Vector Machine

OVA ..... One-Versus-All

SVM-RFE ..... Support Vector Machine - Recursive Feature Elimination

TP ..... True Positive

FP ..... False Positive

TN ..... True Negative

FN ..... False Negative

Chapter 1

Introduction

1.1 Problem Statement

Ampelography -deriving from the Greek words “ampelos” meaning vine and “graphe” meaning description- is the field of botany that studies the identification and classification of grapevines. Specifically, it is the characterization of grapevine cultivars through visual inspection of the grapevine (Tassie, 2010). It is crucial to achieve correct identification for several reasons. First of all, it is essential for successful wine marketing. But despite that, mistakes in identification can lead to adulteration or fraud, which therefore can have a great financial impact on today’s wine-making industry (Adão et al., 2019).

The first attempts for ampelographic classification focused on identification of grapes and berries. Though this method is now considered uncertain and relatively error - prone. After World War II, the ”Father of Ampelography”, Pierre Galet introduced a system for identifying varieties based on the shape, contours and characteristics of the leaves of the vines, petioles, growing shoots, shoot tips, grape clusters, as well as the colour, size, seed content and flavour of the grapes. Mature leaves of grapevines contribute significantly in the identification process. It is possible to distinguish further among grapevine varieties if you pay careful attention to the distinct details that are present on vine’s mature leaves. It is notable that each grapevine variety’s leaves are unique as they possess characteristic shapes, coloring, size and other visual descriptors.

The past few years, there have been improvements in the identification and classification, using modern methods and software environments, that have replaced the traditional ampelographic approach. A drawback to the use of these modern methods though is that in

most cases they require expensive instruments, training and special skills to accurately identify grapevine cultivars. DNA testing is one of the most commonly used modern methods, providing high accuracy and reliability. Although accurate and reliable, DNA testing has a disadvantage as it requires sophisticated equipment. So, there is still a place for traditional ampelography in modern viticulture. The challenge is to update the classical ampelographic method to an automated process.

Image processing, the processing of digital images by means of a digital computer, has a crucial importance in automated systems, as it contributes to identification tasks. The development in digital photography and image processing tools allows us to obtain several visible mature leaf descriptors ,in an automated way, replacing the classical methods. These descriptors will determine the variety to which a particular leaf belongs to. The purpose of this thesis is to exploit image processing tools in order to capture the visible information of mature leaves and then use this information to perform automated ampelographic classification with the aid of machine learning algorithms.

1.2 Previous & Related Work

1.2.1 Grape Vine Leaves

The approach by Chitwood et al. used Elliptical Fourier Descriptors (EFDs) to quantify grape vine leaf shape, as well as Generalized Procrustes Analysis based on venation landmarks in order to analyse the venation patterning. Their data prove the existence of a genetic basis for the diversity present in grape leaves (Chitwood et al., 2014). However, their approach is only descriptive, showing different cultivars clustered and separated and they did not perform automated classification.

Fractal dimension has been used as a tool for grape vine leaf description as vine leaves possess a highly complex structure which makes them appropriate for characterization using fractal analysis. Stefano Mancuso uses the box counting method in his approach, proposing that vine leaves are fractal. According to him, it is important to define good shape measures and features so that leaf shapes can be compared and analysed by meaningful and objective criteria (Mancuso, 2001). He does not perform any classification either. His aim is to emphasize to the usefulness of fractal analysis in ampelography.

In a study that uses 16 different grapevine cultivars, the extraction of color and shape

parameters of the mature leaf is done automatically using computer vision algorithms and image analysis through a Matlab code (Fuentes et al., 2018). Near-infrared spectroscopy to obtain the chemical fingerprint of the leaves was also used in order to compare the accuracy of the two methods. The first method rendered an accuracy of 94%, while the second one (NIR) rendered an accuracy of 92%. In their approach, they use a dataset that consists of scanned leaves. They use both shape and colour features in their experiment. Although their approach presents fairly good results, there are some constraints regarding the equipment. Scanners may not be practical due to their portability. Even a small portable scanner can be a little bulky to carry around.

Another study, is based on the analysis of leaf images, taken with an RGB sensor (Marques et al., 2019). For each leaf 101 features were extracted, regarding both colour and shape. The approach was applied in 240 leaf images of three different grapevine varieties and it achieved an accuracy of 87%. Although efficient, in their approach they test only 3 different varieties. Moreover, they propose a high-dimensional feature space which is not always efficient regarding classification process.

1.2.2 Leaf Pattern Recognition

Nowadays, there is an increasing interest in the automated classification of different plant species based on their leaves, using machine learning algorithms. Leaf recognition process usually follows the steps shown in Fig. 1.1. There have been many approaches in leaf classification in order to develop automated systems that can help ordinary people to identify plants based on their leaves. The most common features used by researchers can be categorized into shape, contour, texture, vein and color features. Some researchers, combine features of the mentioned categories in order to develop an accurate method to classify different kind of plants (Kadir et al., 2013). Other researchers are based exclusively on one category. For example, Sathwik et al, used texture analysis in order to classify medicinal plant leaves (Sathwik et al., 2013). An approach that combines Fourier descriptors with other shape features was investigated to identify 100 kinds of leaves, achieving an accuracy rate of 88% (Kadir, 2015).

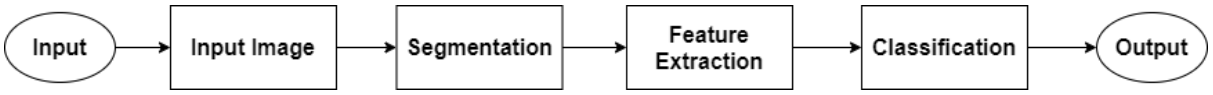


Figure 1.1: Leaf Recognition Steps

1.3 Purpose, Objectives & Innovation

1.3.1 Purpose

The main purpose of this thesis is to upgrade traditional ampelography for vine identification into a modern method, taking advantage of the capabilities provided due to development in image processing and machine learning fields. Using image processing tools, mature leaf, which is the region of interest, can be detected and segmented from its background in order to extract from it several mature leaf descriptors in an immediate, automated way. Then, machine learning algorithms are supplied with these descriptors and trained to recognize leaves of different grapevine cultivars.

1.3.2 Objectives

In this thesis, three main objectives are considered. The first one, is the proper preprocessing of the input grapevine leaf images. Specifically, before proceeding in any other step, the leaf must be separated efficiently from its background. To do this, a threshold based segmentation algorithm is implemented. The second objective, is the extraction of features that approach the descriptors used in traditional ampelographic identification. In traditional ampelography, descriptors regarding leaf's morphology are of high importance. Therefore, to capture this information, shape and contour features were extracted from the binary leaf image which is previously acquired from the segmentation process. Finally, the third objective concerns the reliability of the developed automated system for the ampelographic identification. In order to achieve reliability, satisfying evaluation metrics, such as high accuracy, have to be ensured from the side of the machine learning algorithms. Towards that end, different classification algorithms were evaluated. These include: Naive Bayes, Decision Trees and Support Vector Machines (SVM) using both quadratic and linear kernel. To further improve the accuracy of the evaluated models, feature selection algorithms were developed, for the retention of sufficient information and the rejection of redundant data and noise.

1.3.3 Innovation

The increasing use of innovative computer technology allows the use of digitalized methods for plant identification. In this thesis, image processing and machine learning algorithms

through a customized code written in Matlab were used in order to detect visual grape leaf descriptors and develop an automated system for vine identification. Digital images of grapevine leaves, acquired using a common digital camera, are used from the system for the identification. Until now, the ampelographic identification was performed in a manual way, which is time consuming. Furthermore, the modern methods, like DNA testing, require a complicated software environment. The processing of a digital photography using image processing tools can provide immediate results, unlike naked-eye observation of the grape vine leaf. Moreover, no specialized knowledge or skills will be required in order to use the system. It should be noted that the proposed method, in addition to being fast and efficient, can be also developed with no significant cost. Therefore, it is obvious that the proposed automated system is practical for routine grapevine cultivar classification.

1.4 Thesis Outline

The thesis is organized as follows:

1. In Chapter 2, background information concerning ampelography as well as the technical part of this work is introduced.
2. In Chapter 3, the methods used to approach the problem are explained.
    - First, a segmentation method is proposed, regarding the constraints of our dataset.
    - In the feature extraction phase, several morphology and contour features are proposed, as well as a method for vein extraction is introduced.
    - Next, two feature selection methods are analysed, in order to acquire an optimal feature subset.
    - Three classification models are described and will be tested. These are: Naive Bayes, Decision Tree, SVM with linear and SVM with quadratic kernel.
    - Finally, a description for the metrics that are used to evaluate the system's performance is provided.
3. In Chapter 4, the experimental results for the classification methods are presented along with the evaluation using the metrics described in Chapter 3.
4. The thesis concludes with Chapter 5, which suggests possible future work.

Chapter 2

Background

2.1 Ampelographic Background

2.1.1 The concept of Ampelography

Ampelography, a special branch of viticulture, has as main objectives the description, distinction and evaluation of cultivated grapevine varieties with the ultimate goal of their classification. In classic ampelography three methods are used in order to achieve the objectives pursued. The ampelographic description, the comparative ampelography and the experimental ampelography. The ampelographic description aims at distinguishing and classifying of the grapevine varieties and determining their identity, based on external characteristics of the organs. Comparative ampelography deals with the problem of synonyms of cultivated grapevine varieties in different places and with the research of the clonal composition of these populations. Experimental ampelography deals with the investigation and solution of problems related to the origin of varieties using methods of Genetics and Phytogeography as well as historical data.

2.1.2 Classification systems

Morphological classification

S. Helbling in 1777 classified the varieties into groups according to the color of the grapes and into subgroups based on the shape of the grapes (elongated-round). C. A. Frege in 1804 published a classification system based on grapes, while D. S. Poxas Clémente y Rubio (1807),

separated the varieties based on the density of hair on the leaves in combination with their cultivating properties. Christ, Acerbi (1825) and E. von Vest (1826) based the classification on grape and leaf characteristics while Di Rovasenta (1877) was based on the color and the taste of the grapes. V. Krimbas (1938) invented a system for classifying the varieties of wine grapes grown in Greece, based on the characters of the grape, the grape seed and the relations of these viticultural elements. In addition, he used leaf descriptors (Davidis, 1982). Rovasenda (1877) also proposed a classification system based on the density of hair on the leaves, the color and the density of hair on the tip as well as the color, shape and taste of the grapes. L. Levevoux (1946) used the morphological type of flower to distinguish varieties.

Ampelometric classification

Ampelometric classification is based on the measurement of the dimensions of grapes, leaves and angles formed by the veins, etc. This method was used by Metzger (1828), Goethe (1887) by measuring the angles of the main leaf veins, A. Rodrigez (1938), Ravaz (1902), Rodriquez (1952), Alleweldt et Dettweiller (1989) and others. The most well-known Ampelometric process is that of Galet (1979), in which the shape of the leaves is expressed by the proportions of the lengths of the lateral veins to the length of the main vein and by the sum of the angles formed by certain veins.

Phenological or physiological classification

This classification of varieties is based on the various phenological stages of the grapevine, such as the growth of the latent buds, flowering, maturation and leaf fall (Molon 1906). Pulliat (1888, 1897) classifies the grapevine varieties into early, first, second, third and fourth periods, adding sub-periods. Today, this classification corresponds to the early, mid-early, normal ripening period, mid-late and late varieties (Stavrakakis 2010).

Geographical classification

In the late 1920s, an attempt was made to classify the varieties of the European vine based on their geographical distribution (Vavilov 1926) and continued in the 1940s (Pirovano 1943, Negrul 1946). A notable effort is considered by Negrul, who was based on the morphological and biological characteristics of varieties found in different ecological systems and classified

them into three major groups: orientalis, which thrived in the Caucasus, pontica, in Black Sea region, and the occidentalis found in the westernmost parts of Europe (Unwin, 2003).

Phenotypic classification

Phenotypic classification is the division of varieties into groups of the same phenotype. It is based on the density of hair on the growing shoot tip, the herbaceous stem and the mature leaves, as well as on the measurements of the leaf characteristics for the determination of the leaf type (Stavrakakis 2010).

2.1.3 Ampelographic description

Modern ampelography has as its main objects the description, distinction and classification of the species and varieties of the grapevine, the study of all the factors that contribute to the phenotypic variation and their cultivation and finally, the economic evaluation in productive viticulture. For this purpose, ampelographic description has been developed. The study of the varieties is carried out with the method of ampelographic description according to the OIV list of descriptors (OIV 2009). The ampelographic characteristics examined refer to characteristics of the shoot tip, herbaceous stem, young and mature leaf, flower, grape, vines, phenological stages and leaf ampelometry.

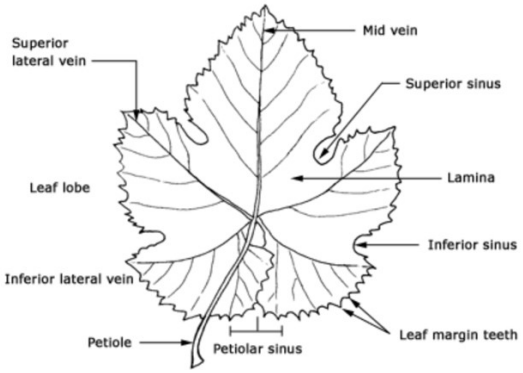
Shoot Tip Descriptors

Source: (Tassie, 2010)

Figure 2.1: Growing Tips

Many scientist claim that the first thing they look at is the shoot tip, if there is one. The most important descriptor that shoot tip provides is its anthocyanin coloration and density of hairs

on tip. Specifically, it can be fuzzy, hairless, shiny or covered in white cottony hair. One other useful descriptor provided by the shoot tip is its colour, as well as its shape. Fig. 2.1 illustrates shoot tips.

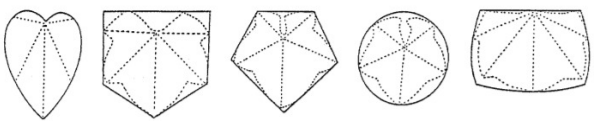


Source: (Tassie, 2010)

Figure 2.2: Mature Leaf Parts

Mature Leaf Descriptors

Another grape vine part that provides various descriptors and makes an important contribution in identification is the mature leaf. Each grapevine variety's leaves possess several distinct features that make them unique and therefore they can be used in order to distinguish among varieties. Some of the traditional ampelographic features include: the density of prostrate and erect hairs on / between the main veins on lower side of blade, the number of lobes, size of teeth, length of teeth, ratio length/width of teeth, shape of blade (Fig. 2.3), size of blade, color of the upper side of blade, goffering of blade, undulation of blade between main and lateral veins, blistering of upper side of blade, anthocyanin coloration of main veins on upper and lower side of blade, profile of blade in cross section, general shape of petiole sinus, degree of opening / overlapping of petiole sinus, tooth at petiole sinus, petiole sinus limited by veins, shape of upper lateral sinus, depth of upper lateral sinus, shape of base, length of petiole compared to middle vein (Ipgri, 1997). Grapevine mature leaf's parts are illustrated in Fig. 2.2.

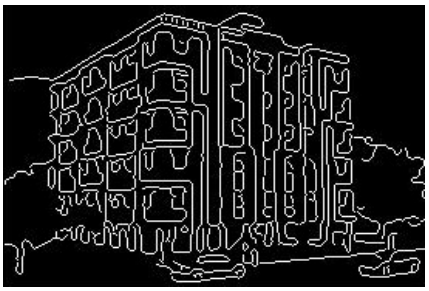


Source: (Ipgri, 1997)

Figure 2.3: Different leaf blade shapes

Source: (Gurusamy et al., 2014)

(a) Original Image



Source: (Gurusamy et al., 2014)

(b) Segmented Image

Figure 2.4: Edge Based Segmentation Results

2.2 Technical Background

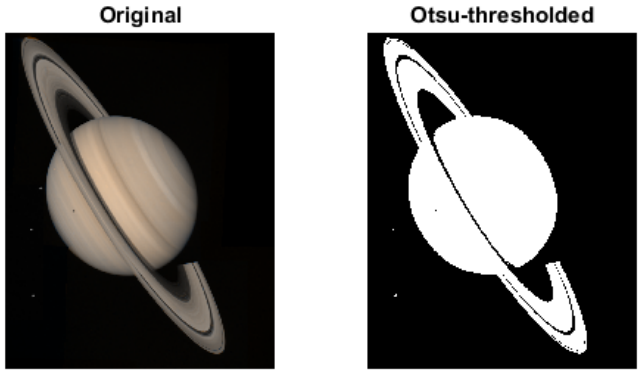
2.2.1 Segmentation

In digital image processing, segmentation technique is performed when an image has to be simplified to something that is more meaningful and easier to analyse. Specifically, during segmentation process, a digital image is partitioned into two or more segments. Image segmentation is commonly used when objects and boundaries need to be located in image. The various segmentation methods and algorithms that have been developed are usually classified in four main categories (Gurusamy et al., 2014).

1. Edge based
2. Threshold
3. Region based
4. Clustering

Edge based segmentation

This technique is based on observing the intensity changes in an image. With the aid of intensity differences, object boundaries can be easily located within image. Edge detection operators are divided into two big categories, first order derivative operators and second order derivative operators. Canny edge detector is an example of a commonly used second derivative operator. In Fig. 2.4 edge based segmentation results are illustrated.



Source: **http://www.biomecardio.com/matlab/otsu_doc.html**

Figure 2.5: Otsu's Threshold Method Results

Threshold Technique

Threshold is the simplest method of image segmentation, based on a threshold value, calculated usually using the original image converted into grayscale. Specifically, image pixels are divided regarding their intensity value. Histogram peaks can be used to acquire the thresholds. There are two main categories of thresholding, global and local. Global threshold T is constant for the whole image. On the basis of T, the segmented image $s(x, y)$ can be obtained from the grayscale image $g(x, y)$ as follows:

$$s(x,y) = \begin{cases} 1, & \text{if } g(x,y) \geq T \\ 0, & \text{otherwise} \end{cases}$$

In local thresholding, the threshold value depends on the neighboor of each pixel. Sample thresholding methods is Otsu's thresholding which is a global threshold and adaptive local thresholding which is a local thresholding method. Fig. 2.5 illustrates a segmented image using Otsu's threshold.

Region Based Technique

Region based techniques for segmentation are based on grouping pixels or subregions into same type of regions. The most common used methods of Region based segmentation are region growing, region splitting and region merging. In region growing, the approach starts with an initial set of points and regarding these points grows regions by appending to the set those neighboring pixels that have similar properties. Alternatively, in region splitting, the

Source: (Zheng et al., 2018)

Figure 2.6: K-Means Segmentation

approach starts with the whole image as a single region and subdivides the regions that do not satisfy a condition of homogeneity. In region merging, small regions that have similar characteristics are merged.

Clustering Segmentation

In clustering segmentation technique, a clustering algorithm is performed in image pixels. The idea is that each cluster usually contains a group of similar pixels that belong to a specific region. The basic clustering algorithm is K-means, which clusters, or partitions the given data into K-clusters. The algorithm is an iterative process. First, K centroids are initialized and then the euclidean distance from the centroids is calculated for each pixel. The pixel is assigned to the closest centroid based on the euclidean distance value. After having assigned all of the pixels, the position of each centroid is recalculated using the relation $c_k = \frac{1}{k} \sum_{y \in c_k} \sum_{x \in c_k} p(x, y)$. The process is repeated until a number of maximum iterations is reached or until an error value is satisfied. K-means segmentation results are shown in Fig. 2.6.

2.2.2 Supervised & Unsupervised Learning

Machine learning algorithms are designed to learn and improve over time when exposed to data. They are usually divided in two main categories: supervised and unsupervised learning.

Supervised Learning

In supervised methods, learning occurs under supervision. The machine is trained using data which is well labelled. In other words, in the given dataset, there is prior knowledge of what

the output should look like. Thus, in supervised learning the model is given a set of labelled data and it is trained to map the input with the correct output accurately. Supervised learning is further categorized as Regression and Classification problems.

Unsupervised Learning

Contrary to supervised learning, unsupervised learning is used in order to deal with data which is unlabelled. The task of unsupervised methods is to find interesting patterns and relationships between the inputs based on their patterns, similarities or differences. Since the model is given unlabelled data, there is no error or metrics in order to evaluate the potential solution. The most common unsupervised learning method is cluster analysis.

2.2.3 Classification

In machine learning, classification is the problem that has to do with the correct assignment of a new observation into the category it belongs. The algorithms that implement classification are known as classifiers. The term classifier also refers to the mathematical function that maps the input data to a certain category. Classification is usually categorized either as binary or multi-class classification. In binary classification, only two classes are involved, while in multi-class classification, there are 3 or more classes.

Binary Classification

Binary classification problems involve the decision whether or not an item has some qualitative property. The class for the item with the property is assigned the class label 1 and the other class is assigned the class label 0. Popular algorithms used for binary classification include: logistic regression, k-Nearest neighbours, Decision Trees, Naive Bayes.

Multi-Class Classification

Most classification algorithms permit the use of more than two classes. However, some of them are by nature binary, e.g. Support Vector Machines. These algorithms, can be turned into multinomial classifiers using a variety of strategies. Multi-class problems are usually addressed using various proposed techniques. Some of them include: Neural Networks, Extreme Learning Machines (ELM), k - Nearest Neighbours, Naive Bayes, Decision Trees.

2.2.4 Feature Selection

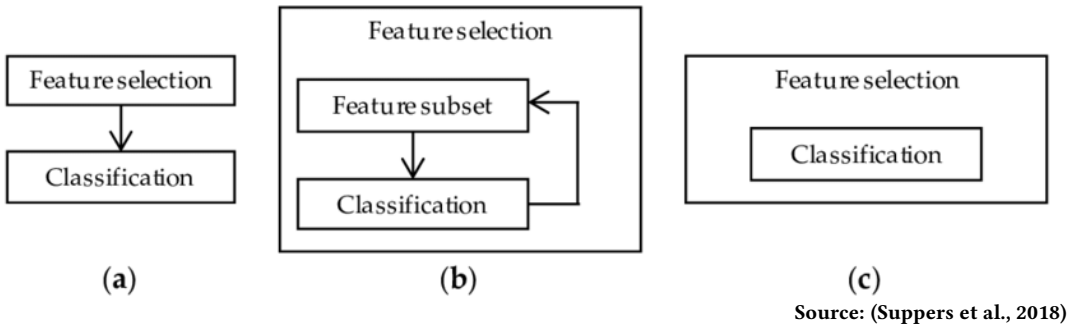
In classification problems, many of the candidate features chosen are either partially or completely irrelevant/redundant to the target concept. These features have to be excluded before performing classification tasks. Feature Selection is the process of selecting an optimal subset of features to use in the model construction. Specifically, it is the elimination of the feature number to those that are believed to be most useful to a model in order to predict the target variable. It is useful to perform feature selection before modeling data for several reasons.

1. Using irrelevant features can decrease the accuracy of the models and train the model based on irrelevant features. Less misleading data leads to improvement of modelling accuracy.

2. An optimal subset of features can reduce overfitting. Less redundant data means less opportunity to make decisions based on noise.

3. It reduces training time. Fewer data points reduce algorithm complexity and algorithms train faster.

Over the past few years, several feature selection methods are proposed and have proven to be efficient in handling high-dimensional data. Feature selection methods can be categorized in a number of ways. The most common one is the categorization according to the dependency of the feature selection search with the construction of the classification model (Saeys et al., 2007). This dependency is illustrated in Fig. 2.7, where (a) depicts filter, (b) wrapper and (c) embedded feature selection methods.



Source: (Suppers et al., 2018)

Figure 2.7: Feature Selection Methods

Filter Methods

Filter methods are widely used for feature selection as they can achieve high computational efficiency without running any learning algorithms. The optimal feature subset is selected using variable ordering. In particular, after the variables are ordered, the less important are discarded, regarding a defined threshold value. The main drawback of these methods is that they are independent of the employed data modelling algorithm. That means that filter methods will select the features even if the latter don't fit in the classification model, thus making them unreliable (Aziz et al., 2017).

Wrapper Methods

Wrapper methods, depend on the performance of classifiers to obtain a feature subset. They don't use feature relevant criteria like filter methods, but the predictive accuracy of a data mining method in order to determine the fitness of a selected subset, by integrating the data mining method as a black box. The aim of this method is to find the subset with the maximum evaluation, by following a trial and error method. This approach forces the method to execute cross validation on small datasets in order to find the most accurate estimation, resulting in better overall performance. The main drawback of the wrapper feature selection methods is that they are very expensive regarding time and computations, when implemented on high dimensional feature space.

Embedded Methods

Embedded methods try to compensate for the drawbacks in the Filter and Wrapper methods. They select the features which best contribute to the accuracy of the model during the modelling algorithm's execution. Specifically, embedded methods do not separate the learning from the feature selection part. These methods are thus embedded in the algorithm either as its normal or extended functionality. Embedded methods are much less prone to overfitting and they are more accurate than filter methods.

Chapter 3

Proposed Methodology

3.1 Segmentation

Before extracting any visual feature, segmentation process is carried out in order to separate leaves from their background accurately. The shape of the leaf has a significant role in recognition thus the segmentation step is of high importance. Usually, it can be accomplished by converting leaf images in grayscale and then applying a threshold (Kadir et al., 2013), (Wu et al., 2007). In some cases segmentation process can be quite challenging though. The input image is typically taken under controlled conditions, however, images taken from real environment are more constrained (Buoncompagni et al., 2015). With regard to our dataset, leaves are placed on a white background in order to acquire the images for the experiment. The constraint here is the shadow that leaves cast on their background. The grayscale conversion approach suffers in the presence of the shadow and it leads to inaccurate segmentation.

3.1.1 Visible Spectral Indexes

In order to overcome the problem concerning the shadows on the background, we took advantage of the fact that leaves in our dataset display a color that varies from bright to dark green. To identify the leaf greenness and separate it from the background along with the shadow, we used a visible spectral-index based method. The visible spectral-index based strategy is a commonly used strategy in many researches for greenness identification (Liu et al., 2013). Such strategies include: the Excess Green Index (EG), Excess Red Index (ER), the vegetative index (VEG), the color index of vegetation extraction (CIVE), the combined index (COM), etc. All these methods are based on the fact that plants have larger green indexes than others in

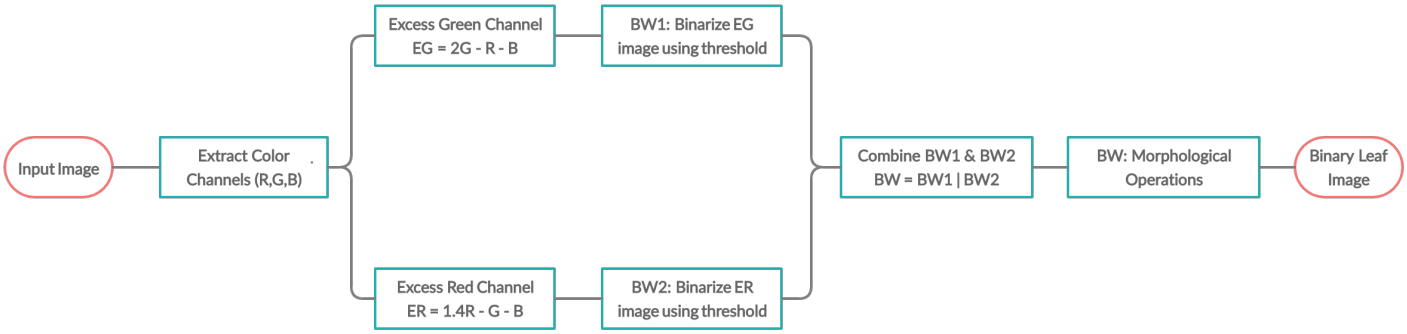


Figure 3.1: Flow Diagram of Segmentation Process

the normalized RGB color space.

Excess Green index

Excess Green index is used to identify the greenness of a leaf and it is defined as follows:

$$EG = 2G - R - B \tag{3.1}$$

where R, G and B are the color components of the input leaf image.

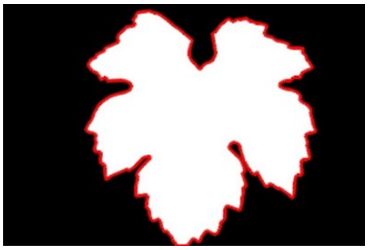
Excess Red index

In order to detect the red parts of the leaves, such as petioles and veins, we use another visible spectral index, excess red index, which is defined as follows:

$$ER = 1.4R - G - B \tag{3.2}$$

where R, G and B are the color components of the input leaf image.

After the two indexes are obtained we perform threshold-based segmentation in each index and then we combine the results in order to obtain the binary leaf. The binary leaf is first dilated and then eroded to remove small regions, holes, or points of noise. The overall process in order to obtain the binary leaf image is shown in Fig. 3.1. Using the binary image, where leaf objects are marked as 1 and background objects are marked as 0, the leaf is separated from its background.



(a) Leaf Perimeter

(b) A leaf and its bounding box

Figure 3.2: Leaf's Perimeter & Bounding Box Visualization

3.2 Feature Extraction

Usually, a leaf is identified using shape, contour, color, texture or vein features. In this thesis, we are going to extract shape and vein features. Using a digital camera, we can capture the shape of every leaf and that makes shape features ideal to use in our method. Texture features are not considered as it is difficult to capture texture using a common digital camera. The color of leaves in our dataset varies from dark to light green and therefore it does not provide significant information. It may be used in future work along with the shape features in order to improve the system's accuracy.

3.2.1 Shape Features

We defined shape features on the basis of morphological features and contour features.

Geometric Features

The basic geometrical features of the leaf are:

(i) Leaf Area: Number of pixels of value 1 on binary leaf image.

(ii) Leaf Perimeter: Number of pixels along the closed contour of the leaf (Fig. 3.2a).

(iii) Equivalent Circular Diameter: Diameter of a circle with the same area as the leaf.

(iv) Physiological Length: The distance between two terminals of the main vein.

(v) Physiological Width: The longest distance between two terminals of the leaf that is orthogonal to the main vein.

Physiological Length & Width are calculated using Matlab's bounding box (Fig. 3.2b).

Morphological Features

Morphological features are calculated combining the geometric features mentioned above to form ratios that are unitless.

Aspect Ratio: The ratio of physiological width W_p to physiological length L_p. Leaves with AR values close to 1 are circular in shape, whether they possess lobing or not. Leaves with values less than 1 are taller rather than wide and vice versa.

$$AspectRatio = \frac{W_p}{L_p}$$

Rectangularity: How similar a leaf is to a rectangle. It is defined as the ratio of the product of physiological length L_p and physiological width W_p and the leaf area A.

$$Rectangularity = \frac{L_p * W_p}{A}$$

Narrow Factor: Narrowness of the leaf. It is defined as the ratio of the equivalent circular diameter of the leaf D and physiological length L_p.

$$Narrow\,Factor = \frac{D}{L_p}$$

Perimeter Ratio of Diameter: The ratio of the perimeter P of the leaf to the equivalent circular diameter D of the leaf.

$$Perimeter\,Ratio\,of\,Diameter = \frac{P}{D}$$

Perimeter Ratio of Physiological Length and Physiological Width: The ratio of the perimeter of the leaf P to the sum of its physiological length L_p and physiological width W_p.

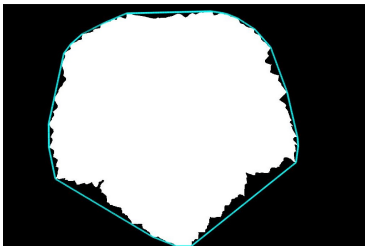
$$Perimeter\,Ratio\,of\,Physiological\,Length\,and\,Physiological\,Width = \frac{P}{L_p + W_p}$$

Compactness: Defined as the ratio of the product of area A with $4 * pi$ to the square of leaf perimeter P. It is sensitive to lobing and serration in the context of grape leaves.

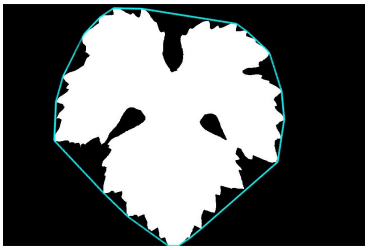
$$Compactness = \frac{4 * pi * A}{P^2}$$

Convexity: The ratio of convex hull perimeter C to leaf perimeter P.

$$Convexity = \frac{C}{P}$$



(a) Unlobed Leaf



(b) Lobed Leaf

Figure 3.3: Leaves and their Convex Hulls

Solidity: The ratio of leaf area A to convex hull area A_c. As the lobation is the main source of concavity in the leaf boundary, it is apparently the cause of the difference between polygons (Fig. 4). Thus, by measuring this difference, the unlobed leaves can be easily detected. The more the leaf's lobation, the more area difference will occur between the decimated polygon and its convex hull (Fig. 3.3).

$$Solidity = \frac{A}{A_c}$$

Roundness: How similar a leaf is to a circle, leaves with value close to 1 are circular. Defined as the ratio of the product of area A with $4 * pi$ to the square of convex hull perimeter C.

$$Roundness = \frac{4 * pi * A}{C^2}$$

Contour-Centroid Distance (CCD) Features

Although the CCD is translation-invariant and can be made scale-invariant by normalization, it requires accurate rotational alignment. Non rotationally - aligned, unnormalized CCDs, however, are still useful for calculating several features. Contour-Centroid Distance $D(i)$ is defined as follows:

$$D(i) = \sqrt{|C_x - E(i)_x|^2 + |C_y - E(i)_y|^2}$$

Where, $D(i)$ is the distance between the centroid of the leaf region and the i_{th} leaf contour pixel. C_x , C_y are the coordinates of the centroid of the leaf region, and, $E(i)_x$, $E(i)_y$ are the coordinates of i_{th} leaf contour pixel.

A leaf with its centroid and its contour points is illustrated in Fig. 3.4.

Using the distance $D(i)$, the following 6 features are calculated.

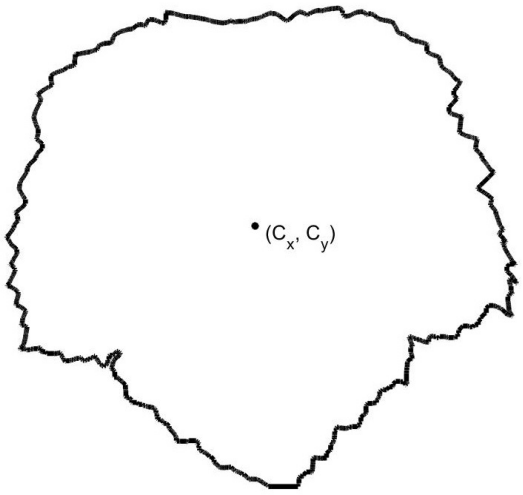


Figure 3.4: Contour points and Centroid of the leaf

Minimum Radial Distance Ratio: Minimum radial distance r_{min} to mean radial distance ratio r_{μ}.

$$Minimum\ Radial\ Distance\ Ratio = \frac{r_{min}}{r_{\mu}}$$

Maximum Radial Distance Ratio: Maximum radial distance r_{max} to mean radial distance ratio r_{μ}.

$$Maximum\ Radial\ Distance\ Ratio = \frac{r_{max}}{r_{\mu}}$$

Mean Radial Distance Ratio: Average radial distance of the leaf r_L to average radial distance of its convex hull r_C.

$$Mean\ Radial\ Distance\ Ratio = \frac{r_L}{r_C}$$

Dispersion: Ratio of the radius of the circumscribed circle r_c and the radius of the inscribed circle r_i of broad leaf's boundary (Fig. 3.5). A feature to deal with a leaf that has irregular shape. The more irregular the leaf's shape, the higher the dispersion value.

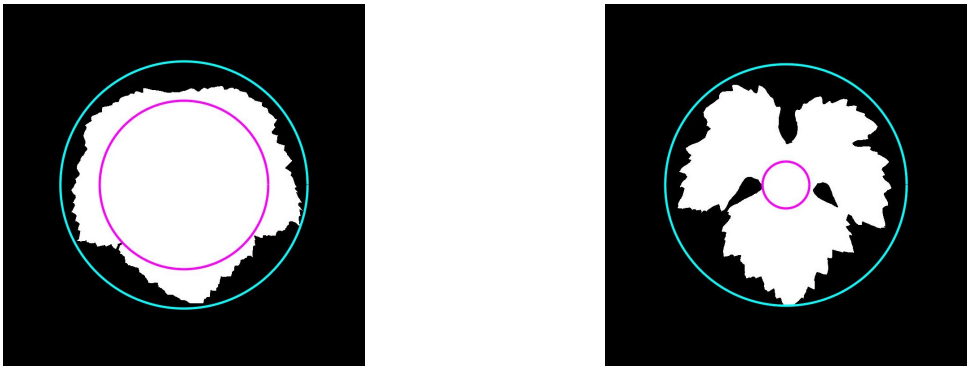
$$Dispersion = \frac{r_c}{r_i}$$

Circularity: Circularity is the ratio of the mean distance between the centroid of the leaf and all of the bounding points μ_R and the standard deviation of the distance from the centroid to the boundary points s_R.

$$Circularity = \frac{\mu_R}{s_R}$$

Sphericity: A feature defined as the ratio of the product of area A with $4 * pi$ to the maximum radial distance r_{max} multiplied by 2.

$$Sphericity = \frac{4 * pi * A}{2 * r_{max}}$$



(a) Leaf with regular shape

(b) Leaf with irregular shape

Figure 3.5: Dispersion Feature

3.2.2 Vein Extraction

If you think of a leaf like a hand, then its veins are the fingerprints and tell their own story. Characteristics present on leaf veins, such as anthocyanin colouration of main veins on upper side of blade, make a major contribution in grapevine leaf identification. Moreover, vein length or the angles formed between the main veins of the mature leaf, are some of the strongest grapevine leaf descriptors measured in classic ampelometry. Using image processing tools we can extract the leaf venation pattern. As already mentioned, leaf veins resemble to fingerprints or vessels. Therefore, it is feasible to adapt fingerprint or vessel detection methods to identify the veins of a leaf. Combined filters are proved to be efficient methods for vessel enhancement (Oliveira et al., 2016). In our approach, we will use a combined Matched and Frangi filter to enhance leave veins in order to perform segmentation.

Matched Filter

A matched filter as introduced by Chaudhuri et al., was developed in order to detect piecewise linear segments of leaf veins in leaf images. Matched filters convolve the image with a 2-D kernel that enhances the vein structure (Chaudhuri et al., 1989).

Frangi Filter

Frangi filters use the eigenvectors of the Hessian to compute the likeliness of an image region to contain vessels or other image ridges (Frangi et al., 1998). In our case, the ridges in the image are the leaf veins.

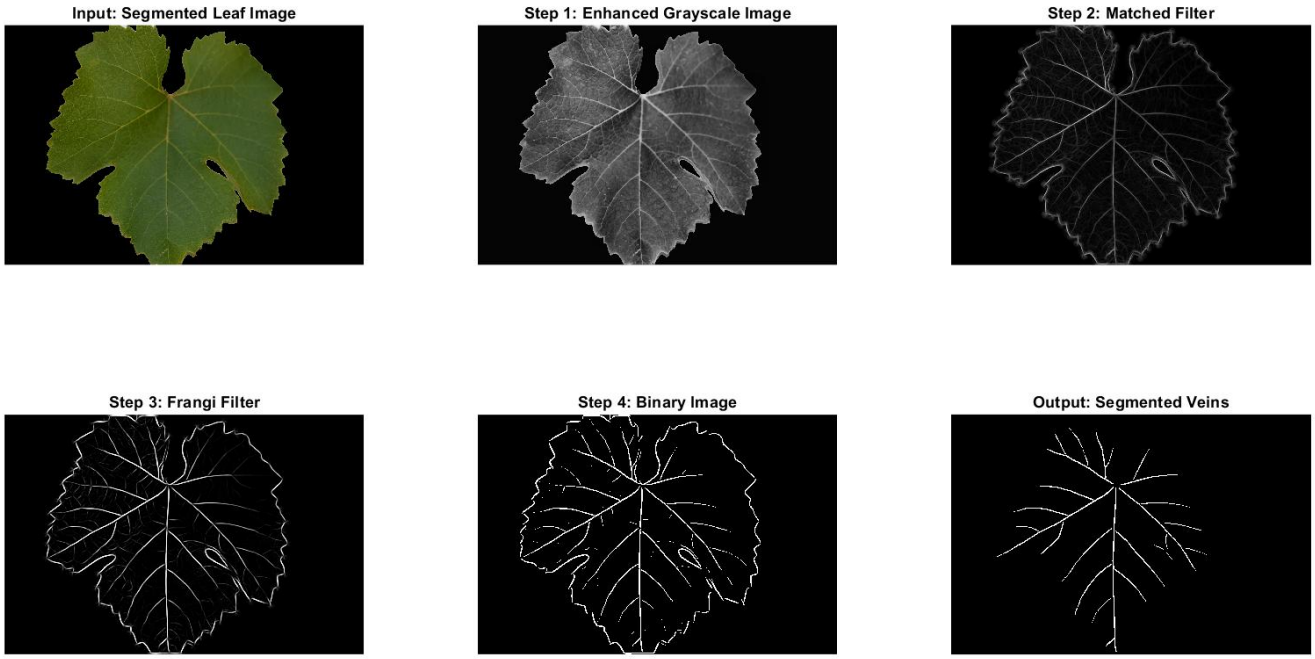


Figure 3.6: Vein Extraction Steps

Before applying the combination of the filters, leaf image is converted in LAB color space and L channel is enhanced. Then we convert it back to RGB and turn it to grayscale. After this process, matched and Frangi filters will be applied. The main motivation for filter combination is the complementarity that they provide. Matched filters enhance the small vessels better, while Frangi's filter is less sensitive to noise. After applying matching and frangi filters on the segmented leaf image, we perform threshold based segmentation and then we remove the perimeter, in order to acquire the veins. Fig. 3.6 illustrates the steps for vein extraction.

3.3 Feature Selection

Feature selection is the process of reducing the number of features when developing a predictive model. To reduce the feature number and acquire an optimal feature subset, two feature selection methods will be tested. Fisher's score which is a filter method and Support Vector Machine - Recursive Feature Elimination (SVM - RFE) which is an embedded method.

3.3.1 Fisher's Score

In this method, first, leaves with similar shapes are clustered together using Self Organizing Map (SOM). After the clusters are formed, the most discriminant features for each one of the clusters are defined, using Fisher's Score feature selection method. Linear Discriminant Analysis (LDA) is performed in each cluster, using the most discriminant features in order to visualize their discriminatory power and check the class separability. The optimal feature subset includes the most discriminant features of each cluster.

Self Organizing Map (SOM)

Clustering is the task of dividing the population or data points into a number of groups (clusters) such that data points in the same groups are more similar to other data points in the same group and dissimilar to the data points in other groups. Self-organizing map (SOM) is a common used method for this purpose. In particular, SOM is a type of artificial neural network (ANN) that is trained using unsupervised learning to produce a discretized representation of the input space of the training data by grouping similar data together. SOM consists of m neurons located at a regular low-dimensional map, usually a 2-D map. A schematic representation of SOM is shown in Fig. 3.7.

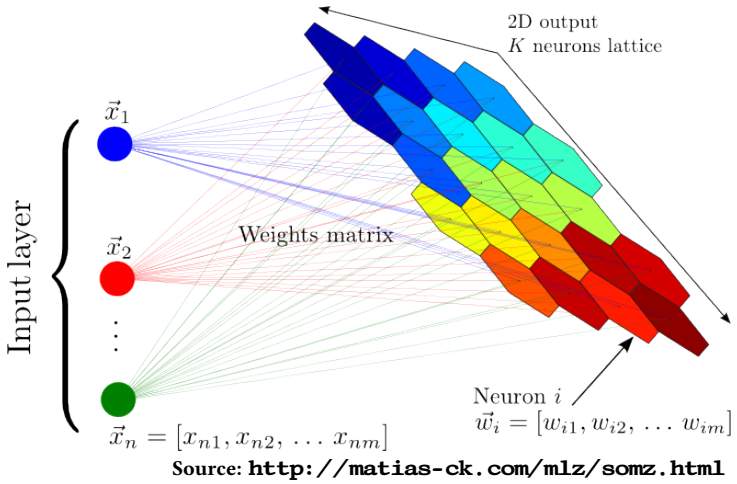


Figure 3.7: Self-Organizing Map Scheme

SOMs apply competitive learning, which involves back propagation and gradient descent. Competitive learning is a form of unsupervised learning in ANNs, in which nodes compete for the right to respond to a subset of the input data. A self-organizing map learns to differentiate and distinguish features based on similarities.

SOM Algorithm:

1. Each neuron's weights are initialized.
2. A vector is randomly chosen from the set of training data.
3. Every neuron is examined to calculate which one's weights are more similar to the input vector. The winning neuron is commonly known as the Best Matching Unit (BMU).
4. Then the neighbourhood of the BMU is calculated. The amount of neighbours decreases over time.
5. The winning weight is rewarded with becoming more like the sample vector. The neighbours also become more like the sample vector. The closer a neuron is to the BMU, the more its weights get altered and the farther away the neighbour is from the BMU, the less it learns.
6. Repeat step 2 for N iterations.

Best Matching Unit is a technique which calculates the distance from each weight to the sample vector, by running through all weight vectors. The weight with the shortest distance is the winner.

Fisher's Discriminant Ratio

The Fisher's discriminant ratio (FDR) is a metric used to quantify the discriminatory power of individual features between two equiprobable classes (Theodoridis et al., 2010). The FDR is defined as:

$$FDR = \frac{(m_1 - m_2)^2}{\sigma_1^2 + \sigma_2^2}$$

where m_1 and m_2 are the respective mean values and σ_1^2 and σ_2^2 the respective variances associated with the values of a feature in two classes. Large absolute mean difference and small variances per class imply large FDR. Maximizing the FDR leads to the best separation between the two classes.

C-class problem

When we have C classes, fisher's generalization involves C - 1 discriminant functions (Duda et al., 2012). Within scatter matrix is defined as follows:

$$\tilde{\boldsymbol{S}}_w = \sum_{i=1}^{C} \tilde{\boldsymbol{S}}_i$$

where, $\tilde{\boldsymbol{S}}_i = \sum_{y \in y_i} (\boldsymbol{y} - \tilde{\boldsymbol{m}}_i)(\boldsymbol{y} - \tilde{\boldsymbol{m}}_i)^t$ and $\tilde{\boldsymbol{m}}_i = \frac{1}{n_i} \sum_{\boldsymbol{y} \in y_i} \boldsymbol{y}$ is the mean vector.

General between - class scatter matrix:

$$\tilde{\boldsymbol{S}}_B = \sum_{i=1}^{C} n_i (\tilde{\boldsymbol{m}}_i - \tilde{\boldsymbol{m}})(\tilde{\boldsymbol{m}}_i - \tilde{\boldsymbol{m}})^t$$

where, $\tilde{\boldsymbol{m}}_i$ is defined as above and $\tilde{\boldsymbol{m}} = \frac{1}{n} \sum_{i=1}^{C} n_i \tilde{\boldsymbol{m}}_i$ is the definition of a total mean vector.

The generalization of C - class FDR is defined below:

$$FDR = \frac{\sum_{i=1}^{C} n_i (\tilde{\boldsymbol{m}}_i - \tilde{\boldsymbol{m}})^2}{\sum_{y \in y_i} (\boldsymbol{y} - \tilde{\boldsymbol{m}}_i)^2}$$

Fisher's score is one of the most widely used unsupervised feature selection methods. It belongs to filter methods. The key idea of the Fisher's score defined above, is to find an optimal subset of features, such that in the data space spanned by the selected features, the distances between data points belonging to different classes are as large as possible, while the distances of the data points belonging in the same class are as small as possible.

Linear Discriminant Analysis

Linear discriminant analysis (LDA) is a machine learning tool, used for dimensionality reduction. LDA picks a new dimension that gives maximum separation between means of projected classes and minimum variance within each projected class. In this method, we pick a 2 dimensional space to project the classes of each cluster, in order to visualize the discriminatory power of the selected features based on their FDR.

3.3.2 Support Vector Machine - Recursive Feature Elimination

Support vector machine recursive feature elimination (SVM-RFE) is a typical wrapper feature selection method, which adopts the manner of a sequential backward elimination. It was created in order to perform gene selection for cancer classification (Guyon et al., 2002). SVM-RFE uses the weight magnitude as ranking criterion in order to determine a small subset of informative features that reduces processing time and provides higher classification accuracy. It works by repeatedly training an SVM classifier, with a subset of features, and in each iteration heuristically removing the features with the smaller feature weights. In each iteration, the pa-

rameters of the classification model (SVM) are re-estimated, by implementing the method of cross validation. The SVM-RFE algorithm is described below (Duan et al., 2007):

SVM-RFE Algorithm:

1. Start: ranked feature set $R = []$; selected feature subset $S = [1, ..., d]$;
2. Repeat until all features are ranked:
   (a) Train a linear SVM with features in set S as input variables;
   (b) Compute the weight vector;
   (c) Compute the ranking scores for features in set $S : c_i = (w_i)^2$;
   (d) Find the feature with the smallest ranking score: $e = argmin_i c_i$;
   (e) Update: $R = [e, R]$, $S = S - [e]$;
3. Output: Ranked feature list R.

3.4 Classification

In machine learning and statistics, classification is a supervised learning approach in which the computer program learns from the data provided and makes new observations or classifications. In this study, the classification problem is multinomial. Therefore, strategies capable of solving multi-class classification tasks will be used. Specifically, three different approaches were chosen for classification: probabilistic approach using Naïve Bayes algorithm, hierarchical approach using Decision Tree algorithm, and Support Vector Machine algorithm with linear and quadratic kernel. Support Vector Machines are by nature binary classifiers. In order to extend them to solve multi-class classification problems One-Versus-All (OVA) technique was developed.

3.4.1 Naïve Bayes Classifier

Naïve Bayes classifier is a probabilistic machine learning model based on applying Bayes' theorem with strong (naïve) independence assumptions between the features. It is an easy to build model. Besides its simplicity, Naïve Bayes is also known to outperform even highly sophisticated classification methods.

Bayes Theorem:

$$P(\omega_i|x) = \frac{P(x|\omega_i)P(\omega_i)}{P(x)}$$

Above,

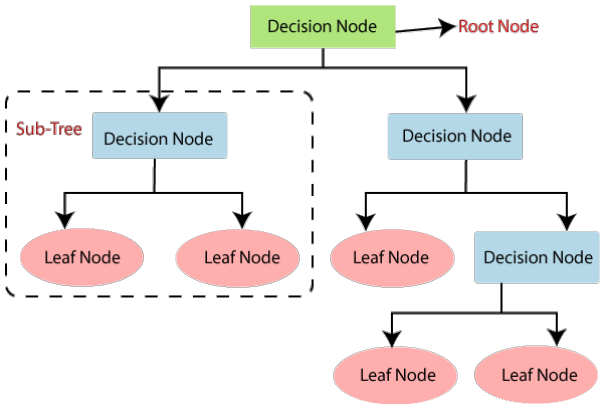
1. $P(\omega_i|x)$ is the posterior probability of class ω_i given predictor x.
2. $P(\omega_i)$ is the prior probability of class.
3. $P(x|\omega_i)$ is the likelihood which is the probability of predictor given class ω_i.
4. $P(x)$ is the prior probability of predictor. It is defined as $P(x) = \sum_{i=1}^{c} P(x|\omega_i)P(\omega_i)$.

The data in each class is distributed to the Gaussian distribution $\mathcal{N}\ (m_i, S_i)$, where m_i is the mean of the class ω_i and S_i is the covariance matrix of the class ω_i. Then, x is assigned to the class ω_i if it fulfils the following equation (Duda et al., 2012):

$$P(\omega_i|x) > P(\omega_j|x),\ \forall j \neq i$$

3.4.2 Decision Tree Classifier

Decision tree is a hierarchical machine learning model, that can be used to solve both classification and regression problems. It falls under the category of supervised learning. A decision tree is an efficient way for graphical representation of every possible solution to a problem, based on given conditions. It comprises of 3 basic segments: a root node, a few hidden nodes, and many terminal nodes (leaves). Fig. 3.8 illustrates the structure of a decision tree. Decision trees are preferred because their decision logic can be illustrated and thus they can be easily understood.



Source: https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm

Figure 3.8: Decision Tree Form

Decision Tree Based Method

The decision tree classifier is organized as a series of test questions and conditions in a tree structure. The root and the internal nodes contain attribute test conditions to separate records that have different characteristics. Every terminal node (leaf) is assigned a class label. Begin the tree from the root node, apply the test condition to the record and follow the appropriate branch based on the output. It then lead us either to another internal node, for which a new test condition is applied, or to a leaf node which defines the class label.

3.4.3 Support Vector Machine (SVM)

A support vector machine (SVM) is a supervised machine learning model that uses classification algorithms for binary classification problems. In SVM data items are plotted in n-dimensional space in a coordinate according to its class. SVM algorithm aims at finding an optimal hyper-plane that discriminates the diverse classes by maximizing the gap between data points on the boundaries.

Suppose a given training dataset with 'n' samples $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$, where x_i is the feature vector with m-dimensional feature spaces. Each x_i will be associated with a value y_i indicating if the element belongs to the class (+1) or not (-1). The formal definition of the dataset is:

$$\mathcal{D} = \{(x_i, y_i)|x_i \in \mathbb{R}, y_i \in \{-1, 1\}\}_{i=1}^{n}$$

The equation of the hyperplane can be written:

$$\boldsymbol{w^T x} + \boldsymbol{b} = 0$$

The width of the margin then is:

$$\frac{2|k|}{||w||}$$

.

Therefore, the problem is defined as follows:

$$maximize\, \frac{2|k|}{||w||}$$

$$subject\, to: \; (\boldsymbol{wx} + \boldsymbol{b}) \geq k, \; \forall x\, of\, class\, 1$$

$$(\boldsymbol{wx} + \boldsymbol{b}) \leq -k, \; \forall x\, of\, class\, 2$$

SVM Kernel Functions

SVM algorithms use a set of mathematical functions that are defined as the kernel. The function of a kernel is to take data as input and transform it into the required form. Different SVM algorithms use different types of kernel functions. These functions can be of different types, each one used for different purpose. Some kernel examples include: linear, nonlinear, polynomial, radial basis function (RBF), and sigmoid. The polynomial and RBF are especially useful when the data-points are not linearly separable. In this approach, linear and quadratic kernel will be tested.

One-Versus-All (OVA)

As mentioned above, many classification algorithms naturally permit the use of more than two classes, but some others are by nature binary algorithms, e.g. SVM algorithm. In order to turn SVM algorithm into a multinomial classifier we follow the one-versus-all (OVA) technique. The main idea of OVA technique is to decompose the classification into K binary problems. Each problem discriminates between one class to all the rest. The OVA methodology is described below.

OVA methodology:

1. Suppose a dataset $D = \{(x_i, y_i)\}$, $x_i \in \mathbb{R}^n$, $y_i \in \{1, 2, 3, ..., K\}$
2. Since there are K possible labels, build K different binary SVM classifiers.
3. Let the positive examples be all the points in class i, and let the negative examples be all the points not in class i.
4. Let $f_i = w_i^T x$ be the binary classifier. The "score" $w_i^T x$ can be thought of as the probability that x has label i.
5. The multi-class hypothesis is defined by $f(x) = argmax_i f_i(x)$.

In order to perform one-versus-all classification it is assumed that each class is individually separable from all the others. OVA is easy to implement and works well in practice.

3.5 Evaluation

In order to calculate the metrics used for evaluation of classification results, we have to introduce the following measures:

1. A True Positive (TP) test result is one that detects the condition when the condition is present.
2. A True Negative (TN) test result is one that does not detect the condition when the condition is absent.
3. A False Positive (FP) test result is one that detects the condition when the condition is absent.
4. A False Negative (FN) test result is one that does not detect the condition when the condition is present.

In order to define the above measures in the case of a multi-class classification problem, we may use one-against-all approach. Therefore, supposing there are n classes:

1. "TP of C_n" is all C_n instances that are classified as C_n.
2. "TN of C_n" is all non-C_n instances that are not classified as C_n.
3. "FP of C_n" is all non-C_n instances that are classified as C_n.
4. "FN of C_n" is all C_n instances that are not classified as C_n.

Using these four measures, we calculated the most commonly used metrics for classification evaluation. These include: Accuracy, Specificity, Sensitivity. In order to define them, we used the confusion matrix method. A confusion matrix for n classes is illustrated in Table 3.1.

The total numbers of true positive (TTP), false negative (TFN), false positive (TFP), and true negative (TTN) for each class i will be calculated based on the generalized equations presented

		Predicted Class			
		Class 1	Class 2	...	Class n
Actual Class	Class 1	x_{11}	x_{12}	...	x_{1n}
	Class 2	x_{21}	x_{22}	...	x_{2n}
	.	.	.	.	.
	.	.	.	.	.
	.	.	.	.	.
	Class n	x_{n1}	x_{n2}	...	x_{nn}

Table 3.1: Confusion Matrix

below (Manliguez, 2016).

$$TTP = \sum_{j=1}^{n} x_{jj}$$

$$TFN = \sum_{j=1, j\neq i}^{n} x_{ij}$$

$$TFP = \sum_{j=1, j\neq i}^{n} x_{ji}$$

$$TTN = \sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij} - TTP - TFN - TFP$$

Using the above measures, accuracy, specificity and sensitivity are calculated as follows:

$$Accuracy = \frac{TTP}{TTP + TFP + TFN + TTN}$$

$$Sensitivity = \frac{TTP}{TTP + TFN}$$

$$Specificity = \frac{TTN}{TFP + TTN}$$

Chapter 4

Results & Discussion

In this section, we present the results of the experiments we have conducted to demonstrate the effectiveness of the proposed method.

4.1 Experiment Dataset

Our dataset contains a total of 144 leaf images of 54 different grapevine leaves. The leaf is placed in a white background in order to acquire the image. Some of the images contain a ruler which is removed along with the background in the segmentation step. Most of the leaves cast shadows on their background. Sample images of our dataset are shown in Fig. 4.1.

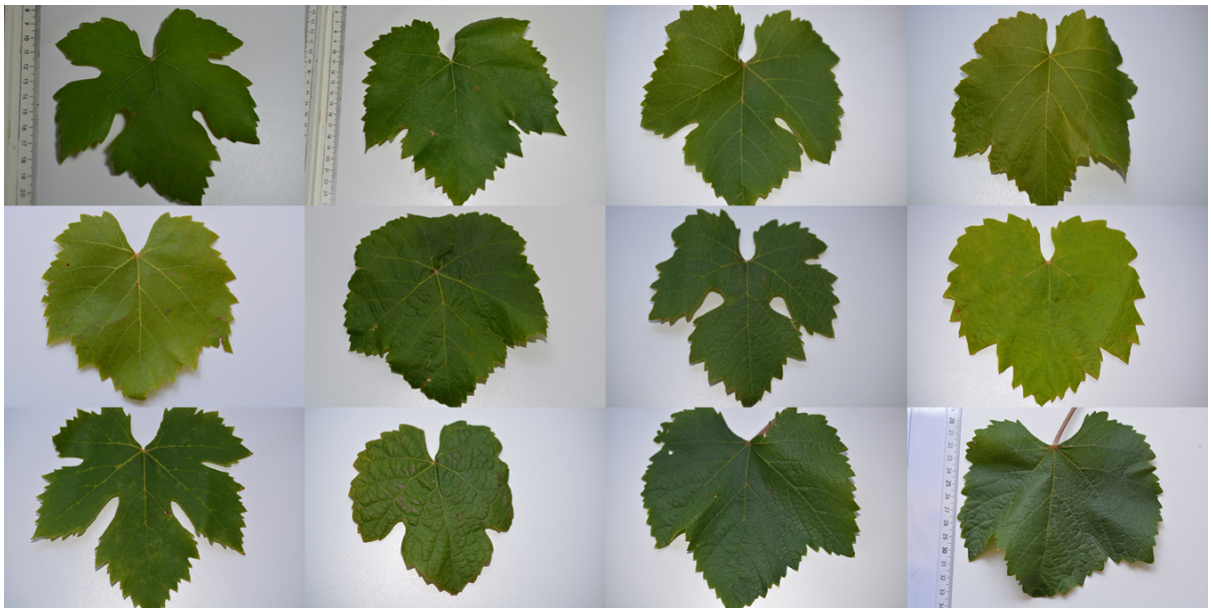


Figure 4.1: Sample images from our dataset

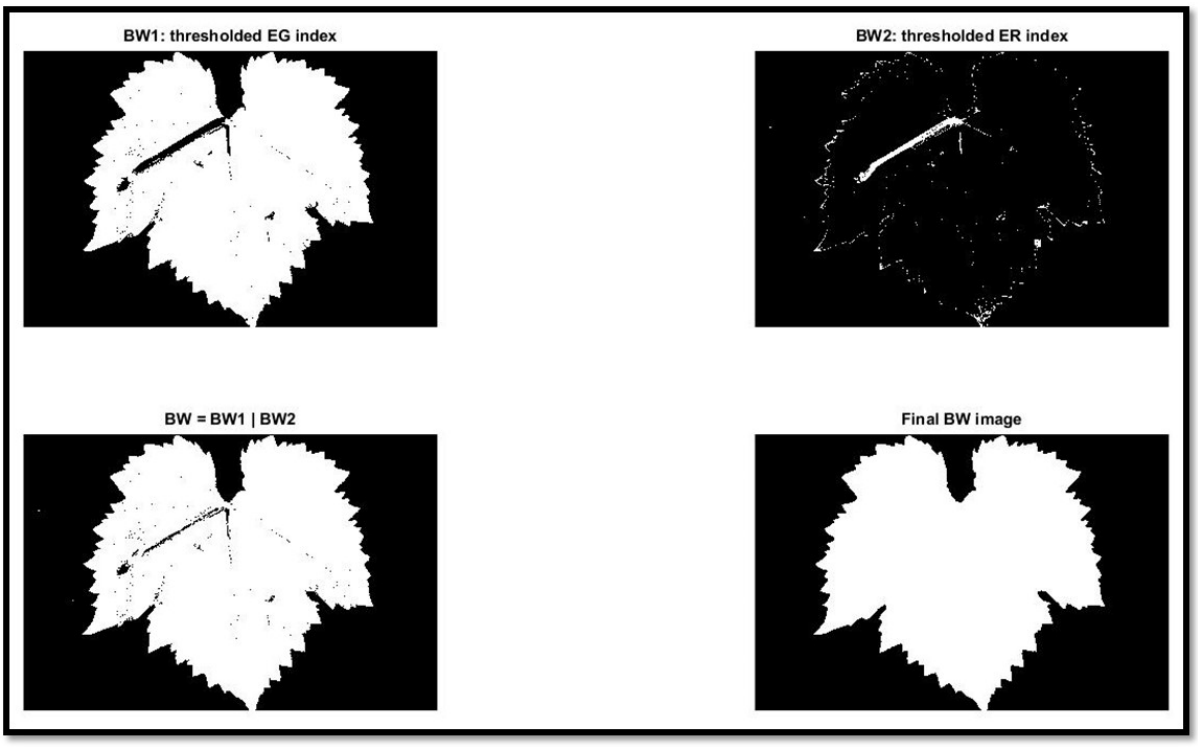


Figure 4.2: Combination of EG & ER binarized images.

4.2 Segmentation Results

Using the proposed visible spectral index methodology for segmentation we managed to separate leaves from their background efficiently. In Fig. 4.2 we can see how the combination of thresholded EG and thresholded ER indexes results in the final binary leaf image.

In Fig. 4.3 we present some indicative segmented leaf samples along with their corresponding binary leaf image.

Figure 4.3: Segmented leaf samples along with their binary images

4.3 Feature Extraction Results

In Fig. 4.4 we present some indicative leaves from our dataset along with their extracted features. Features are numbered as follows: 1. Compactness, 2. Perimeter ratio of diameter, 3. Dispersion, 4. Convexity, 5. Solidity, 6. Aspect Ratio, 7. Circularity, 8. Min Distance Ratio, 9. Max Distance Ratio, 10. Perimeter Ratio of length and width, 11. Roundness, 12. Sphericity, 13. Rectangularity, 14. Narrow Factor, 15. Mean Radial Distance Ratio.

(a) Leaf 1 (b) Leaf 2 (c) Leaf 3 (d) Leaf 4 (e) Leaf 5 (f) Leaf 6 (g) Leaf 7 (h) Leaf 8

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Leaf 1	0.4700	4.5825	1.4981	0.7367	0.9066	1.1218	11.2430	0.7868	1.1787	2.0951	0.8661	7.0600	1.5178	0.9700	0.9520
Leaf 2	0.2449	6.3492	6.1954	0.5716	0.7987	1.0488	3.5181	0.2198	1.3617	2.8922	0.7495	6.4664	1.5330	0.9333	0.8110
Leaf 3	0.3388	5.3972	2.5427	0.6421	0.8649	0.9725	5.8281	0.5130	1.3044	2.4979	0.8217	6.3309	1.4857	0.9129	0.8883
Leaf 4	0.3732	5.1429	2.7969	0.6657	0.8753	1.0523	6.5801	0.4358	1.2188	2.3837	0.8420	7.0089	1.4806	0.9512	0.9097
Leaf 5	0.4054	4.9344	2.3502	0.6990	0.8640	1.0486	8.3399	0.5082	1.1944	2.3043	0.8296	6.9313	1.4586	0.9567	0.9281
Leaf 6	0.2165	6.7527	5.4491	0.5405	0.7938	0.9848	3.5940	0.2557	1.3931	2.9638	0.7411	6.0575	1.6521	0.8711	0.8150
Leaf 7	0.3528	5.2897	3.0598	0.6465	0.8770	1.0152	4.6129	0.4133	1.2645	2.4581	0.8439	7.2541	1.4739	0.9365	0.8621
Leaf 8	0.2499	6.2842	3.5581	0.5661	0.8114	1.0023	4.7757	0.3759	1.3375	2.8111	0.7801	6.2302	1.5906	0.8957	0.8470

Figure 4.4: Indicative Feature Extraction Results

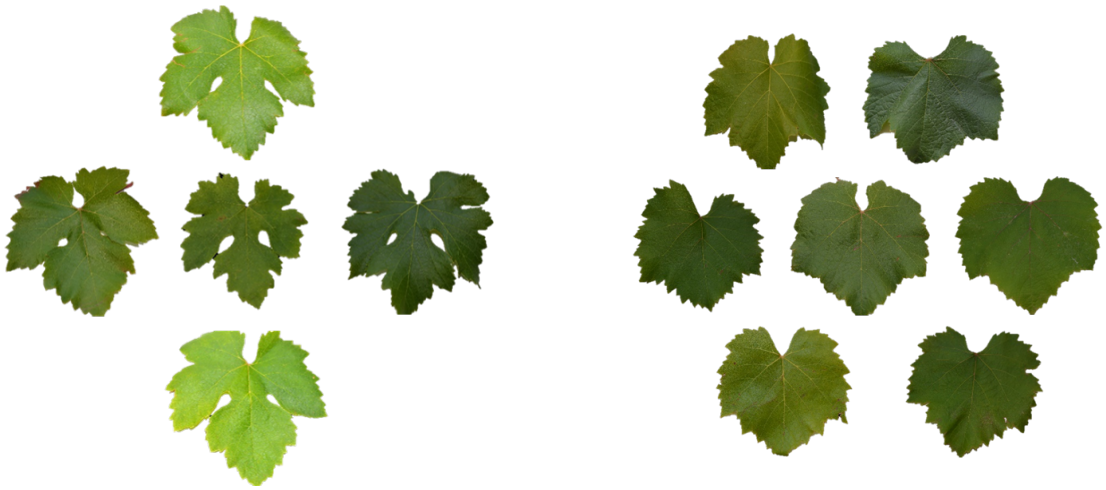
As already mentioned, the value of each extracted feature indicates a certain property regarding leaf's shape. For example, convexity is the relative amount that an object differs from a convex object. High convexity value, close to 1, indicates a relatively convex leaf shape. On the other hand, low convexity value, indicates that a leaf has irregular boundaries. Leaf 1 (Fig. 4.4a) has a convexity value of 0.7367 whereas Leaf 6 (Fig. 4.4f), which has a less convex shape, has a convexity value of 0.5405. Convexity measures local irregularities on a leaf's boundary. Another example is compactness, which is a feature affected from lobation and serration on the context of the leaf. Specifically, it has a low value in the presence of intense serration and deep sinuses, just as in the case of Leaf 6 (Fig. 4.4f) and Leaf 8 (Fig. 4.4h), and greater otherwise.

4.4 Feature Selection Results

4.4.1 Fisher's Score

Clustering: SOM 3x3

In this feature selection method, we first cluster leaves with similar shape together using a self organizing map 3x3 and then we calculate fisher's discriminant ratio for each cluster. SOM 3x3 results in 8 different clusters. Sample clusters of different grapevine leaves with similar shapes are shown in Fig. 4.5.



(a) Cluster 3

(b) Cluster 4

Figure 4.5: Sample Clusters

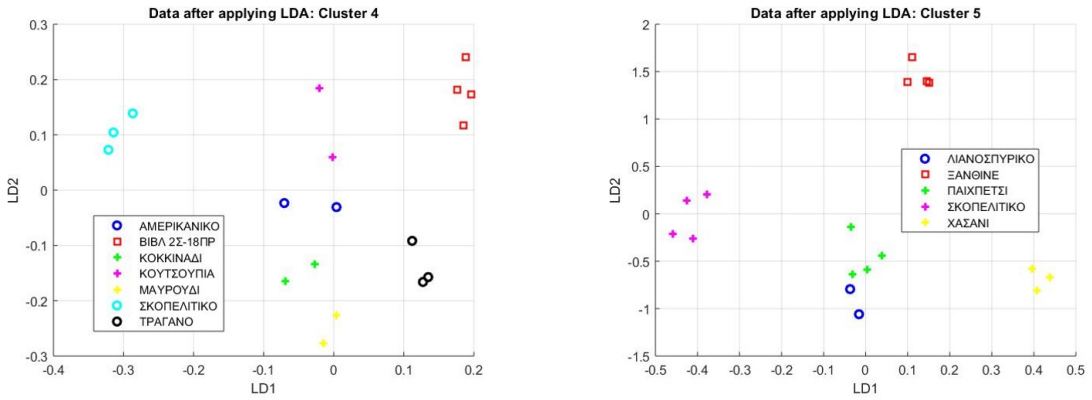
Table 4.1 displays the obtained results from fisher's score. FDR1, FDR3, ... , FDR9 are the fisher's discriminant ratios corresponding to cluster1, cluster3, ..., cluster9 respectively.

Table 4.1: Fisher's Discriminant Ratio

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
FDR1	0.8982	0.9385	2.1816	0.7706	66.1765	9.9105	8.8164	1.5731	1.1518	0.5211	3.4113	1.3594	2.3705	0.9215	8.4552
FDR3	1.8164	1.6569	8.1769	1.0676	7.6515	8.7930	14.2594	3.1441	7.8080	1.1304	4.1583	9.2409	4.8230	7.2411	13.9878
FDR4	8.3578	8.2325	4.5479	9.6389	5.7376	4.4160	3.1441	7.3272	4.0712	8.7325	4.8763	7.6046	2.5887	4.4579	5.7709
FDR5	4.3894	4.5613	1.7629	5.7939	2.9404	7.5597	2.2178	2.1621	1.8264	7.5165	0.9612	5.6328	0.0678	1.9728	5.3797
FDR6	7.0346	5.1290	0.6273	4.1222	12.1367	5.6141	2.3144	0.5324	3.7860	3.4263	14.1193	7.8364	2.0198	3.6736	2.7253
FDR7	2.9589	2.7374	7.1304	2.9216	7.6880	15.8464	1.8172	12.0155	4.2634	2.5272	5.3498	4.1539	3.3111	6.5742	14.6060
FDR8	0.4196	0.3833	1.9871	0.4214	5.8673	6.2186	0.4906	1.7086	5.2807	0.3505	6.5895	1.8192	0.9316	2.2083	0.8858
FDR9	4.5946	4.6232	5.8657	3.3458	20.0509	38.3373	2.7509	8.9132	2.3123	3.2543	13.9379	3.9491	8.7300	17.1663	2.9681

Features presented in table 4.1 are numbered as follows: 1. Compactness, 2. Perimeter ratio of diameter, 3. Dispersion, 4. Convexity, 5. Solidity, 6. Aspect Ratio, 7. Circularity, 8. Min Distance Ratio, 9. Max Distance Ratio, 10. Perimeter Ratio of length and width, 11. Roundness, 12. Sphericity, 13. Rectangularity, 14. Narrow Factor, 15. Mean Radial Distance Ratio.

Features were ordered in a descending order per cluster according to their FDR score and the most informative ones were chosen. The discriminatory power of the feature subset per cluster was visualized using LDA (Fig. 4.6). After selecting the most discriminant features for each cluster, a rate was calculated for each feature by counting the number of clusters for which it was selected as most discriminant. Features with zero rate were excluded from the optimal feature subset. Therefore, the optimal feature subset consists of 13 features. In table 4.2 the rate of each feature is shown.



(a) Cluster 4

(b) Cluster 5

Figure 4.6: LDA Visualization: Sample Clusters

4.4.2 Support Vector Machine - Recursive Feature Elimination

In SVM-RFE method, the algorithm computes the ranking weights for all features and then sorts them according to weight vectors. A support vector machine classifier (One-Versus-All) with a linear kernel was trained for this purpose. After getting the features ranked in descending order, we choose the first 10 to form our optimal feature subset. Feature's Ranking after performing SVM-RFE is presented in table 4.3

Feature	Rate
Aspect Ratio	87.5%
Mean Radial Distance	50%
Sphericity	50%
Solidity	50%
Roundness	37.5%
Convexity	37.5%
Compactness	25%
Perimeter Ratio of D	25%
Circularity	25%
Min Distance Ratio	25%
Max Distance Ratio	25%
Perimeter Ratio of L & W	25%
Narrow Factor	25%
Dispersion	0%
Rectangularity	0%

Table 4.2: Feature's Occurring Rate

Feature	Rank
Circularity	1
Min Distance Ratio	2
Dispersion	3
Mean Radial Distance	4
Max Distance Ratio	5
Solidity	6
Rectangularity	7
Aspect Ratio	8
Compactness	9
Roundness	10
Sphericity	11
Perimeter Ratio of L & W	12
Narrow Factor	13
Perimeter Ratio of D	14
Convexity	15

Table 4.3: SVM-RFE Ranking

4.5 Classification Results

In this section, the different results obtained for each representation with each algorithm are presented. Our dataset contains a total of 144 leaf images, 20 of them were used for testing and the rest for training. To cross validate our model and generate averaged results, we repeatedly perform the classification task 30 times, each time shuffling the 20 test images. In each repetition we calculate the accuracy, the specificity and the sensitivity of the system using the confusion matrix. The conducted experiments are presented below:

- Experiment 1: Perform classification using the 15 extracted features.
- Experiment 2: Perform classification using only the feature subset determined using Fisher's score selection method.
- Experiment 3: Perform classification using only the feature subset determined using SVM-RFE selection method.

4.5.1 Experiment 1

Table 4.4 and Fig. 4.7 show the obtained results of the classification using all 15 features. As we see, the probabilistic approach using Naïve Bayes gives the best result, while hierarchical classification approach using Decision Tree gives the worst results.

		Evaluation		
		Accuracy	Specificity	Sensitivity
Algorithms	Naïve Bayes	0.7817	0.9908	0.7702
	Decision Tree	0.5051	0.9831	0.4965
	SVM (Linear Kernel)	0.7117	0.9895	0.7316
	SVM (Quadratic Kernel)	0.7883	0.9948	0.7991

Table 4.4: Evaluation: 15 Features

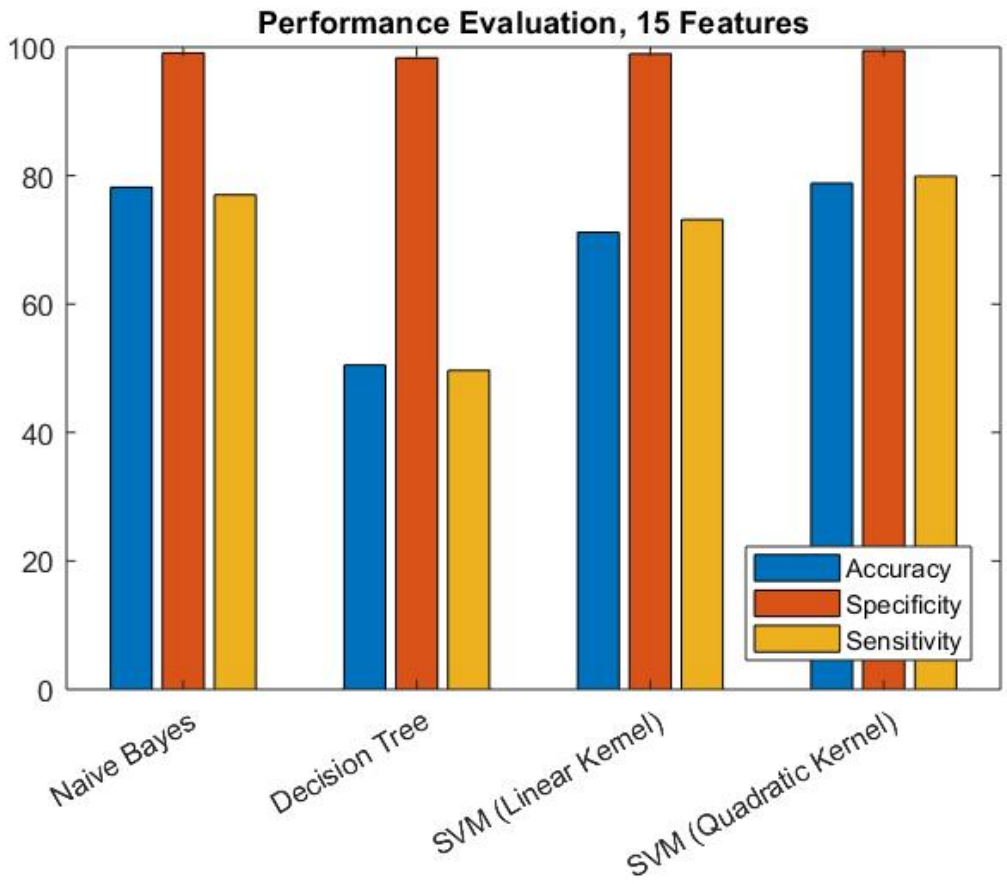


Figure 4.7: Classification Results: 15 Features

4.5.2 Experiment 2

Table 4.5 and Fig. 4.8 show the obtained results of the classification using the subset of 13 features obtained from fisher's score selection method. The performance of Naïve Bayes classifier did not change significantly. SVM with quadratic kernel and decision tree achieved a better performance, while in the case of svm with linear kernel the performance decreased.

		Evaluation		
		Accuracy	Specificity	Sensitivity
Algorithms	Naïve Bayes	0.7717	0.9906	0.7596
	Decision Tree	0.5335	0.9851	0.5254
	SVM (Linear Kernel)	0.6717	0.9862	0.6860
	SVM (Quadratic Kernel)	0.8133	0.9965	0.8254

Table 4.5: Evaluation: 13 Features

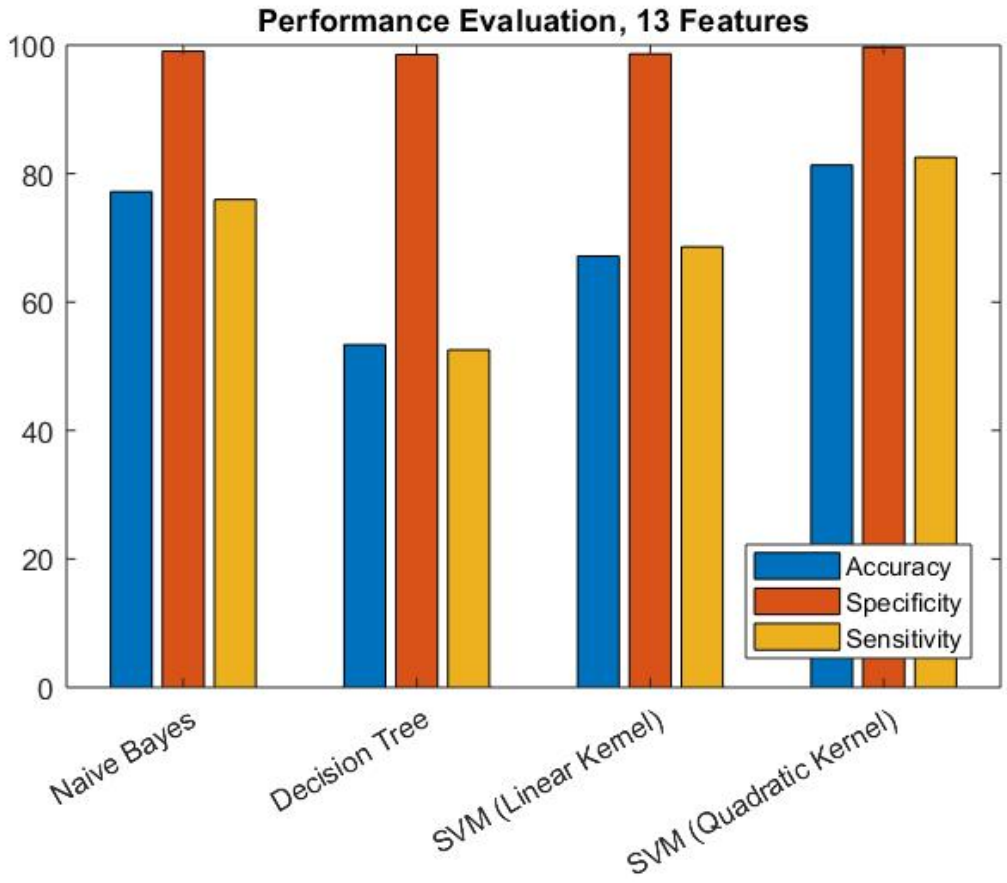


Figure 4.8: Classification Results: 13 Features

4.5.3 Experiment 3

Table 4.6 and Fig. 4.9 show the obtained results of the classification using the subset of 10 features obtained from SVM-RFE feature selection method. As we can see from the obtained results, the performance increased for every classification method. The best results are given from SVM with quadratic kernel.

		Evaluation		
		Accuracy	Specificity	Sensitivity
Algorithms	Naïve Bayes	0.8083	0.9936	0.7982
	Decision Tree	0.5663	0.9854	0.5623
	SVM (Linear Kernel)	0.7500	0.9891	0.7596
	SVM (Quadratic Kernel)	0.8500	0.9947	0.8605

Table 4.6: Evaluation: 10 Features

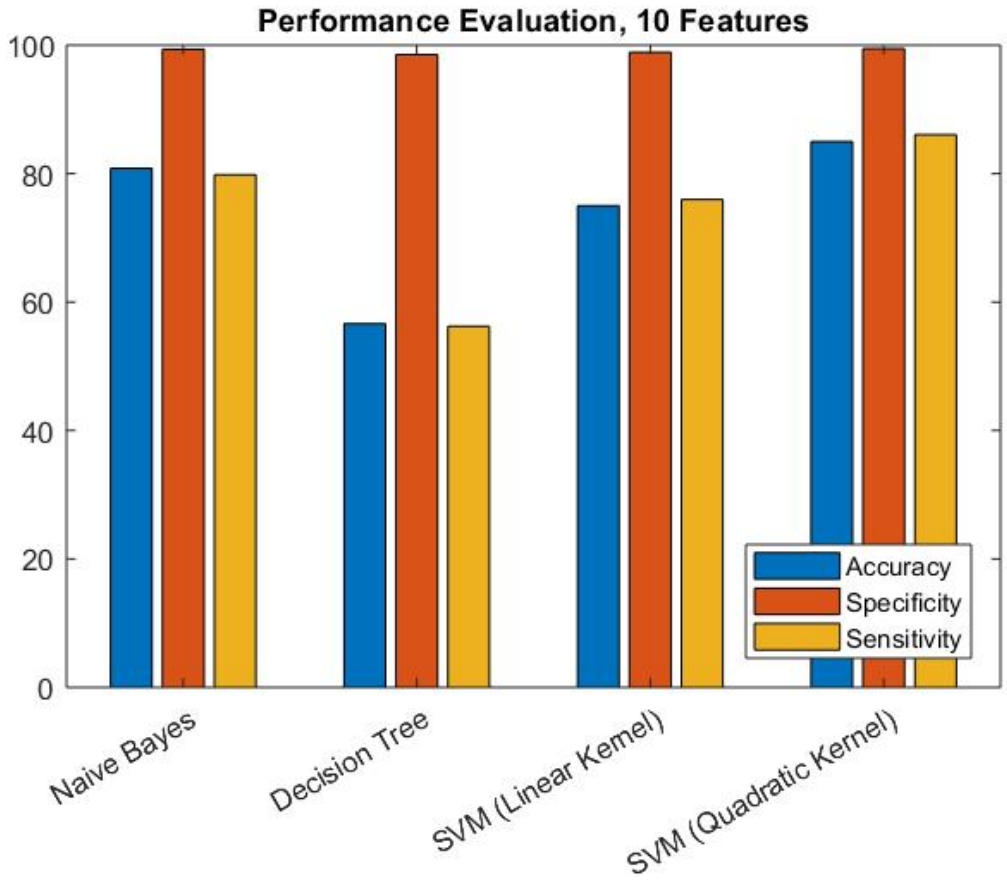


Figure 4.9: Classification Results: 10 Features

4.6 Discussion

4.6.1 Evaluation of the extracted feature set

In traditional ampelography, leaf's shape plays a major role in the identification process. This study implemented 15 shape features, defined on the basis of morphological and contour features, in order to represent grapevine leaf's shape. Our proposed feature set was proved to be efficient for the problem as it fulfils certain conditions. First of all, it is low-dimensional compared to other features and therefore it makes the classification process computationally efficient. Moreover, the computation of the proposed features is simple which implies small execution time. It should be noted that, the proposed features are scale and translation invariant, which means that the location and scaling changing of the leaf can not affect the extracted features. The only drawback is that some of the features, such as dispersion, are not noise resistant. A small 'unwanted' hole on leaf's surface may affect the feature's value significantly. In order to overcome this problem, during the segmentation step, morphological operations were performed. Using the proposed feature set, a satisfying accuracy was achieved.

4.6.2 Comparison of Fisher's Score and SVM-RFE

In order to reduce the dimensionality of the feature space, we tested two different feature selection methods, Fisher's score and SVM-RFE. Fisher's score was a computationally efficient and easy to implement procedure. However, being independent of the classification algorithm implemented, Fisher's score ended up in a feature subset which didn't fit in every classification model tested. Feature selection using SVM-RFE had satisfying results. The selection process was successful, since it managed to filter out the redundant features, without undermining the models' accuracy levels. Especially when it was combined with SVM classifier it was proved to be a powerful tool for classification. A common drawback that is usually discussed regarding SVM-RFE and other wrapper methods, is that they suffer from being computationally expensive. However, in the case of a small dataset, just as in our experiment, SVM-RFE method turned out to be a good solution.

4.6.3 Comparison of classifiers and their capabilities

The classifiers implemented in this experiment are the following: Naïve Bayes, Decision Tree, SVM with linear and SVM with quadratic kernel. Each one has different capabilities and advantages regarding the experiment. After evaluating their performance, we resulted in several observations. It is obvious that Naïve Bayes has a satisfactory performance. In general, this classifier does quite well when we have to deal with low amounts of data just like in our case. However, there is a drawback, as Naïve Bayes delivers optimal classification if the attribute independence assumption holds. This weak point is often addressed using feature selection methods. In terms of Decision Tree, they performed worse than any other implemented classification model. The advantage of this classifier is that the tree can be visualized and hence, for non-technical users, it is easier to explain model implementation. Unlike Decision Tree, SVM achieved high accuracy rates. It is a powerful tool for classification, especially when using a quadratic kernel and it is combined with SVM-RFE method. With these conditions being met, as it can be seen from the results in Table 4.6, SVM achieved an accuracy of 85% outperforming every other classifier. SVM with a linear kernel did not perform that well, provided that our data is not linear separable. However, when it was combined with SVM-RFE it managed to reach a satisfying accuracy of 75%.

Chapter 5

Conclusion & Future Work

Conclusion

The aim of this study was to exploit image processing and machine learning tools in order to upgrade the science of ampelography into a more sophisticated method, using digital leaf images for the identification. The acquired images were pre-processed and the leaf was efficiently segmented from its background. Then, information regarding grapevine leaf's shape was captured using several morphological and contour features, which were used as input in the selected classification models. In order to improve classification's accuracy we tested 2 different feature selection methods. The evaluation results proved that machine learning models are able to produce a rapid and accurate classification result.

Future Work

In the present work, only shape features were used in order to classify grapevine leaves. In the traditional ampelographic approach though, additional descriptors are used, such as color, vein, or hair. In a future scope, the extracted veins could be exploited in order to detect features such as anthocyanin colouration of main veins on upper side of blade, or the density of hair on main veins. This would improve improve the classification accuracy. Furthermore, as the models developed are based on machine learning algorithms, accuracy could be further improved by using more digital images of the existing cultivars for training.

Appendices

Appendix A

Supplementary material

A.1 Segmentation results

Supplementary data to segmentation results can be found online at :

- https://doi.org/10.6084/m9.figshare.12639908.v1
- https://figshare.com/articles/dataset/Segmented_Leaf_Images/12643301

Bibliography

Adão, T., Pinho, T. M., Ferreira, A., Sousa, A., Pádua, L., Sousa, J., … Morais, R. (2019). Digital ampelographer: a cnn based preliminary approach. In *Epia conference on artificial intelligence* (pp. 258–271).

Aziz, R., et al. (2017). Dimension reduction methods for microarray data: a review. *AIMS. Bioengineering*, *4*(1), 179–197.

Buoncompagni, S., et al. (2015). Leaf segmentation under loosely controlled conditions. In *Bmvc* (pp. 133–1).

Chaudhuri, S., et al. (1989). Detection of blood vessels in retinal images using two-dimensional matched filters. *IEEE Transactions on medical imaging*, *8*(3), 263–269.

Chitwood, D. H., Ranjan, A., Martinez, C. C., Headland, L. R., Thiem, T., Kumar, R., … others (2014). A modern ampelography: a genetic basis for leaf shape and venation patterning in grape. *Plant Physiology*, *164*(1), 259–272.

Duan, K.-B., et al. (2007). One-versus-one and one-versus-all multiclass svm-rfe for gene selection in cancer classification. In *European conference on evolutionary computation, machine learning and data mining in bioinformatics* (pp. 47–56).

Duda, R. O., et al. (2012). *Pattern classification.* John Wiley & Sons.

Frangi, A. F., et al. (1998). Multiscale vessel enhancement filtering. In *International conference on medical image computing and computer-assisted intervention* (pp. 130–137).

Fuentes, S., Hernández-Montes, E., Escalona, J., Bota, J., Viejo, C. G., Poblete-Echeverría, C., … Medrano, H. (2018). Automated grapevine cultivar classification based on machine learning using leaf morpho-colorimetry, fractal dimension and near-infrared spectroscopy parameters. *Computers and Electronics in Agriculture*, *151*, 311 - 318. Retrieved from `http://www.sciencedirect.com/science/article/pii/S0168169918302345` doi: https://doi.org/10.1016/j.compag.2018.06.035

Gurusamy, V., et al. (2014). Review on image segmentation techniques.

Guyon, I., et al. (2002). Gene selection for cancer classification using support vector machines. *Machine learning*, *46*(1-3), 389–422.

Ipgri, U. (1997). Oiv. 1997. descriptors for grapevine (vitis spp.). international union for the protection of new varieties of plants, geneva, switzerland/office international de la vigne et du vin, paris, france/international plant genetic resources institute, rome, italy. *This publication is available to download in portable document format from URL: http://www. cgiar. org/ipgri/IPGRI UPOV OIV Via delle Sette Chiese*, *142*(34), 4.

Kadir, A. (2015). Leaf identification using fourier descriptors and other shape features. *Gate to Computer Vision and Pattern Recognition*, *1*(1), 3–7.

Kadir, A., et al. (2013). Leaf classification using shape, color, and texture features. *arXiv preprint arXiv:1401.4447*.

Liu, H., et al. (2013). Developmnet of a green plant image segmentation method of machine vision system for no-tillage fallow weed detection. In *2013 society for engineering in agriculture conference: innovative agricultural technologies for a sustainable future* (p. 95).

Mancuso, S. (2001). The fractal dimension of grapevine leaves as a tool for ampelographic research. *HarFA—Harmonic and Fractal Image Analysis*, 6–8.

Manliguez, C. (2016). Generalized confusion matrix for multiple classes.

Marques, P., et al. (2019). Grapevine varieties classification using machine learning. In *Epia conference on artificial intelligence* (pp. 186–199).

Oliveira, W. S., et al. (2016). Unsupervised retinal vessel segmentation using combined filters. *PloS one*, *11*(2).

Saeys, Y., et al. (2007). A review of feature selection techniques in bioinformatics. *bioinformatics*, *23*(19), 2507–2517.

Sathwik, T., et al. (2013). Classification of selected medicinal plant leaves using texture analysis. In *2013 fourth international conference on computing, communications and networking technologies (icccnt)* (pp. 1–6).

Suppers, A., et al. (2018). Integrated chemometrics and statistics to drive successful proteomics biomarker discovery. *Proteomes*, *6*(2), 20.

Tassie, L. (2010). Vine identification–knowing what you have.

Theodoridis, S., et al. (2010). *Introduction to pattern recognition: a matlab approach*. Academic Press.

Wu, S. G., Bao, F. S., Xu, E. Y., Wang, Y.-X., Chang, Y.-F., & Xiang, Q.-L. (2007). A leaf recognition algorithm for plant classification using probabilistic neural network. In *2007 ieee international symposium on signal processing and information technology* (pp. 11–16).

Zheng, X., Lei, Q., Yao, R., Gong, Y., & Yin, Q. (2018). Image segmentation based on adaptive k-means algorithm. *EURASIP Journal on Image and Video Processing*, *2018*(1), 68.