Computational Methods For The Classification Of Plants

James S. Cope

2014

Faculty of Science, Engineering and Computing

Kingston University, London

Abstract

Plants are of fundamental importance to life on Earth. The shapes of leaves, petals and whole plants are of great significance to plant science, as they can help to distinguish between different species, to measure plant health, and even to model climate change. The current availability of botanists is increasingly failing to meet the growing demands for their expertise. These demands range from amateurs desiring help in identifying plants, to agricultural applications such as automated weeding systems, and to the cataloguing of biodiversity for conservational purposes. This thesis aims to help fill this gap, by exploring computational techniques for the automated analysis and classification of plants from images of their leaves.

The main objective is to provide novel techniques and the required framework for a robust, automated plant identification system. This involves firstly the accurate extraction of different features of the leaf and the generation of appropriate descriptors. One of the biggest challenges involved in working with plants is the high amounts of variation that may occur within a species, and high similarity that exists between some species. Algorithms are introduced which aim to allow accurate classification in spite of this.

With many features of the leaf being available for use in classification, a suitable framework is required for combining them. An efficient method is proposed which selects on a leaf-by-leaf basis which of the leaf features are most likely to be of use. This decreases computational costs whilst increasing accuracy, by ignoring unsuitable features.

Finally a study is carried out looking at how professional botanists view leaf images. Much can be learnt from the behaviour of experts which can be applied to the task at hand. Eye-tracking technology is used to establish the difference between how botanists and non-botanists view leaf images, and preliminary work is performed towards utilizing this information in an automated system.

List of Publications

Chapter 2

J.S. Cope, D. Corney, J.Y. Clark, P. Remagnino, and P. Wilkin "Plant Species Identification Using Digital Morphometrics: A Review ", *Expert Systems With Applications*, vol. 39(8), 2012, pp. 7562 - 7573

Chapter 3

J.S. Cope, P. Remagnino, S. Barman, and P. Wilkin "Plant Texture Classification Using Gabor Co-occurrences ", *International Symposium on Visual Computing (ISVC)*, 2010, pp. 669 - 677

J.S. Cope and P. Remagnino "Classifying Plant Leaves from Their Margins Using Dynamic Time Warping ", *Advanced Concepts for Intelligent Vision Systems (ACIVS)*, 2012, pp. 258 - 267

J.S. Cope, P. Remagnino, S. Barman, and P. Wilkin "The Extraction of Venation from Leaf Images by Evolved Vein Classifiers and Ant Colony Algorithms ", *Advanced Concepts for Intelligent Vision Systems (ACIVS)*, 2010, pp. 135 - 144

Chapter 4

J.S. Cope and P. Remagnino "Utilizing the Hungarian Algorithm for Improved Classification of High-Dimension Probability Density Functions in an Image Recognition Problem ", *Advanced Concepts for Intelligent Vision Systems (ACIVS)*, 2012, pp. 268 - 277

J.S. Cope and P. Remagnino "Classification of High-Dimension PDFs Using the Hungarian Algorithm ", *Structural, Syntactic, and Statistical Pattern Recognition (S+SSPR)*, 2012, pp. 727 - 733

T. Beghin, J.S. Cope, P. Remagnino, and S. Barman "Shape and Texture Based Plant Leaf Classification ", *Advanced Concepts for Intelligent Vision Systems (ACIVS)*, 2010, pp. 345 - 353

C. Mallah, J.S. Cope, J. Orwell, and P. Remagnino "Plant Leaf Classification using Probabilistic Integration of Shape, Texture and Margin Features ", *Signal Processing, Pattern Recognition and Applications*, 2013, pp. 345 - 353

Chapter 5

J.S. Cope, P. Remagnino, S. Mannan, K. Diaz, F.J. Ferri, and P. Wilkin "Reverse engineering expert visual observations: From fixations to the learning of spatial filters with a neural-gas algorithm ", *Expert Systems With Applications*, vol. 40(17), 2013, pp. 6707 - 6712

Contents

List Of Abbreviations

EFA - Elliptic Fourier Analysis

EFD - Elliptic Fourier Descriptors

PCA - Principal Component Analysis

PDF - Probability Density Function

CCD - Centroid-Contour Distance

SF - Shape features

DTW - Dynamic Time Warping

ROC - Receiver Operating Characteristic

ACO - Ant Colony Optimization

EMD - Earth Mover's Distance

BOW - Bag-of-Words

KDE - Kernel Density Estimation

kNN - k-Nearest-Neighbour

GNG - Growing-Neural-Gas

Chapter 1

Introduction

Plants form a fundamental part of life on Earth, producing breathable oxygen, food, fuel, medicine and many more products that are of tremendous use to mankind. They help to regulate the climate, and provide food and habitation to a multitude of other organisms. As such, a good understanding of plant life is highly important. It allows us to improve agricultural methods, increasing productivity, to mitigate our effects on the environment and to develop new pharmaceuticals. In particular, there are threats to many ecosystems from a changing climate and the demands of a growing human population, and so a need to study the biodiversity of different geographical regions.

Botanists are trained to be capable of examining and indentifying specimens, and discovering the connections between different species. As demand for such expertise increasing, there is, however a growing shortage of such individuals. As such, there is now an increasing interest in having an automated system for performing such tasks. Traditionally, botanists rely on largely qualitative descriptors of plant features. Collections of specimens, preserved in herbarium archives, such as those at the Royal Botanic Gardens in Kew, London, are also

of great use, providing known examples of species that can be referred back to (figure 1.1b). There now exist some projects to digitise these collections, thereby making them increasingly accessible. In recent years, well established morphometric techniques have been employed by some botanists, to further improve the understanding of the relatedness of certain species.

(a) Isolated leaf (b) Herbarium specimen (c) Live specimen

Figure 1.1: Example of various types of leaf image. (a) Isolated leaf on a plain background; (b) Herbarium specimen from Royal Botanical Gardens, Kew; and (c) live specimen with complex background.

Whilst botanists may use all aspects of a plant when trying to identify a specimen, for an automated system, certain organs (eg. flowers, leaves, fruit) appear more appropriate than others. For many species, the flowers are highly distinct, yet are only available for short periods of the year, and are three dimensional in nature, leading to difficulty in reliably capturing the required details. These limitations are true also for fruit and seeds. Other organs, such as a tree's branches or the root system are present throughout the year, but are again difficult to capture in a form appropriate for automated analysis. As such, leaves appear to be the most ideal organ to use. They are available for examination for much of the year, even year-round in the case of evergreen

species. They are also typically more two dimensional (flatter), and therefore can be accurately imaged using a camera or a flat-bed scanner.

1.1 Aims and Objectives

The aim of this work is to help fill a gap in botanical expertise, by exploring computational techniques for the automated analysis and classification of plants from images of their leaves, providing novel techniques and the required framework for a robust, automated plant identification system.

There are several objectives towards achieving this. The first is to be able to extract and describe the required information from leaf images. Leaves have many components that can be useful for identification, including the shape, margin characteristics, the texture, venation patterns, and other aspects such as the presence of hairs. It is important to first be able to accurately extract these components from a leaf, and create adequate descriptors that can be compared to those from other leaves of known species.

One of the main challenges for species identification comes from the fact that some species have high intra-species variation, whilst conversely, the variation between different species can be very low. As such, for some leaf components for some sets of species, there may be significant overlap. One of the objectives here to develop machine learning techniques that are capable of taking this issue of variation into account.

A third regards how different modalities (i.e. leaf components and feature-sets) can be combined. Whilst it may be possible to classify leaves with a reasonable accuracy using only single components, it seems obvious that there is great benefit in using multiple components. As such, an appropriate framework is required. Further to this, due to the aforementioned issues of inter- and intra-variation, some components may be appropriate to use for some species, but

not for others. Therefore, a part of this objective is to allow for the automated selection of components and feature-sets to be used, on a leaf-by-leaf basis.

A final objective is to learn how professional botanists view leaf images, and to apply this knowledge to the problem at hand. This will involve the establishment, through the use of eye-tracking technology, of the differences between how botanists and non-botanists perform this task, and, moreover, investigation into the application of this information in an automated system.

1.2 Applications

Potentially, there are many practical application of this work. One such application is to provide tools to support botanists. As previously discussed, botanical expertise is currently in limited supply, and so automated identification systems could help to reduce their workload. Furthmore, whilst some morphometric analysis of leaves is currently performed by some botanists, this typically requires time-consuming manual measurements to be made, and is not conviently available to many.

An automated species identification system would also be of great use to amateur botanists, gardeners and other interested persons, who may desire to be able to indentify a plant but lack the skills and knowledge to do so unassisted. Indeed, with the increasing prevalence of smart phone devices, there currently exist a few rudimentary tools for doing so [3, 75, 143].

There has also been interest in this area from within the agricultural sector. New ways are constantly being sought to increase crop yield and farm efficiency whilst decreasing costs. There have been several studies [44, 125] into the use of robotic farm helpers for the purposes and both gathering crops and eliminating weeds. Both of these tasks would require to be able to distinguish between the crops being grown and other unwanted plants.

For both scientific and conservation purposes, the study of the geographical distribution of plants is of great interest, particularly in relation to the planet's changing climate. It is well established that the climate in which a plant grows affects the shape of its leaves [116], and so it is desirable to be able to quantify this, performing comparisons between different climate zones, and for different years within the same geographical location. As many habitats are currently shrinking, whether due to changing environment conditions or to human activity, it is important to be able to identify the distribution of different species in order to properly assess the impact. An automated identification system would allow the range of flora within a location to be catalogued without those performing it to require the specialist training that is in short supply.

1.3 Outline

The remainder of this thesis is structured as follows. In Chapter 4, numerous new methods for the extraction and description of the main leaf components, including the margins, texture and venation, are described. A comparative study of the most commonly used existing leaf-shape analysis techniques is also presented. Chapter 5 describes methods for reliably identifying leaves whilst taking into account the intra-species variation, before examining methods for combining the classification of multiple modalities, and the dynamic selection of feature-sets to use, to both increase accuracy, and decrease computational requirements. In Chapter 6, a study, using eye-tracker technology, of how botanists study leaves is reported, looking at how this information could be used to improve computer-based systems.

Chapter 2

Literature Review

2.1 Introduction

Although morphometrics and image processing are well-established and broad disciplines, botanical morphometrics presents some specific challenges.Leaves and flowers are non-rigid objects, leading to a variety of deformations. Many leaves have a three-dimensional nature, increasing the difficulty in producing good quality leaf images, whilst resulting in the loss of useful structure information. Archived specimens may also be damaged as they are dried and pressed, but even live specimens may have insect, disease or mechanical damage. Automated systems must be robust to such deformations.

Any system that is concerned with distinguishing between different classes of plants must be aware of the large intra-class (see Figure 2.1), and small inter-class variation that is typical of botanical samples. For example, a number of classifiers have been developed [54, 75] that identify the species of a specimen from a digital image of it. These must be robust in order to distinguish between very similar looking specimens from different species, when a single species may

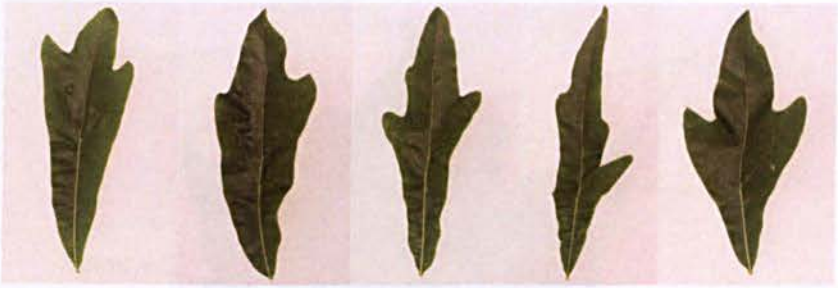


Figure 2.1: Variation in leaves taken from a single specimen of *Quercus nigra*.

by itself produce a very wide range of different leaves. See Figures 2.3 and 5.5 for further examples of the variety of leaf shapes found.

Distinguishing between a large number of classes is inherently more complex than distinguishing between just a few, and typically requires far more data to achieve satisfactory performance. Even if a study is restricted to a single genus, it may contain many species, each of which will encompass variation between its constituent populations. The flowering plant genus *Dioscorea*, for example, contains over 600 species [47], so even single-genus studies can be very challenging. On a related note,as the leaves develop, the shape may vary continuously along a single stem, or discretely (known as leaf heteroblasty), which can further confound shape analysis unless great care is taken of the specimen sources.

Different features are often needed to distinguish different categories of plant. For example, whilst leaf shape may be sufficient to distinguish between some species, other species may have very similar leaf shapes to each other, but have different coloured leaves. No single aspect, or kind of aspect, may be sufficient to separate all the categories, making feature selection a challenging problem.

This chapter will give an overview of the previous work has been carried out in the field of computational methods for the analysis of plants, covering techniques for the extraction and comparison of diffent plant features as well as

Figure 2.2: The main components of a typical leaf.

outlining many of the possible applications of this work.

2.2 Leaf Analysis Methods

There are many aspects of a plant leaf's structure and appearance that are used by botanists in plant morphological research. The most useful of these leaf-components in comparative biology are usually the two-dimensional shape of a leaf or petal, the characters of the leaf margin (such as the teeth), and the structure of the vein network. Of these, the outline shape has received by far the most attention when applying computational techniques to botanical image processing.

2.2.1 Leaf Shape

There are several reasons underlying the focus on leaf shape. Firstly, the shape has perhaps the most discriminative power. Although leaves from the same plant may differ in detail, it is often the case that different species have characteristic leaf shapes, and these have often been used by botanists to identify species.

Whereas differences in margin character or vein structure may be fairly subtle, shape differences are often more obvious, even to the non-expert. In many cases, leaf *size* is largely determined by the environment, while *shape* is more heritable. Secondly, this is the easiest aspect to extract automatically. If a leaf is imaged against a plain black or white background, then simple threshold techniques can be used to separate the leaf from the background, and the outline can then be found by simply isolating those pixels of the leaf that border the background. Thirdly, there are numerous existing morphometric techniques which can be applied to leaf shape that have already proven their worth for other biological problems and which may already be familiar to many botanists. Finally, the gross structure of a leaf may be preserved even if the leaf specimen is damaged, possibly through age. For example, many dried leaves turn brown, so colour is not usually a useful feature by itself. Note also that many of the shape-based methods discussed here have also been applied to petal, sepal or whole flower shape, as discussed in Section 2.2.6.

Figure 2.2 shows some of the main components of leaves with their corresponding botanical terms, while Figure 2.3 illustrates some of the variety of leaf shapes found.

We now discuss a number of approaches to leaf shape analysis, including Fourier analysis, contour signatures, landmark analysis, shape features, fractal dimensions and texture analysis.

2.2.1.1 Elliptic Fourier Descriptors

The most common shape analysis technique applied to leaves appears to be the elliptic Fourier descriptor (EFD) [73]. Here, leaf shape is analyzed in the frequency domain, rather than the spatial domain. A set number of Fourier harmonics are calculated for the outline, each of which has four coefficients. This set of coefficients forms the Fourier descriptor, with higher numbers of

Figure 2.3: Example of leaf shapes.

harmonics providing more precise descriptions. (Hearn [54] suggests that 10 Fourier harmonics are necessary to accurately represent leaf shape to distinguish between a range of species.) Typically, principal component analysis (PCA) is then applied to the descriptor, to reduce dimensionality and aid discrimination, identifying the main sources of variation within the data. An early example of this approach is by White et al. [142], who found EFDs to be superior to landmark measures, chain codes and moment invariants when characterizing leaf outlines. Elliptic Fourier descriptors can easily be normalised to represent shapes independently of their orientation, size or location, easing comparison between shapes.

McLellan and Endler [90] compared Fourier analysis with several other methods for describing leaf shape. They point out that few landmarks are readily identifiable on most leaves, except perhaps those that have regular lobes, and demonstrate that Fourier analysis can discriminate successfully between various leaf groups. They do note however that none of the methods they considered

was greatly superior to any other.

One advantage of EFDs is that a shape can be reconstructed from its descriptor, as shown in Figure 2.4. Whilst this may be of little use from a classification standpoint, it can provide useful botanical information - a method for helping to explain shape variation is to reconstruct the shape for some "average" descriptor, and then to create reconstructions from this descriptor as it is modified along the first few principal components.

Hearn [54] used a combination of Fourier analysis and Procrustes analysis [46] (a simple shape registration method, based on rotation, translation and scaling) to perform species identification using a large database of 2420 leaves from 151 different species, achieving a 72% classification accuracy. Other recent examples of the use of EFDs to analyze leaf shape include Andrade et al. [4], Furuta et al. [41], Neto et al. [99] and Lexer et al. [79].

A closely related method is "eigenshape analysis". Here, the sequence of angular deviations that define a contour (the angles between adjacent points positioned evenly around the contour) is measured, typically being normalized by choosing a common starting point defined by a landmark. Singular value decomposition is then used to identify the principal components [87], which can be used as inputs to a subsequent classifier or for comparison. Ray has extended this work and applied it to leaf shape analysis [113]. This work consisted of dividing the outline into several segments using recognizable landmarks (see Section 2.2.1.3), and then analyzing each segment using singular value decomposition. One difficulty with this approach is the problem of identifying homologous landmarks in leaves (ie. points which can be guaranteed to exist on all the leaves being examined). While this can be difficult within a single species, it is often impossible between species, as is discussed further in Section 2.2.1.3.

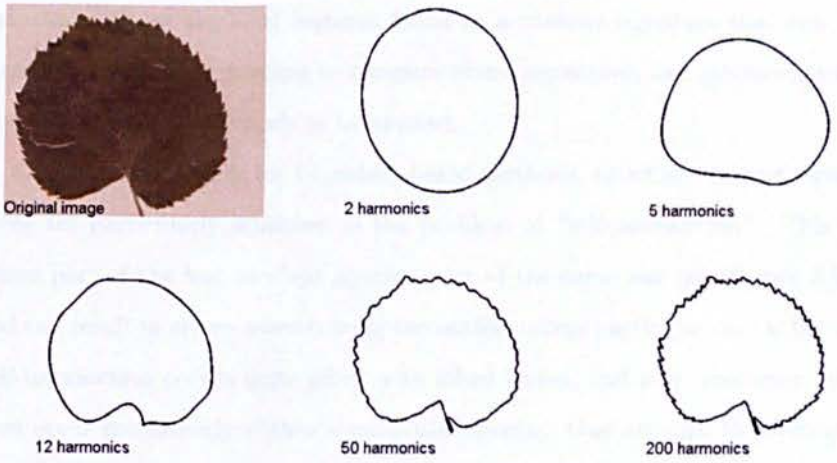


Figure 2.4: An example of elliptic Fourier analysis. As more harmonics are used to reconstruct the original outline, more detail is preserved.

2.2.1.2 Contour Signatures

A number of methods make use of contour signatures. A contour signature for a shape is a sequence of values calculated at points taken around a leaf's outline, beginning at some start point, and tracing the outline in either a clockwise or anti-clockwise direction. One of the most straightforward of these is the centroid-contour distance (CCD). This signature consists of the sequence of distances between the centre of the shape, and the outline points. Other such signatures include the centroid-angle, and the sequence of tangents to the outline. As with EFDs, the aim of creating contour signatures is to represent the shape as a vector, independent of orientation and location. Normalisation can also be applied to enforce independence of scale.

Meade et al. [91] attempted to increase accuracy when applying the CCD to leaves by correlating the frequency of points for measurement with the extent of curvature, whilst Wang et al. [140, 139] applied a thinning-based method to the shape to identify consistent start points for the CCD, avoiding the need to align the signatures before they can be compared. Ye et al. [146] used time-series

shapelets. These are local features found in a contour signature that can be matched, rather than needing to compare entire signatures, and allows existing time-series analysis methods to be applied.

One major difficulty for boundary-based methods, to which contour signatures are particularly sensitive, is the problem of "self-intersection". This is where part of the leaf overlaps another part of the same leaf (see Figure 2.5), and can result in errors when tracing the outline unless particular care is taken. Self-intersection occurs quite often with lobed leaves, and may, moreover, not even occur consistently within a particular species. One attempt to overcome this problem was made by Mokhtarian et al. [95]. They assumed that darker areas of the leaf represented regions where overlap occurred, and used this to try to extract the true outline. They then used the curvature scale space method (CSS) [94] to compare outlines. The main limitation of this method is that it may only work with thin and/or backlit leaves, where sufficient light can pass through the leaf to create the darker areas of overlap.

Figure 2.5: An example of a leaf exhibiting self-intersection – several of the lobes overlap neighbouring lobes

2.2.1.3 Landmarks and Landmark-Related Linear Measurements

Another common morphometric method is the use of landmarks and linear measurements. A landmark is a biologically definable point on an organism (such as the insection-point, Figure 2.2), that can be sensibly compared between related organisms. In some cases, these are homologous points, but may instead be local maxima or minima (for example the points furthest from the central axis), as discussed by Bookstein [13]. Linear and angular measurements between them can then be used to characterize the organism's shape. Landmark methods have been successfully applied to various animal species, and have the advantage of being easy for a human to understand. "Traditional morphometrics" analyzes measurements such as the overall length and width of an object, in contrast to "geometric morphometrics", which uses either outlines (such as methods discussed in Section 2.2.1.1) or specific landmarks and the distances between them [2].

Haigh et al. [51] used leaflet lengths and widths along with measurements of flowers and petioles to differentiate two closely related species of *Dioscorea*. Jensen et al. [65] studied three species of *Acer* using the angles and distances between the manually located lobe apices and sinus bases. Warp deformation grids were also used to study variation. Young [148] used leaf landmarks to compare plants of a single species grown in different conditions. The plants were also imaged at different ages to discover when the method would have the best discriminatory ability. A related method is the inner-distance measure, a metric based on the lengths of the shortest routes between outline points without passing outside of the shape, which was used by Ling et al. [82].

A number of disadvantages exist, however, when applying landmark methods to leaves or other plant organs. The first of these is the difficulty of automatic extraction. For example the leaf's apex (tip) may be hard to distinguish from

the tip of a lobe, whilst the appearance of the insertion point (where the petiole, or leaf stalk, meets the leaf blade) may vary greatly depending on the base angle and how the petiole has been cut. Furthermore, even the length of a leaf may be hard to measure if the leaf is asymmetrical and the main vein does not align with the shape's primary axis. For these reasons, studies involving landmarks and linear measurements have often involved manual data extraction by experts, severely limiting the scale of any system based on them.

The other major problem here is the inconsistency in available landmarks between different species or other taxa. Indeed, the only landmarks present in almost all leaves are the apex and the insertion point, and in the case of peltate leaves (where the stalk is connected near the middle of the blade), the latter does not even appear in the outline shape. As a result, most of the studies using landmarks concentrate on specific taxa where the required features are known to be present.

One of the most significant developments in comparative biology in the last 30 years has been the development of phylogenetic reconstruction methods using morphological data, and latterly nucleic acid or protein sequence data. These methods differ from those dealt with elsewhere in this section in that they use only shared derived characters to infer (phylogenetic) relationship rather than using total overall resemblance for identification or species delimitation. The concept of homology has particular importance in cladistics (the grouping of organisms based on the shared characteristics of their common ancestors) and is perhaps more tightly defined [106]. There has been theoretical debate over the use of continuous and hence morphometric morphological character data in cladistics. Several approaches have been suggested, such as that of Thiele [134]. Zelditch et al. [150] even attempted to use geometric morphometric methods, such as partial warps, to acquire novel phylogenetic character data in

fish, although such techniques have not been widely used in systematics as a whole or been taken up by plant systematists.

2.2.1.4 Shape Features

Similar to linear measurements are shape features, which are also typically limited to analysing the outline of a shape. These are various quantitative shape descriptors that are typically intuitive, easy to calculate, and applicable to a wide variety of different shapes. Commonly used features include the shape's aspect ratio, measures of rectangularity and circularity, and the perimeter to area ratio, amongst others. Some studies have also used more leaf-specific features, for example Pauwels et al. [107] uses a measure of "lobedness". A more general set of features are "invariant moments", which are statistical descriptors of a shape that are invariant to translation, rotation and scale [58], [132].

When analyzing leaves, Lee and Chen [77] argue that "region-based features", such as compactness and the aspect ratio, are more useful than outline contour features because of the difficulty in identifying meaningful landmark points, or in registering different contours against each other. They found that a simple nearest-neighbour classifier using region-based features produced better results than a contour-based method, at least on the 60 species they used as a test case.

Once such a set of features has been extracted from the images, a variety of classifiers can be used in their analysis. A "move median centres" hypersphere classifier was developed by Du and colleagues [33], [138] that uses a series of hyperspheres to identify species in a space defined by a set of shape features and invariant moments. Another study using shape features was carried out by Wu et al. [145], who used an artificial neural network to identify 32 species of Chinese plants from images of single leaves with 90% accuracy, and compared the results against a number of other classifiers.

While shape features have achieved some positive results, they are of limited use for understanding variation. Although the effects of some features may be obvious, such as changes to the height-to-width ratio, variation in other features may be hard to understand because of the difficulty or impossibility of reconstructing shapes from features. For example, the perimeter to area ratio provides a measure of the "complexity" of a shape, but there are many ways in which a leaf might be altered to produce the same change in this value, without affecting the values of many other common shape features.

A more general risk with shape features is that any attempt to describe the shape of a leaf using only (say) 5–10 features may oversimplify matters to the extent that meaningful analysis becomes impossible, even if it is sufficient to assign a small set of test images to the correct categories. Furthermore, many such single-value descriptors are highly correlated with each other [90], making the task of choosing sufficient *independent* variables to distinguish categories of interest especially difficult.

2.2.1.5 Polygon Fitting and Fractal Dimensions

The fractal dimension of an object is a real number used to represent the dimensional space to which the object belongs. This can provide a useful measure of the "complexity" of a shape, which may then be used as an input feature for a classifier, for example. There are many ways to calculate an object's fractal dimension, with the Minkowski-Bouligand method [1] being a popular choice due to its precision and the existence of a multi-scale version. A few attempts have been made to use fractal dimensions to identify leaves. McLellan [90] used the fractal dimension as a single value descriptor alongside other descriptors. Plotze [108] used the positions of feature points in the curves produced by the multi-scale Minkowski-Bouligand fractal dimension, whilst Backes [6] also used the multi-scale Minkowski-Bouligand method, but compared Fourier descriptors

calculated for the curves. Bruno et al. [14] compare box-counting and multi-scale Minkowski estimates of fractal dimension, and used linear discriminant analysis to identify a number of plant species. McLellan and Endler [90] showed that fractal dimension tends to be highly correlated with the perimeter to area ratio (or "dissection index"), suggesting it is of limited additional benefit.

As with shape features, whilst some good results have been achieved, with Plotze [108] claiming a 100% identification rate on a database of 10 species of *Passiflora*, their usefulness in explaining variation is somewhat limited. Given the wide variety of leaf shapes present (e.g. Figure 2.3), characterizing shape by any single measure of complexity may discard too much useful information, suggesting that fractal dimension measures may only be useful in combination with other features.

Du et al. [32] created polygonal representations of leaves, and used these to perform comparisons, while Im et al. [63] represented leaf outlines as a series of super-imposed triangles, which could then be normalized and registered against each other for comparison. The method was shown to correctly identify 14 Japanese plant species, but relies on a number of heuristic assumptions, which may limit the method's applicability to more general tasks.

2.2.2 Venation Extraction and Analysis

After their shape, the next most studied aspect of leaves is the vein structure, also referred to as the venation. Veins provide leaves with structure and a transport mechanism for water, minerals, sugars and other substances. The pattern of veins in a leaf can be used to help identify the plant's taxon. Although the fine detail may vary, the overall pattern of veins is conserved within many species. Veins are often clearly visible with a high contrast compared to the rest of the leaf blade (see Figures 2.6 and 2.7).

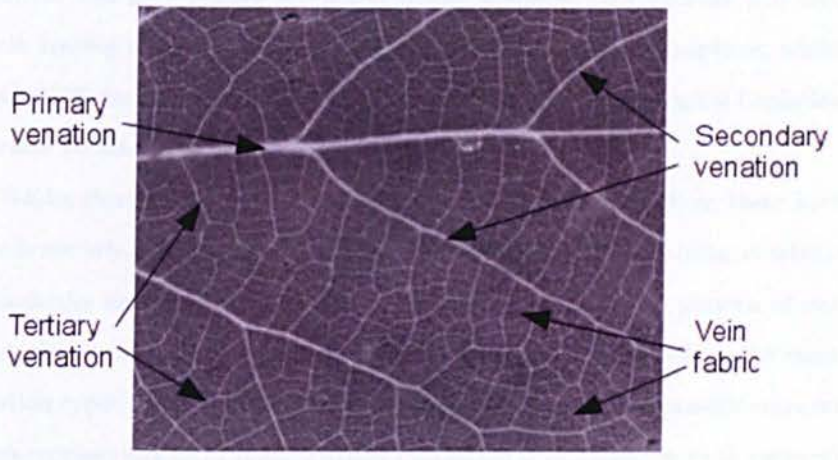


Figure 2.6: Leaf vein structure

A wide variety of methods have been applied to the extraction (and less often, representation) of the vein networks, although arguably with limited success thus far. Clarke et al. [25] compared a couple of simple methods (a scale-space analysis algorithm and a smoothing and edge detection algorithm) to results achieved manually using Adobe Photoshop. They report the quality of the results as judged by some expert botanists, and although the manual results were preferred, the results showed some hope for automatic methods, for at least some species.

Li and Chi [81] successfully extracted the venation from leaf sub-images using Independent Component Analysis (ICA) [26], though when used on whole leaves, the results were no better than using the Prewitt edge detection operator. Artificial ant swarms were used by Mullen [96] to trace venation and outlines in leaves via an edge detection method. Some of the best vein extraction results were achieved by Fu and Chi [39] using a combined thresholding and neural network approach. Their experiments were, however, performed using leaves which had been photographed using a fluorescent light bank to enhance the

venation, and such images are not generally available. Kirchgeßner [70] used a vein tracing method with extracted veins represented using b-splines, whilst Plotze [108] used a Fourier high-pass filter followed by a morphological Laplacian operator to extract venation.

Whilst there have been several attempts at extracting venation, there have been fewer attempts to analyze or compare it, with most of these using synthetic or manually extracted vein images. Park et al. [104] used the pattern of end points and branch points to classify each vein structure as one of the main venation types (see Figure 2.7), and Nam et al. [98] performed classification on graph representations of veins. Further evaluation is required before the general value of venation analysis can be determined.

Figure 2.7: Example of vein structures.

2.2.3 Leaf Margin Analysis

The leaf margin, the outer edge of the lamina, often contains a pattern of "teeth" – small serrated portions of leaf, distinct from the typically larger and smoother lobes (see Figure 2.8 for examples). Despite being a useful aspect for botanists to use when describing leaves, the margin has seen very little use in automated leaf analysis. Indeed, it has been claimed that "no computer algorithm can reliably detect leaf teeth" [118] as yet. This may be due to the fact that teeth are not present in all species of plant; that they are damaged or missing before

or after specimen collection; or due to the difficulty in acquiring quantitative measurements automatically. Nonetheless, teeth are an important feature of many plant species, with botanists using qualitative descriptors of the tooth curvature [36]. Teeth size and number can also be useful indicators of climate and of growth patterns [117], and are even used to make predictions about prehistoric climates using fossilized leaves [36].

Studies using the leaf margin normally combine it with other features and measurements. Clark [23] [24] and Rumpunen [120] both use manually taken measurements of the tooth length and width, used alongside various linear shape measurements. McLellan [90] used the sum of the angles between lines connecting adjacent contour points along with other single value leaf features, and Wang [139] compared histograms of the angles at points spread around the contour.

For taxa that possess teeth, if sufficient, undamaged leaves are available, then teeth area, size and numbers may be useful parameters to measure. Clearly, for taxa that do not possess teeth, other methods must be used – as noted in Section 2.1, different analytical tasks may require different features.

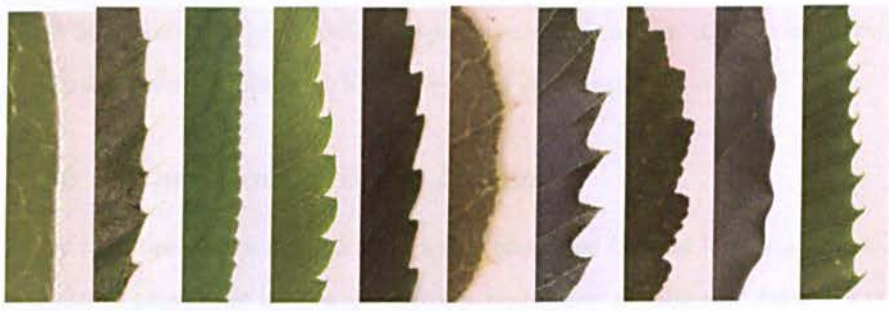


Figure 2.8: Example of leaf margins.

2.2.4 Leaf Texture Analysis

Besides analysing outlines, a number of both traditional and novel texture analysis techniques have been applied to leaves. Backes et al. have applied multi-scale fractal dimensions [6] and deterministic tourist walks [8] to plant species identification by leaf texture, although their experiments involved very limited datasets which makes them hard to evaluate. Casanova et al. [18] used an array of Gabor filters on a larger dataset, calculating the energy for the response of each filter applied, and achieved reasonable results, whilst Liu et al. have presented a method based on wavelet transforms and support vector machines [85]. Other techniques used include Fourier descriptors and grey-scale co-occurrence matrices.

Whilst the above studies were all performed on texture windows acquired using traditional imaging techniques (i.e. cameras and scanners), Ramos [110] used images acquired using a scanning electron microscope (SEM), and Backes [7] used magnified cross-sections of the leaf surface epidermis (the outer-most layer of cells). While these provide interesting results, such images and devices are not commonly available on a large scale.

Where texture is preserved in a specimen, such analysis may prove useful, especially when combined with outline-based shape analysis.

2.2.5 Other Lamina-Based Methods

There have been a few other studies which have used the leaf lamina (surface), or features present on it, in ways different from those already discussed. Gu et al. [49] processed the laminae using a series of wavelet transforms and Gaussian interpolation to produce a leaf "skeleton" (a thin structure representing the interior of the leaf), which is used to calculate a number of run-length features: measure of short runs; measure of long runs; distribution of grey-scales;

distribution of lengths and the percentage of runs.

Qualitative descriptors of the hairs sometimes found on the lamina were used by Clark [24]. These were manually identified and described, and pose a problem for automated systems due to their three-dimensional nature which makes positive identification from a two-dimensional image difficult, even at very high resolutions. Surface glands are another potentially useful lamina feature that have been largely ignored thus far in computational methods.

One intriguing option is to apply 3D imaging and modelling methods to leaf shapes (or to flowers; see below). Ma et al. [86] describe one such method which uses volumetric information from a 3D scanner to reconstruct leaves and branches of plants, though it is not clear how this would would work on a large scale system. Teng et al. [133] combine several 2D photos of the same scene to extract 3D structure, and use the 2D and 3D information together to segment the image, using normalized cuts, finding the leaf boundary. They then use centroid contour distance (CCD, as discussed in Section 2.2.1.2) to classify leaves into broad classes, such as palmate or cordate (see Figure 2.3). Similar work is described by Song et al. [130], where stereo image pairs were analyzed using stereo matching and a self-organizing map. The resulting surface models contained sufficient detail to allow measurements of leaf and flower height, as well as shape.

2.2.6 Flowers and Other Plant Organs

Although the focus of this thesis, leaves are not the only plant organs on which image processing and morphometric techniques have been applied. Traditional "keys" often make use of descriptions of flowers and/or of fruits, but these are often only available for a few days or weeks of the year.

A number of methods have been proposed to identify plants from digital im-

ages of their flowers. Although colour is a more common distinguishing feature here, many methods used to analyze leaf shape can also be used (see earlier sections). Nilsback et al. [100] combined a generic shape model of petals and flowers with a colour-based segmentation algorithm. The end result was a good segmentation of the image, with species identification left to future work.

Das et al. [28] demonstrated the use of colour alone to identify a range of flowers in a database related to patents covering novel flower hybrids. Their method allows the database to be searched by colour name or by example image, although no shape information is extracted or used. A colour-histogram segmentation method was used by Hong et al. [57] and then used with the centroid contour distance (CCD; see Section 2.2.1.2) and angle code histograms to form a classifier. They demonstrated that this method works better than using colour information alone to identify a set of 14 species. This again suggests that outline shape is an important character to consider, especially in combination with other features.

Elliptical Fourier descriptors (Section 2.2.1.1) were used by Yoshioka et al. [147] to study the shape of the petals of *Primula sieboldii*, whilst Wilkin [144] used linear measurements of floral organs, seeds and fruits as well as leaves and PCA methods to investigate whether a closely related group of species in Africa were morphologically distinct or not. They discovered that they in fact formed a single morphological entity and hence all belonged to one species. Gage and Wilkin [42] used EFA on the outlines of tepals (elements of the outer part of a flower, such as petals and sepals) of three closely related species of *Sternbergia* to investigate whether they really formed distinct morphological entities. Clark [24] used linear measurements of bracts, specialized leaf-like organs, in a study of *Tilia*, and Huang et al. [61] analyzed bark texture using Gabor filters and radial basis probabilistic neural networks.

At a smaller scale, the growth of individual grains of barley has been modelled by 3D reconstruction from multiple 2D microscopic images [50]. This allowed both "virtual dissecting" of the grains as an educational tool, and also visualization of gene expressions via mRNA localization. At a smaller scale still, Oakely and Falcon-Lang used a scanning electron microscope to analyze the vessels found in fossilized wood tissue [101]. They used principal component analysis to identify two distinct "morphotypes", which correspond to one known and one novel species of plant growing in Europe around 95 million years ago.

Moving underground, a number of studies have used image processing techniques to analyze root structures in the "rhizosphere" (the region that roots grow in, including the soil, soil microbes, and the roots themselves). For example, Huang et al. [60] used digital images of roots captured by placing a small camera inside a transparent tube placed beneath growing plants. They then used expert knowledge of root shapes and structures (such as roots being elongated and having symmetric edges), to combine multiple sources of information and to fit polynomial curves to the roots, and use a graph theoretic model to describe them. More recently, Zeng et al. [151] used image intensity to distinguish root pixels from soil pixels. They then used a point process to combine and connect segments to efficiently identify complete root systems.

These studies show that while the clear majority of botanical morphometrics research has focussed on leaves, due to their ready availability and use for discriminating between taxa, other plant organs, when available, should not be ignored.

2.3 Applications

In this section, discussion moves beyond specific algorithms in isolation and methods designed for the laboratory, to considering a number of complete sys-

tems and prototypes, designed for practical use in the field. In order to have an impact in the real world, it is important to demonstrate that an algorithm can be applied in practice, and can scale up from a few idealised examples to larger and more complex problems. Systems designed to identify species from plant images; several agricultural applications; and scientific research tools regarding species variation and distribution, and how this relates to the climate are all reviewed.

2.3.1 General-Purpose Species Identification

Plant identification is currently particularly important because of concerns about climate change and the resultant changes in geographic distribution and abundance of species. Development of new crops often depends on the incorporation of genes from wild relatives of existing crops, so it is important to keep track of the distribution of all plant taxa. Automated identification of plant species, for example using leaf images, is a worthwhile goal because of the current combination of rapidly dwindling biodiversity, and the dearth of suitably qualified taxonomists, particularly in the parts of the world which currently have the greatest numbers of species, and those with the largest number of "endemics" (species restricted to that geographic area).

The species to which an organism belongs is often regarded as its most significant taxonomic rank. Accurately identifying an organism to species level allows access to the existing knowledge available linked to that specific name, such as what other species the taxon in question may breed or hybridise with, what its uses are, and so on. A robust automated species identification system would also allow people with only limited botanical training and expertise to carry out valuable field work.

A number of systems have been developed that aim to recognize plant species

from the shapes of their leaves, based on algorithms such as those in Section 2.2.1. One such plant identification system is described by Du et al. [32]. They argue that any *global* shape-based method is likely to perform poorly on images of damaged or overlapping leaves because parts of the leaf perimeter are missing or obscured. Instead, they suggest that local shape-based methods are more robust for this type of task. Their system matches leaves from images by fitting polygons to the contour and using a modified Fourier descriptor with dynamic programming to perform the matching. It aims to be robust with regard to damaged or overlapping leaves, as well as blurred or noisy images. They claim a 92% accuracy for their method on one sample of over 2000 "clean" images, representing 25 different species, compared with 75%-92% for other methods, and that their method is more robust than others for images of incomplete or blurred leaves.

The increasing power and availability of cheap hand-held computers, including personal digital assistants (PDAs) and smart phones, has led to a number of prototype applications. The goal of allowing users, both professional botanists and interested amateurs, to go out into the field and identify plant species using an automated system is a highly desirable goal, although the task is challenging, not least because of the very large number of plant species that may be encountered.

One major and ongoing project aims to produce an "electronic field guide" to plants in the USA [3]. The user can photograph a single leaf, and the system will display images of twenty plant species that have the closest match in terms of shape according to their Inner-Distance Shape Context algorithm, which extends the shape context work of Belongie et al. [10]. A related prototype from the same project includes an "augmented reality" feature [9], and provides a visual display of a herbarium specimen for side-by-side comparison to the

plant in question [143].

The CLOVER system [97] allows users to provide a sketch or a photograph of a leaf using a hand-held computer, which then accesses a remote server. The server retrieves possible matches based on leaf shape, using several shape matching methods including an enhanced version of the minimum perimeter polygons algorithm, and returns the matches to the device to display to the user. The prototype described is demonstrated to work effectively at recognizing plants from leaves, using over 1000 images from a Korean flora, and with the inevitable trade-off between recall and precision.

A similar system uses fuzzy logic and the centroid-contour distance to identify plant species from Taiwan [21]. However, this requires the user to select various characteristics of the plant from a series of menu options, rather than using morphometric analysis directly.

Each of these general-purpose prototypes has been demonstrated to work successfully on at least a small number of species, and under more or less stringent conditions. Currently, there is no such system that is available for everyday use, although interest remains high [83, 75].

2.3.2 Agriculture

Rather than trying to identify a plant as belonging to one particular species, it is sometimes sufficient to perform a binary classification of a plant (for example, as healthy or not healthy), without needing to be concerned about the exact taxon to which it belongs. One goal of automated or "precision" agriculture [16] is to allow targeted administration of weed killer, fertilizer or water as appropriate from an autonomous robotic tractor, not least to minimize the negative impact on the environment of large scale agriculture. To do this, the system must obviously identify plants as belonging to one category or the other, such as

"weed" vs. "crop".

As is often the case with machine vision systems, variable lighting conditions can make image processing very hard. One proposed solution is to control the lighting by building a light-proof "tent" that can be carried on wheels behind a tractor, and which contains lamps inside it along with a camera. One such system successfully distinguishes between crop plants (cabbages and carrots) and weed plants (anything else) growing in field conditions [56]. Whether carrying round such a bulky tent is feasible or not on a larger scale, it is certainly not ideal.

A similar system uses rails to guide a vehicle carrying the camera along carefully laid out plots [44]. Rather than carrying its own lights, the system is only used under standardized illumination conditions (e.g. bright but overcast). This system extracts shape features such as leaf circularity and area and uses a maximum likelihood estimator to identify leaves that are weeds (specifically dock leaves, *Rumex obtusifolius*) in grassland, with around 85%-90% accuracy. A different system to identify dock leaves is described by Šeatović [125], which uses a scanning laser mounted on a wheeled vehicle to generate 3D point clouds. These are then segmented to separate out leaves from their background, and a few simple rules, based on leaf size, are used to distinguish the dock leaves from other leaves in the meadow.

A related attempt to distinguish weeds, crops and soil in field conditions uses morphological image processing [129]. This attempts to identify the centre of each leaf by using colour threshold segmentation and locating the leaf veins. The system locates the veins using a combination of morphological opening and hierarchical clustering. The final classification makes use of *a priori* knowledge about features of the target plant species. A similar system combines morphological processing with an artificial neural network classifier has also been

suggested [103]. A combination of colour segmentation and morphological programming has also been used towards the development of a robotic cucumber harvester [109]. A variety of methods to distinguish various crops from weeds and soil are discussed by Burgos-Artizzu et al. [16], including colour segmentation and morphological processing. The paper also provides a useful overview of research into "precision agriculture", which aims to use modern technology to optimize crop production, allowing for local variation in soil, landscape, nutrients and so on.

2.3.3 Intraspecific Variation, Geographical Distribution, and Climate

It has long been known that the climate in which a plant grows affects the shape of its leaves [118]. Recent work has extended this by using digital image analysis to enhance the botanical and climatic measurements. Huff et al. [62] collected leaves from temperate and tropical woodlands. They analyzed the leaves and measured the shape factor, and found a correlation with the mean annual temperature. The work was then extended to a wider variety of environments (17 in total) in North America [118]. Here, a variety of simple digital image analysis methods were used to semi-automatically measure features such as leaf blade area, tooth area, number of teeth, and major and minor axis lengths. These features were then compared to climatic measurements from the different field locations. Finally, correlations between leaf shape and climate were measured. They confirmed previous findings that plants growing in colder environments tend to have more teeth and larger tooth areas than similar plants growing in warmer environments. One of the goals of this body of work is to support analysis of leaf fossils, with the aim of estimating paleoclimatic conditions. By establishing how leaves from living plants have shapes that correlate with their

environments, it is hoped that fossil leaf shapes can indicate how the Earth's climate has changed in the past, at both a global and a local scale.

An early study by Dickinson et al. [31] used manual digitization (via a tablet) to identify landmarks on cross-sections of leaves, and principal component analysis to analyze the data. They identified both geographic variation between collection sites and also identified intermediate forms of specimens, suggesting various hybridizations had occurred. As mentioned earlier, work by Wilkin and Gage [42] [144] used morphometric analysis to identify species boundaries. In botany, identifying taxon boundaries is often as important as identifying to which taxa a particular specimen belongs.

2.4 Summary

In this chapter, a wide range of morphometric methods used in a wide range of botanical applications have been discussed. It should be clear that no single method provides a panacea for all problems, but rather that appropriate methods must be chosen for each task at hand. Plants are extremely diverse in shape, size and colour. A method that works very well on one group may rely on features that are absent in another taxon. For example, landmarks may be readily definable and identifiable for some taxa, such as those with distinctive lobes, but not for others.

Given the large scale nature of botanical morphometrics and image processing, automation is essential. Any system that requires significant manual effort, for example in tracing leaf outlines or locating landmarks, is unlikely to be practical when scaled up to thousands of specimens. Despite this, in some cases the user may be remain involved in the process with no great cost: if an electronic field-guide provides say ten predictions of species, rather than one, the user may be able to readily choose the most likely answer [3]. Related to this

is the issue of processing speed. The user of a hand-held field-guide may require responses interactively and so (near) instantaneously, whereas if tool is to be used on a large set of images in a botanical laboratory, it may be acceptable to wait overnight for a comprehensive result – assuming no human interaction is needed.

Chapter 3

A Leaf Dataset

For analysing leaf classification techniques, a suitable leaf-image dataset is required. Within the literature, a wide range of different size and quality leaf-image datasets are used. Few of these are publically available, and those that are are tend to either represent a small number of species, have few samples per species or are of inconsistent image quality. Due to this, a new dataset was compiled for this project, making use of the resources available at the Royal Botanic Gardens, Kew. This chapter discusses the construction of that dataset, including examples of all species included.

3.1 Selection and collection of the leaves

With specimen from over 30,000 species of plant it is important to make an appropriate selection of which to include in the dataset. As discussed in Chapter 2, there are many different aspects to a leaf, each of which can take on many forms (for example, types of leaf shape include oval, lobed and palmate), and it is desirable to have as many of these as possible represented, to ensure that methods developed work well on all types of leaves. On the other hand, there

is a need to be able to distinguish between species with similar features. Another consideration is the inclusion of species with high intra-species variation, as these present a particular challenge. With these factors in mind, 100 different species were selected from a wide range of genera, including 38 species of Quercus and 11 species of Acer to test the ability of methods to identify between closely related species.

Leaves were all collected from species-labelled plants at the Kew Royal Botanic Gardens, and images were captured within an hour of collection, to minimise degradation due to having been removed from the plant. Where possible, healthy, undamaged, adult leaves were selected. Leaves were collected from multiple specimens of the same species where available, so that any inter-specimen variation that may occur is represented. A total of 16 leaves were collected for each species, to provide adequate training and testing data.

3.2 Capturing the leaf images

Images were captured in an indoor, artificially lit environment using a 10 megapixel digital SLR camera. Natural light was excluded as far as possible and a light-diffusion screen was set up to improve lighting consistency. Leaves were placed onto herbarium sheets (stiff white paper used for mounting herbarium specimens) to provide a uniform, low-reflection background. For leaves that were curved in a manner which meant they did not lie flat, double-sided adhesive tape was used to hold them in place. The camera was held by a tripod approximately 40cm directly above the leaves, and a wired remote was used to operate it, to minimise any camera movement. Images were captured of both sides of each leaf, as these may contain different details.

3.3 Use of the dataset in this thesis

Where possible the full dataset of 1600 leaves (100 species, 16 leaves per species) has been used for testing methods in this thesis. Due to the time required in collecting and preparing the dataset, a subset of 32 species is used in some parts (primarily Section 5.1.1), as the full set was not available at the time of testing. Likewise, some early work (Sections 4.2.2 and 4.5) was performed on a different dataset of 8 leaves each from 32 species of Quercus.

Figure 3.1: Dataset part one.

Figure 3.2: Dataset part two.

Figure 3.3: Dataset part three.

Figure 3.4: Dataset part four.

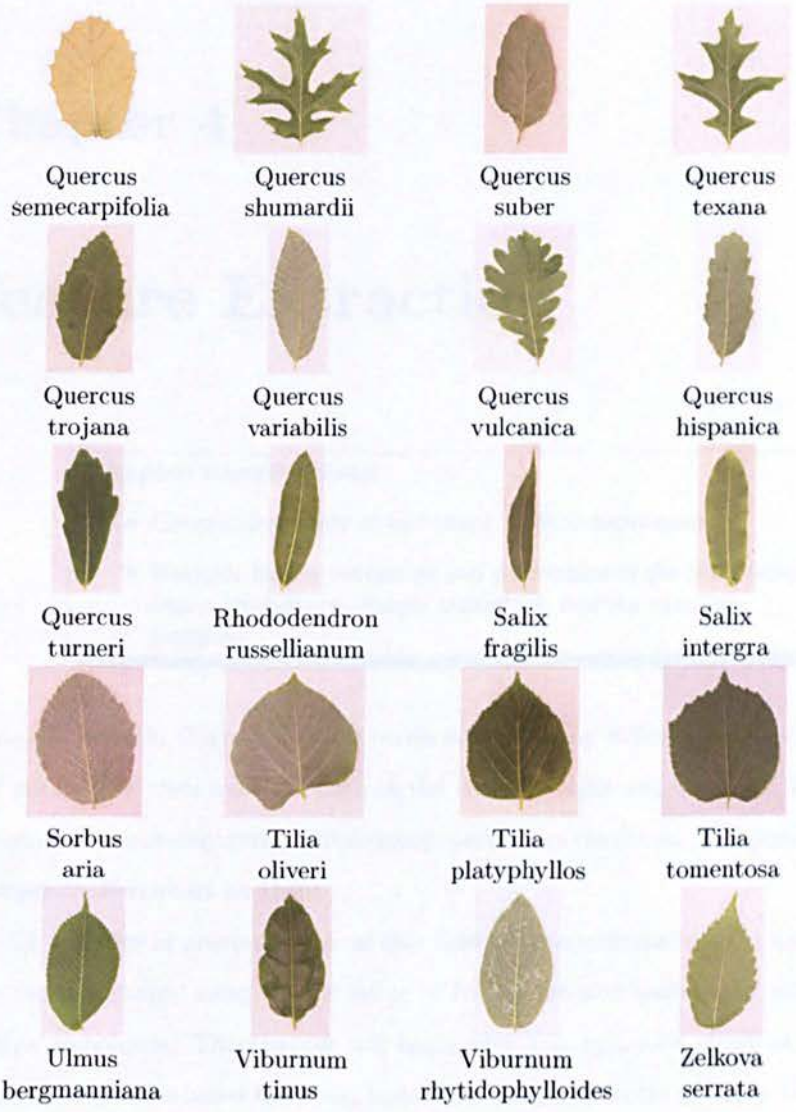


Figure 3.5: Dataset part five.

Chapter 4

Feature Extraction

Chapter contributions:

- Comparative study of leaf-shape analysis techniques.
- Methods for the extraction and description of the leaf macro- and micro-texture, margin characters, and the venation patterns.

As discussed in Chapter 2, plant leaves contain many different components that are used in their analysis, such as the shape, margin and venation. It is important to accurately extract these components from the leaves, and generate appropriate descriptors for them.

The majority of previous work in this field has concentrated on the use of only the leaf shape, using a wide array of traditional and leaf-specific shape analysis techniques. This chapter will begin with a comparative study of the most popular shape-based methods, before continuing on to the primary focus of new methods based on the other major leaf components, namely the texture, the margins and the venation.

4.1 Leaf Shape

As previously discussed in Section 2.2.1, by far the most commonly used leaf feature for automated analysis is the leaf shape. There are likely a number of reasons for this. It is perhaps the most obvious aspect to use - the wide variety of shapes can be easily observed, and might suggest a high discriminative power. The shape is also particularly easy to extract and can be done simple thresholding if the leaf is set against a plain background. Moreover, there is a large array of existing morphometric and shape analysis techniques that can be applied to the problem. In this section, the most commonly used techniques will be applied to the same dataset, in order to evaluate the advantages of each.

4.1.1 Study of Existing Techniques for Leaf Shape Analysis

Of the commonly used techniques some are apparently unsuitable for general leaf classification. Linear measurements, such as angles and spans measure across certain parts of the leaf [51], may appeal to botanists due to their similarity to traditional botanical descriptors, albeit less qualitative. However, some previous catagorization of the leaves, for example as lobed, un-lobed or palmate (lobes radiating from the base of the leaf), may be required in order to select an appropriate set of measurements, and some manual intervention may be required to locate the correct measuring points. Use of landmarks tends to be restricted to studies involving small sets of similar species, due to the lack of landmarks that can said to be common across all types of leaves. For example, Jenson et al [65] used the relative positions of the lobe tips and bases for comparing two 5-lobed species of Acer.

In this study, three common and widely applicable methods will be used: shape features, centroid-contour signatures, and elliptic Fourier descriptors.

Whilst several other techniques, including fractal dimensions and polygon fitting, have been applied to the problem, their use has been limited, and they do not appear to hold any advantages over the above techniques.

4.1.1.1 Shape Features

Shape features are sets of non-leaf-specific geometrical and morphological characteristics that have been selected as appropriate for adequately describing a leaf. One appeal of these is that they are somewhat intuitive, in that it is apparent how a value, such as the aspect-ratio, relates to the leaf's shape. Despite this, these features contain insufficient information to allow for the reconstruction of the shape, and it is quite possible that two visibly different leaves could produce the same set of features. A further problem is that there will likely be a high level of correlation between some features, although this may be resolved through the use of feature selection techniques. Whilst there exists no definitive set of shape features, certain features suitable for the task of leaf classification have been used in multiple instances. The set used here is based on those.

Firstly, the minimum bounding rectangle (MBR) and convex hull (CH) of the leaf are calculated. The leaf is orientated through calculation of the MBR. The following values can then be defined: height, h, and width, w, of the MBR; the areas of the leaf, MBR and CH as A_L, A_{MBR} and A_{CH}; the perimeters of the leaf, MBR and CH as P_L, P_{MBR} and P_{CH}; and the minimum and maximum distances from the centroid to the contour, CCD_{min} and CCD_{max}. The 8 features used are then as follows:

1. Aspect ratio

$$F_1 = \frac{h}{w}$$

2. Rectagularity

$$F_2 = \frac{A_L}{A_{MBR}}$$

3. Ellipticality

$$F_3 = \frac{4A_L}{hw\pi}$$

4. Solidity

$$F_4 = \frac{A_L}{A_{CH}}$$

5. Perimeter Convexity

$$F_5 = \frac{P_L}{P_{CH}}$$

6. Sphericity

$$F_6 = \frac{CCD_{min}}{CCD_{max}}$$

7. Form factor

$$F_7 = \frac{4\pi A_L}{P_L}$$

8. Gravity

$$F_8 = \left| \frac{\bar{y}}{h} - \frac{1}{2} \right|$$

4.1.1.2 Centroid-Contour Signatures

Contour signatures are sequences of values calculated at points spaced around the perimeter of a shape. The most commonly used of these is the centroid-contour distance - the sequence of distances from the centroid of the shape to each boundary point, although several others also exist. Points on the contour can either be evenly spaced in terms of the distance around the perimeter or the angle around the centroid, although problems arise with this latter method when a line extended from the centroid crosses the contour multiple times. Some, such

as Meade et al. [91] have used uneven spacing, such as increased sampling at sections of higher curvature.

Here, two different signatures are used, the centroid-contour distance (CCD) and the angle at centroid angle signature (CAS).

$$CCD(i) = \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2}$$

$$CAS(i) = |\tan(\frac{x_i - x_c}{y_i - y_c}) - \tan(\frac{x_0 - x_c}{y_0 - y_c})|$$

Where x_i,y_i are the x and y co-ordinates respectively for the i^{th} contour point, and x_c, y_c are the x and y co-ordinates of the leaf's centroid.

The CCD is normalised such that all the values in the sequence sum to unity. When comparing two pairs of signatures, orientation invariance is achieved via cross-correlation, whereby the distance between them is measured for every offset of one against the other, and the minimum of these distances is used. This is equivalent to rotating one leaf in relation to another until the difference between the two is minimised.

4.1.1.3 Elliptic Fourier Descriptors

The Elliptic Fourier descriptor of a shape is comprised of the set of coefficients for the first k harmonics of the elliptic Fourier expansion of the contour coordinates. These are given, for the n^{th} harmonic, as

$$a_n = \frac{P}{2n^2\pi^2} \sum_i \frac{\Delta x_i}{\Delta p_i} \left(\cos \frac{2n\pi p_i}{P} - \cos \frac{2n\pi p_{i-1}}{P} \right)$$

$$b_n = \frac{P}{2n^2\pi^2} \sum_i \frac{\Delta x_i}{\Delta p_i} \left(\sin \frac{2n\pi p_i}{P} - \sin \frac{2n\pi p_{i-1}}{P} \right)$$

$$c_n = \frac{P}{2n^2\pi^2} \sum_i \frac{\Delta y_i}{\Delta p_i} \left(\cos \frac{2n\pi p_i}{P} - \cos \frac{2n\pi p_{i-1}}{P} \right)$$

$$d_n = \frac{P}{2n^2\pi^2} \sum_i \frac{\Delta y_i}{\Delta p_i} \left(\sin \frac{2n\pi p_i}{P} - \sin \frac{2n\pi p_{i-1}}{P} \right)$$

where x_i,y_i are the coordinates for the i^{th} point, p_i is the distance around the contour to the i^{th} point, P is the total perimeter distance, and Δx_i,Δy_i,Δp_i are the respective distances between points i and $i-1$.

$$\Delta x_i = x_i - x_{i-1}$$

$$\Delta y_i = y_i - y_{i-1}$$

$$\Delta p_i = p_i - p_{i-1}$$

Elliptic Fourier descriptors are popular with botanists due to the ability to reconstruct the shape from the descriptor. By using PCA to find the main sources of variation within a dataset, this variation can then be visualised by creating descriptors for leaves that have been increased or decreased along each of the principal components [4, 41, 79, 147].

4.1.2 Results and Evaluation

The methods are evaluated on a dataset containing 16 leaves from each of 100 different species. A 16-fold cross-validation is performed, such that one leaf from each species is used each time in the testing set, whilst the remaining leaves are used as the training set. Classification is performed using the k-nearest-neighbour technique, with $k = 15$. Table 4.1 shows the average rates of correct classification and standard deviations for each of the methods. The standard deviation given here is the deviation in classification rates between

different species.

Method	Result (%)	SD (%)
Shape Features	60.8	26.3
Contour Signatures	59.6	33.4
Elliptic Fourier	57.8	30.7

Table 4.1: Results for the three shape analysis methods.

As can be seen, all three methods performed similarly, with the shape features producing slightly better results, at 60.8%. The shape features also had a lower standard deviation than the other two methods, at 26.3%, although for all three this is quite high. Looking at which species each method performed better than the others on helps to explain this high variance.

Whilst the signatures and the EFD performed similarly on most species, there were several species on which they both performed significantly better than the shape features, and others on which the shape features performed much better. Examples of these species can be seen in Figures 4.1 and 4.2 respectively. Species for which the shape features performed best typically had more complex structures, often with high levels of intra-species variation. Many of these have lobed leaves with varying numbers of lobes from leaf to leaf, whereas species for which the number and position of lobes on each leaf remained constant tended to achieve comparable results for all methods. On the other hand, the contour signatures and elliptic Fourier descriptors got better results on leaves with simpler oval shapes, where the intra-species variance is less, but so too is the inter-species variance. The reason for this appears to be that it captures the general properties of the shape, and is so more resilient to these slight variations, whereas the other methods more precisely capture the more subtle details required for distinguishing between species with similar-shaped leaves.

Although the results for all three methods were relatively low on a dataset of this size (100 species), when formulated as a retrieval problem, where the

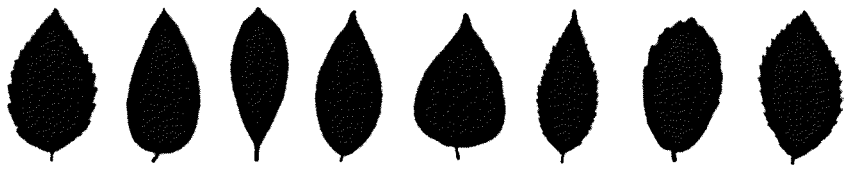


Figure 4.1: Examples of species for which Elliptic Fourier and Contour Signatures are more likely to achieve correct classification than the Shape Features.

top n species are returned, shape-based methods can be seen as an effective means of eliminating the majority of species, aiding the accuracy and reducing the computational time required by other methods performed on top of this. In figure 4.3, the y-axis indicates the fraction of cases for which the correct species appears within the number of returned results in the x-axis. For example, for EFD approximately 90% of leaves tested had the correct species appear within the first 5 species retrieved. For shape features, the correct species appears in the first 8 species retrieved in over 99% of cases, whilst for all methods the correct species always appears within the first 14 out of the 100 species. This will be explored further in Chapter 5.

4.2 Leaf Texture

Much of the texture present on a leaf is due to the venation, with other sources of texture including hairs and glands. The veins on leaves typically have a hierarchical structure, and this venation can be separated into two main groups:

Figure 4.2: Examples of species for which Shape Features are more likely to achieve correct classification than Elliptic Fourier and Contour Signatures.

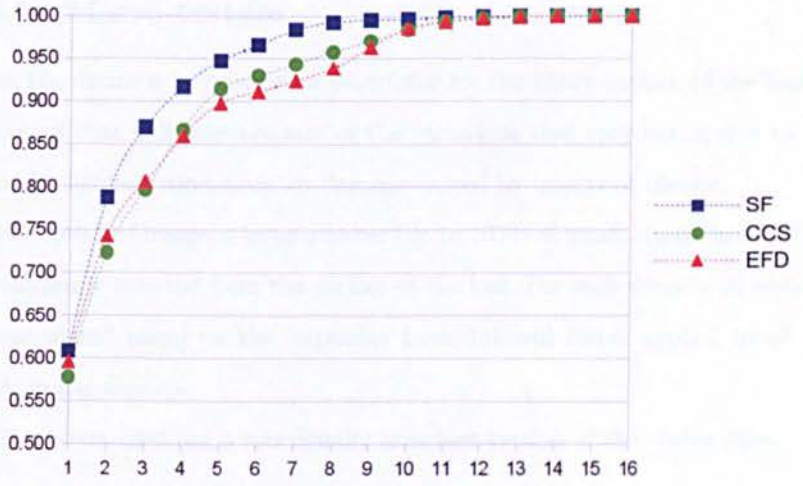


Figure 4.3: Species retrieval results, showing rate at which the correct species is within the top n returned, for shape features (SF), centroid-contour signatures (CCS) and elliptic Fourier descriptors (EFD).

the low-order vein framework, consisting of the larger primary and secondary veins, and the higher-order vein fabric which occupies the spaces in between (see Figure 2.6). Because of this, it is beneficial to consider the texture on both a macro scale and a micro scale. For the former, descriptors are generated that describe the entire surface of a leaf, whilst for the latter, the texture from between the larger veins is extracted.

Another aspect of the texture worth considering is that on many leaves, the lower (abaxial) side and upper (adaxial) side are very different. Typically, the venation is more prominent on the abaxial side, with hairs and glands also being common features here, whilst the adaxial side is more likely to have a waxy texture (usually to prevent excessive water loss).

4.2.1 Macro-texture

Here, the desire is to generate a descriptor for the entire surface of the leaf, in a manner that will take account of the variations that may occur, due to, for example, lighting conditions, or damage caused by insects or disease.

For each leaf image, a large number (up to 1024) of small, fixed-size windows are randomly selected from the surface of the leaf. For each window 20 features are calculated based on the responses from different filters applied to all the pixels in the window.

The filters used are a rotationally invariant version of the Gabor filter:

$$g(x, y) = \exp \frac{r^2}{2\sigma^2} \cos \frac{2\pi r}{\lambda}$$

where $r = \sqrt{x^2 + y^2}$ is the distance from the centre of the filter, σ is the standard deviation, and λ is the wavelength, set to be $\lambda = 3\sigma$. Five different scale filters are used, produced by varying σ. The wavelength is fixed in relation to σ so that the filters are scaled versions of each other. Each filter is convolved with the window and four features are then calculated for that filter for the window:

1. Average positive value

$$\sum_{\substack{(i,j)\in W \\ s_j \geq 0}} \frac{f_{ij}}{|W|}$$

2. Average negative value

$$\sum_{\substack{(i,j)\in W \\ s_j \leq 0}} \frac{f_{ij}}{|W|}$$

3. Energy

$$\sum_{(i,j)\in W} \frac{{f_{ij}}^2}{|W|}$$

4. Entropy

$$-\sum_{(i,j)\in W}\frac{|f_{ij}|}{|W|}\log\frac{|f_{ij}|}{|W|}$$

Where W is the current window, f_{ij} is the response for the current filter at pixel (i,j), and $|W|$ is the size of the window.

Further details and analysis of the data generated here is performed in Section 5.1.1, including discussion of parameters used, and techniques for classifying leaves based on this data.

4.2.2 Micro-texture

When working at a high enough resolution to be able to extract useful information from the vein fabric, it is not practical, due to computational requirements, to cover the entire leaf surface. Instead, a selection of sample windows should be chosen. If texture samples (windows) are extracted randomly from a leaf, the level and quality of the vein framework present in a sample may vary greatly, depending on the precise position of the sample on the leaf. For this reason, a simple method is suggested for extracting samples which as far as possible contain only the vein fabric, as the contents of these samples should be more consistent (Figure 4.4).

The first stage is to reduce the scale of the image by convolving it with a Gaussian kernel and then sub-sampling. This has the effect of smoothing out much of the detail in the vein fabric, whilst retaining the main venation. Next, the image background, the paper on which the leaf is mounted, is removed. This can be done using Otsu's thresholding method [102]. An edge detection operator is then applied to the foreground of the image to provide a rough measure of the areas with strong edges in this scale space. A large number of potential windows (10 times the number we intend to use) are sampled at random from the foreground (containing only the leaf) and are sorted according to the sum of

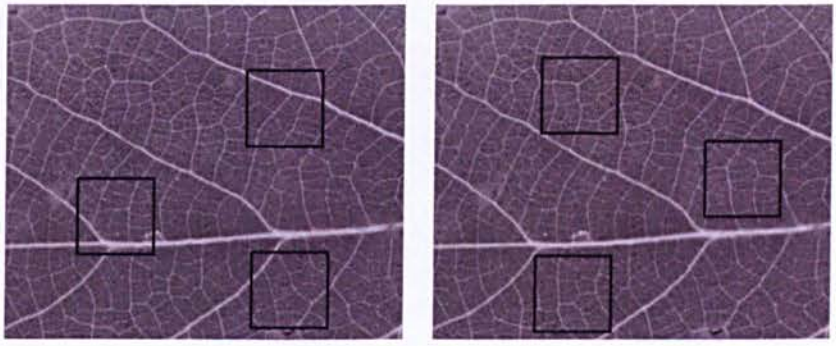


Figure 4.4: Random sampling (left) compared with desired sampling (right), shown on an x-ray image for increased contrast.

the squared edge magnitude for all the pixels within the window. The desired number of non-overlapping sample windows (in this case, 8 were used) with the lowest sum can then be selected for use. A number of examples of windows selected by this method are given in Figure 4.5.

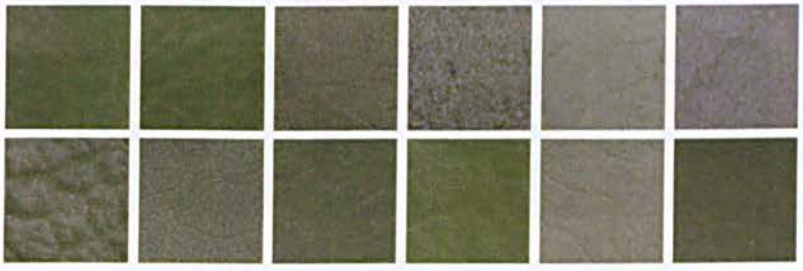


Figure 4.5: Extracted texture samples from 12 species of Quercus (Oak).

4.2.2.1 Gabor Filters

The texture analysis method presented here is based around the joint distributions of responses to Gabor filters. A Gabor filter [30] is essentially a sinusoid modulated by a Gaussian function. It can be expressed as follows:

$$G(x,y) = \exp(\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}) \cos(\frac{2\pi x'}{\lambda} + \psi), \tag{4.1}$$

where:

- $x' = x\cos\theta + y\sin\theta$
- $y' = y\cos\theta - x\sin\theta$
- θ is the orientation of the filter.
- γ is the filter aspect ratio.
- σ is the standard deviation of the Gaussian.
- λ is the wavelength of the sinusoid.
- ψ is phase offset.

Gabor filters have been applied to a large range of computer vision problems including image segmentation [124] and face detection [59]. Of particular interest are the links found between Gabor filters and the human visual system [29].

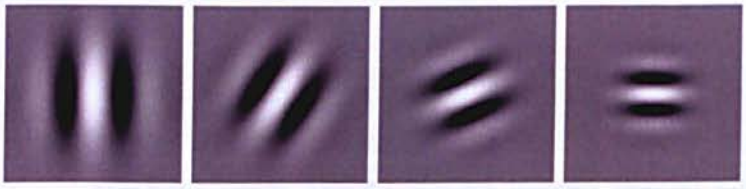


Figure 4.6: Examples of the Gabor filters used here.

4.2.2.2 Texture Analysis From Gabor Co-Occurrences

A bank of 128 Gabor filters is created, where for filter G_{mn}, $\sigma = 1.5 * 1.2^{m-1}$, $\lambda = \frac{\sigma\pi}{2}$ and $\theta = \frac{n\pi}{16}$, with $m \in [0,7]$ and $n \in [0,15]$ referring to the filter scale and angle respectively. For all filters, $\gamma = 1$ and $\psi = 0$. The full set of filters is applied to each texture, but for each scale only the value corresponding to the highest absolute response for all the orientations is recorded for each pixel. This ensures that the method is rotation invariant.

The results of the filtering for an image are combined into a series of co-occurrence matrices [52], whereby for each pair of scales, the resulting matrix describes the probability of a pixel producing one response value for the first scale, and another for the second.

$$C_{kl}(i,j) = P(g_k(x,y) = i, g_l(x,y) = j) \tag{4.2}$$

where $g_m(x,y) = \max_{n=0..15}(G_{mn}(x,y) * I(x,y))$ is the maximum response from convolving the filters for scale m with the image I at point (x,y), and (i,j) is a pair of response values. Examples of these matrices for values $k = 0$ and $l = 1to7$ are given in Table 4.2. The x-axis and y-axis for each matrix cover the range of response values for each of the two filters, with the greyscale value representing the frequency at which the two filters gave the corresponding pair of responses.

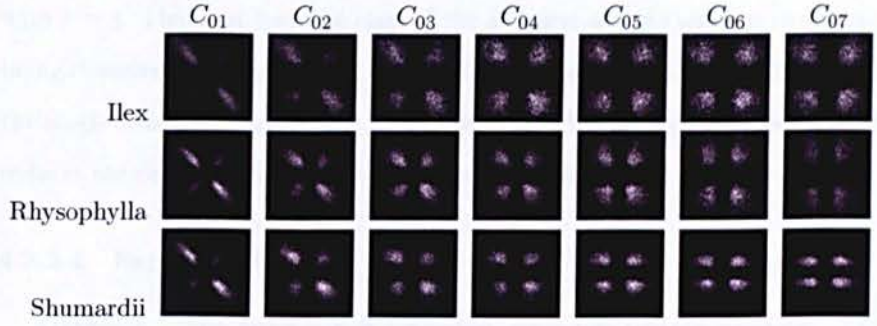


Table 4.2: Examples of Gabor co-occurrence matrices for 3 species of Quercus

4.2.2.3 Classifying Micro-Textures

To classify textures, the corresponding co-occurrence matrices for different textures are compared directly. This is done by treating the co-occurrence matrices

as probability distribution functions (pdfs), by simply dividing each value by the sum of all values, and using the Jeffery-divergence distance measure – the symmetric version of the Kullback-Leiber divergence [15]. For two pdfs, f_a and f_b, the distance between them, $JD(f_a, f_b)$, is calculated as follows:

$$JD(f_a, f_b) = \sum_i \sum_j f_a(i,j) log \frac{2f_a(i,j)}{f_a(i,j) + f_b(i,j)} + f_b(i,j) log \frac{2f_b(i,j)}{f_a(i,j) + f_b(i,j)} \tag{4.3}$$

The distance between two images A and B is then:

$$D(A,B) = \sum_k \sum_{l,l \neq k} JD(C^A_{kl}, C^B_{kl}) \tag{4.4}$$

Where C^A_{kl} and C^B_{kl} are respectively the co-occurrence matrices at scale k,l for images A and B.

The final classification is performed using the the k-nearest neighbour method, with $k = 3$. The most frequent class of the 3 closest texture samples to the one being classified is chosen. In the case that all 3 classes are different, the class of the single closest texture sample is used instead. This strategy was chosen as it reduces the risk of classification errors due to outliers.

4.2.2.4 Experiments

Datasets. The method was evaluated using two texture datasets. The first dataset was constructed using the method described previously. For each of 8 leaves from 32 different species, 8 64 × 64 windows were selected. This window size was chosen as it was found to be small enough to allow the windows from leaves with dense vein frameworks to fit between the main veins. Eight windows were then used to provide an adequate overall sample size, whilst more would require more computation and may not be possible for particularly small

leaves without significant overlap between windows. Each of the 8 samples for a leaf was filtered, using the set of Gabor filters, before they were combined into a single set of co-occurrence matrices. The second dataset used 8 windows sampled at random from the same leaves, to illustrate the value of the texture extraction method for selecting suitable windows.

Comparison Methods. For comparison, the above datasets were also used with a number of traditional texture analysis methods:

- Fourier Coefficients:

  The Fourier Transform of each window was calculated. From this, a vector of 64 features was found, whereby the i^{th} feature $f_i = \sum_{\theta=0}^{\pi} |F(i\frac{w}{64}, \theta)|$, where $F(r, \theta)$ is the Fourier Transform in polar form, and w is the half the image width [66].

- Gabor Filters:

  The same set of Gabor filters used in Section 4.2.2.2 is applied to each image. The energy in each resulting image is then calculated as $e_{\sigma\theta} = \sum_x \sum_y (G_{mn}(x, y) * I(x, y))^2$. The set of energies for each scale are then averaged resulting in 8 rotationally invariant features.

- Co-occurrence Matrices:

  The traditional co-occurrence matrices were produced, using angles of 0 rad,$\frac{\pi}{4}$ rad,$\frac{\pi}{2}$ rad and $\frac{3\pi}{4}$ rad and distances of 1,2 and 3. For each distance, a set of 14 textural features is calculated, as described by Haralick [52].

4.2.2.5 Results

The results for the experiments are given in Table 4.3, with values representing the percentage of leaves correctly classified. All the algorithms performed better

on the dataset created using the method in Section 4.2.2 than on the dataset of randomly selected windows, showing the value of this method of leaf texture extraction. For all datasets the new method performed best, with the basic Gabor method performing worse. The improvement between the two datasets was greatest for the Fourier method, suggesting that it is better at capturing finer detail presented in the vein fabric, however its performance was still not able to match the proposed method.

	Vein Fabric	Random Windows
Gabor Co-occurences	85.16	79.69
Gabor	50.78	45.70
Fourier	82.42	62.89
Co-occurrence Matrices	69.14	61.72

Table 4.3: Results for the two datasets (%)

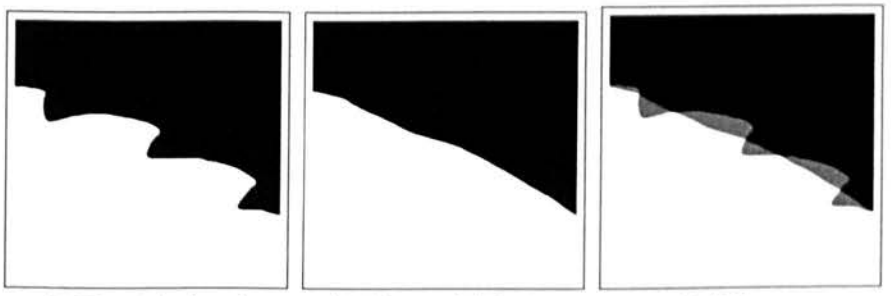
4.3 Margin Characteristics

Study of the leaf margin – the pattern around the edge of the leaf including details such as the teeth (see Figure 2.8 for examples) – provides valuable information about climate and other conditions in which a plant species has evolved. It is therefore important to be able to accurately extract the margin from the leaf, independently of the shape, and allow for meaningful comparison between different leaves' margins.

4.3.1 Extracting The Margin

The first step is to extract the leaf's margin. Having extracted a mask of the leaf, a modal filter is applied to the mask to acquire a smoothed version of the leaf's shape (see Figure 4.7). This filter sets each pixel of the smoothed leaf to be part of the leaf if the majority of the original pixels within the filter's radius

were part of the leaf, and to be part of the background otherwise. This has the effect of removing the tips of the teeth and filling in any small gaps. By varying the filter's radius, different levels of smoothing can be achieved, with sufficiently large radii remove all detail from the margin. Here, a radius of 15 pixels was used. From this, m evenly spaced points around the contour are calculated (in this case, $m = 8192$ was used), encompassing the entire outline. For each of these points, a corresponding point on the original outline is calculated. This is done by first estimating the line that is normal to the edge of the leaf at this point as being perpendicular to the line which runs between the two points at distance k either side of the current point. The sub-pixel point at which this line intersects the original leaf's outline is then found by linear interpolation. The distance between this point and the current point is then calculated (see Figure 4.8), and these distances for all the points in the smoothed outline are combined in order to produce a margin signature, $\mathbf{s} = \langle s_1, ..., s_m \rangle$. Examples of extracted margin signatures are given in Figure 4.9.



(a) The original mask (b) After modal filtering (c) Difference

Figure 4.7: Using modal filters to extract the teeth.

The extracted margin is partitioned into n overlapping windows, $\mathbf{x} = \langle x_1, ..., x_n \rangle$, of equal size and spacing (in this case $n = \frac{m}{8}$ and the window size used is $\frac{m}{128}$). This is done for a number of reasons. Firstly, the exact number of teeth will vary between leaves of the same species, which may cause problems when

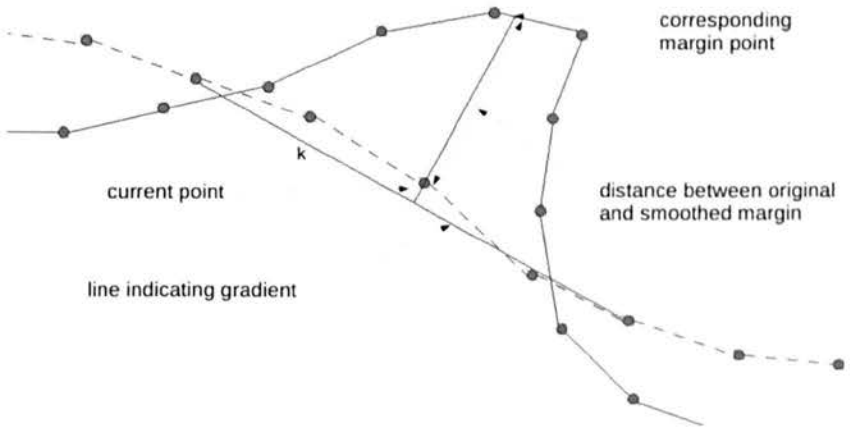


Figure 4.8: Calculating a point on the margin signature.

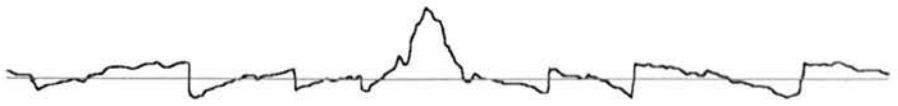
attempting to align their margins. By using windows, each window will provide a description for the area of the margin within it, and so be more robust to this variation. Furthermore, this allows for an adequate description of the margin whilst using a much smaller number of datapoints, and so reducing computation time to as little as one-eighth. By overlapping the windows, sensitivity to their exact position in $\mathbf{s}$ is reduced.

For each point within a window, x_i, 3 values are calculated:

1. Magnitude - This is the signed distance between the smoothed margin point and its corresponding point in the original margin, where the sign is determined by whether the original margin point lines inside or outside of the smoothed margin.

2. Gradient - The signed difference between the current point in the margin signature and the next point.

3. Curvature - The angle at the current point between the previous point and the next point in the signature.



(a) Acer circinatum

(b) Populus grandidentata

(c) Quercus cerris

Figure 4.9: Examples of segments from extracted margin signatures. The y-axis represents the distance from the smoothed margin and the original margin.

For each of these, 2 features are then calculated for the window, giving a total of 6 features per window:

- Average positive value:

$$\sum_{\substack{s_j \in x_i \\ s_j \geq 0}} \frac{s_j}{|x_i|}$$

- Average negative value:

$$\sum_{\substack{s_j \in x_i \\ s_j \leq 0}} \frac{s_j}{|x_i|}$$

Where x_i is the current window, s_j is the value at a point within the signature, and $|x_i|$ is the size of the window.

4.4 Locating The Apex And Insertion Point

The only two landmarks which consistently exist are the 'insertion point' - where the petiole, or stem, attaches to the leaf - and the apex - the tip of the leaf (see Figure 2.2). It would therefore be useful to be able to locate these two points. To do this, the extracted margins can be utilised.

To locate these two points, potential candidate points are first identified by selecting the local maxima from the margin signature which have an absolute magnitude greater than 25% of the global maximum. This value of 25% was used as it was found to small enough so that, for all leaves in the dataset, the true apex and insertion points were selected amongst the candidates. Based on the principle that both sides of a leaf - from insertion point to apex - will be approximately a reflection of each other, dynamic time warping can be used to identify the two points on the margin for which the difference between the two sides is minimised, and so are most likely to be the insertion point and apex.

4.4.1 Dynamic Time Warping

Dynamic time warping (hereon DTW) [122] is a technique for measuring the similarity between two different sequences. During the comparison, it allows parts of the signals to be stretched or compressed to a certain extent, thereby accounting for the sequences being of different lengths (for instance, due to differences in speed) or containing natural distortions. A typical application for DTW is speech recognition, where people may speak at different speeds, or elongate different sounds. It has also seen use for a number of computer vision problems, including face detection [136] and action recognition [126].

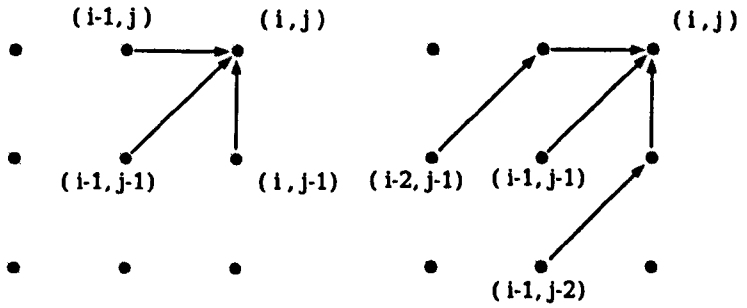
Given two sequences $\mathbf{x} = \langle x_1, ..., x_m \rangle$, $\mathbf{y} = \langle y_1, ..., y_n \rangle$ an $m \times n$ cost matrix $\mathbf{C}$ is calculated, whereby value c_{ij} is the distance between points x_i and y_j. Under the assumption that point x_1 corresponds to point y_1 (i.e. the same starting

points), and x_m to y_n, a monotonic path through $\mathbf{C}$ is found, beginning at c_{00} and ending at c_{mn}, such that the sum of the values at the nodes visited is minimized. This path then represents the optimal alignment of points in $\mathbf{x}$ to those in $\mathbf{y}$. This can be calculated relatively efficiently (quadratic complexity) by recursively accumulating the costs in a matrix $\mathbf{D}$. The value d_{ij} is calculated as follows:

$$d_{ij} = \begin{cases} c_{ij} & \text{if } (i = 1 \wedge j = 1) \\ \infty & \text{if } (i = 1 \wedge j > 1) \vee (j = 1 \wedge i > 1) \\ c_{ij} + \min \begin{pmatrix} d_{i-1,j} \\ d_{i,j-1} \\ d_{i-1,j-1} \end{pmatrix} & \text{otherwise} \end{cases} \tag{4.5}$$

Once all the values in $\mathbf{D}$ have been calculated, the measure of the similarity between the two sequences is given by d_{mn}.

There are a number of extensions to the standard DTW algorithm that have been proposed in the literature [122, 123]. Calculating d_{ij} by using Equation (4.5) results in a path which travels monotonically between adjacent cells, either horizontally, vertically or diagonally. Since a continued horizontal or vertical movement represents the compression of a subsequence to unit length, or the stretching of a single point to a much longer length, this could result in unrealistic distortions. To counter this we add the condition that every step that is made horizontally or vertically must also be accompanied by a diagonal step (see Figure 4.10). This restricts the maximum distortion of a subsequence to a level that is realistic for this type of data, whilst ensuring that distortions carry an additional cost due to resulting in longer paths.



(a) The standard formulation (b) Step functions used here

Figure 4.10: Legal steps for path when calculating DTW.

If two sequences are similar, the optimal DTW path will be close to the diagonal of the cost matrix, where $i = j$. If the optimal path diverges from this by more than a certain amount, it is unlikely that the two sequences are from the same class. This allows a constraint to be added to improve the speed of the algorithm. By only calculating d_{ij} for $|i-j| \leq k$, the complexity can be reduced from $\mathcal{O}(n^2)$ to $\mathcal{O}(kn)$ where $k \ll n$, without risking finding a sub-optimal path, when the two sequences are from the same class [123]. With these improvements included, the equation for calculating d_{ij} becomes:

$$d_{ij} = \begin{cases} \infty & \text{if } |i-j| > k \\ c_{ij} & \text{if } (i = 1 \wedge j = 1) \\ \infty & \text{if } (i = 1 \wedge j > 1) \vee (j = 1 \wedge i > 1) \\ c_{ij} + \min \begin{pmatrix} d_{i-2,j-1} + c_{i-1,j}, \\ d_{i-1,j-2} + c_{i,j-1}, \\ d_{i-1,j-1} \end{pmatrix} & \text{otherwise} \end{cases} \tag{4.6}$$

4.4.2 Finding The Points Of Margin Symmetry

For a given candidate point, the corresponding window, x_i is identified from the circular sequence of windows $\mathbf{x} = \langle x_1, ..., x_n \rangle$ for the leaf. Two sequences, $\mathbf{a} = \langle a_1, ..., a_{\frac{n}{2}+w} \rangle$, $\mathbf{b} = \langle b_1, ..., b_{\frac{n}{2}+w} \rangle$ are generated, where $a_1 = b_1 = x_i$, $a_j = x_{i+j}$, $b_k = x_{i-k}$ and n is the total number of windows. Since the insertion point and apex may not lie directly opposite each other, the sequences **a** and **b** are continued for a distance of w beyond the mid-point $x_{i+\frac{n}{2}}$, such that the ends of the sequences are overlapping. A value of $w = \frac{n}{8}$ was used.

The accumulated cost matrix **D** is generated as described in Section 4.4.1. Because the last w points in the two sequences are the reverse of each other, similarity is calculated as the minimum d_{jk} where $j+k = n$. Using this method, the insertion point and apex were correctly identified in 97.75% of test cases, with one or the other being correctly found in 99.25% of the 1600 leaves in the dataset.

These apices and insertion points are used in Section 5.1.2 for performing classification of leaf based on the above leaf margin descriptors.

4.5 Venation Patterns

In this section a couple of techniques are presented for the extraction of leaf venation. The first method uses a genetic algorithm to evolve classifiers for detecting veins on a pixel-by-pixel basis, whilst the second utilizes an ant colony algorithm to try to extract continuous vein segments.

4.5.1 Extraction By Evolved Vein Classifiers

4.5.1.1 Classifying The Vein Pixels

A genetic algorithm is used to evolve a set of classifiers for detecting vein pixels. Each classifier consists of a pair of bounds for each of the features used. If the values of all the features for a pixel fall within all the bounds for a classifier, then it is classified as vein. The vein pixels found by all the classifiers in the set are combined, and all other pixels are classified as non-vein. These classifiers are similar to those used by Liu & Tang [84]. More specifically, the set of vein pixels, V, is determined as follows:

$$V = \{(x,y)|0 \leq x < w, 0 \leq y < h,$$
$$\exists c \in C|\forall f_i \in F_{xy}, c_{i0} \leq f_i \leq c_{i1}\}$$

where

- w,h are the image height and width respectively
- C is the set of all classifiers
- c_{i0} is the lower bound for the i^{th} feature for the classifier c
- c_{i1} is the upper bound for the i^{th} feature for the classifier c
- F_{xy} is the set of feature values for the pixel at (x,y)
- f_i is the value for the i^{th} feature

4.5.1.2 Feature Extraction

A set of 9 features are extracted for each pixel for use in classification. The features used are as follows:

1. Pixel greyscale value $f_1 = I(x, y)$.

2. Edge gradient magnitude (from Sobel), f_2.

3. Average of greyscale values in a 7×7 neighbourhood,

$$f_3 = \frac{1}{49} \sum_{\substack{x-3 \leq i \leq x+3 \\ y-3 \leq j \leq y+3}} I(i, j).$$

4. Greyscale value minus neighbourhood average,

$$f_4 = I(x, y) - \frac{1}{49} \sum_{\substack{x-3 \leq i \leq x+3 \\ y-3 \leq j \leq y+3}} I(i, j).$$

5. Greyscale value minus leaf lamina average,

$$f_5 = I(x, y) - \frac{1}{|lamina|} \sum_{\substack{0 \leq i < width \\ 0 \leq j < height \\ (i,j) \in lamina}} I(i, j).$$

   Where *lamina* is the set of all pixels which are part of the leaf's lamina, found by using Otsu's thresholding [102] to remove the leaf from the background.

The average local gradient direction of pixels in a 11×11 neighbourhood around the current pixel is calculated. This size neighbourhood was chosen because for most vein pixels this will include both sides of the vein. The greyscale values of the points 5 pixels from the current one in both directions along the gradient and perpendicular to the gradient are calculated. If the current pixel is part of a vein, the pixels perpendicular to the gradient direction are likely to also be vein pixels, and so similar to the current pixel, whilst the pixels along

the gradient direction are likely to be non-vein, and therefore quite different.

$$i_1 = I(x + 5sin(\alpha), y + 5cos(\alpha))$$

$$i_2 = I(x - 5sin(\alpha), y - 5cos(\alpha))$$

$$j_1 = I(x + 5sin(\alpha + \frac{\pi}{2}), y + 5cos(\alpha + \frac{\pi}{2}))$$

$$j_2 = I(x - 5sin(\alpha + \frac{\pi}{2}), y - 5cos(\alpha + \frac{\pi}{2}))$$

Where α is the gradient direction.

The remaining features are then:

6. The absolute difference between pixels, i_1 and i_2, either side of potential vein $f_6 = |i_1 - i_2|$

7. The absolute difference between pixelsj_1 and j_2, along potential vein

$$f_7 = |j_1 - j_2|$$

8. Greyscale value minus average value of the two pixels either side of the potential vein

$$f_8 = I(x, y) - \frac{i_1 + i_2}{2}$$

9. Greyscale value minus average value of the two pixels along the potential vein

$$f_9 = I(x, y) - \frac{j_1 + j_2}{2}$$

To allow the same genetic operators to be used on features with very varied distributions, the feature values for the training data are mapped to a uniform distribution. This mapping is recorded and applied to any data being subsequently classified.

4.5.1.3 Evolving The Classifiers

Classifiers are evolved one after another using a genetic algorithm, and added to the classifier set until no more classifiers with a fitness above a certain threshold can be generated within a maximum number of iterations. The only genetic operators used are mutations, as crossover operations are likely to combine classifiers that work on different types of vein pixels, thereby having a negative effect. For example, a classifier that finds thin sections of vein may require higher edge gradient values and lower greyscale values than a classifier finding the pixels in the middle of thicker veins. Crossing over these two classifiers would result in ones which classified neither of these vein pixel types. Bounds are mutated with probability 0.3 by adding or subtracting an amount randomly drawn from the range [0,0.01]. The population is re-initialised after each classifier is added to the set. Each individual is initialised by centring the bounds around the feature values for a vein pixel randomly selected from the training data, with the width of the bounds drawn from a Gaussian distribution. This increases the likelihood of the classifier being effective, as one vein pixel will always be correctly classified by it, along with any similar vein pixels.

The fitness function used is as follows:

$$fitness_i = \frac{|T_i \setminus \bigcup_{j \in C} T_j|}{|F_i \setminus \bigcup_{j \in C} F_j| + k},$$

where:

- T_i is the set of vein pixels correctly classified by classifier i (true positives).
- F_i is the set of non-vein pixels incorrectly classified by classifier i (false positives).
- C is the current set of classifiers selected in previous iterations and k is a constant.

This function grants high fitnesses to individuals which, if added to the classifier set, would significantly increase the number of true positives, but not the number of false positives. The fitness of a classifier is therefore dependent on the order in which they are selected. The constant k is used to adjust the balance between a high true positive/false positive ratio, and a high total number of true positives. If k is set too low the ratio will be very high, but the final classifier set may over-fit the training data. If k is set too high it will result in a high number of false positives. A value of $k = 5$ was found to be appropriate.

4.5.1.4 Redundancy

Classifiers can potentially be made redundant by other classifiers added to set later. In other words, a classifier may no longer uniquely classify many vein pixels whilst still incorrectly classifying some non-vein pixels. It is beneficial to remove such classifiers as this may greatly reduce the number of false positives whilst only slightly reducing the number of true positives.

Redundant classifiers are identified by removing candidates from the set and measuring any improvement in overall classification quality. The classifier whose removal produces the largest increase in quality is permanently removed from the set. This process is repeated until no more classifiers are found to be redundant.

4.5.1.5 Results

The classifier was trained using 8000 pixels manually selected from 14 leaf images, 2 from each of 7 species. These pixels were then manually labelled as either vein or non-vein. The resulting classifier was then tested on 7 new leaf images, one from each of the species used for training. The ROC curve in Figure 4.11 shows the results (solid line). With a false positive rate of 0.0166, a true positive rate of 0.853 was achieved.

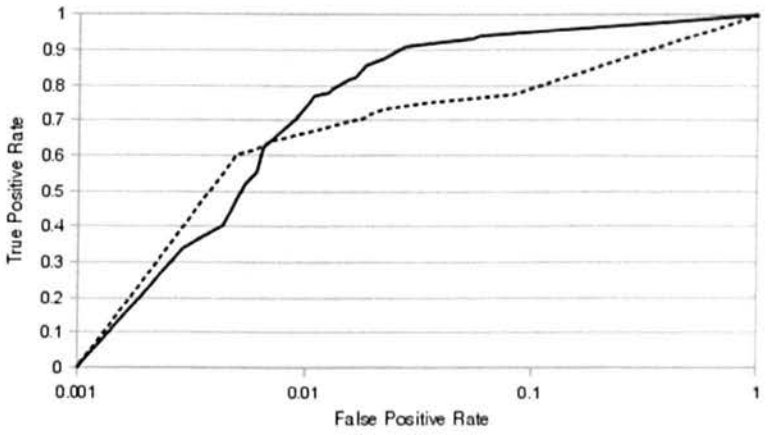


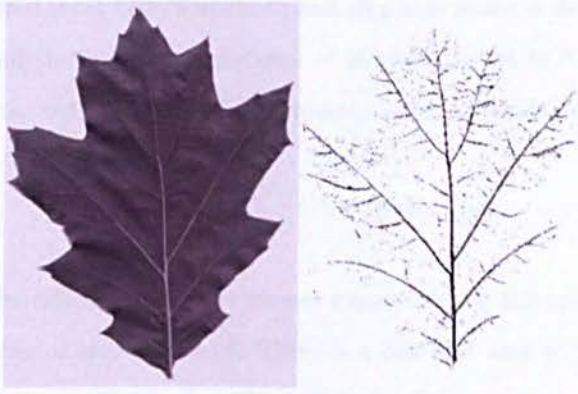
Figure 4.11: ROC Curve. Solid line - evolved classifiers. Dashed line - ant algorithm.

The classifier was also used on the full leaf images from the test set, in order to extract the full venation pattern. Examples of these results are shown in Figure 4.12.

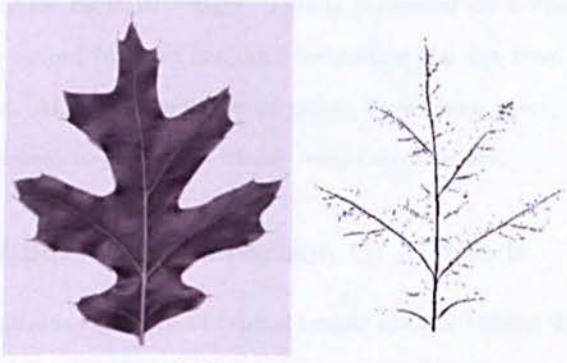
4.5.2 Extraction By Ant Colonies

The second approach to vein extraction is to use an ant colony algorithm. A population of ant-like agents are placed at random across the image. These "ants" then move across the image, moving from pixel to pixel based upon some heuristic evaluation of that pixel, known as the pixel's visibility, and also based on the level of "pheromone" at that pixel. The pheromones are an indicator deposited by ants to signal to other ants the value of a particular pixel. As time progresses, the pheromone levels build up to create a pheromone map for the image, with high levels in desirable regions, and low levels in undesirable regions. In this case, the edge magnitude is used as the measure of visibility, to encourage the ants to traverse along the veins, thereby extracting continuous sections of venation.

(a) Quercus Shumardii



(b) Quercus Rubra



(c) Quercus Ellipsoidalis

Figure 4.12: Results for extraction by evolved classifers

The probability, P_{ij}, of an ant at pixel i moving to pixel j is calculated as follows:

$$P_{ij} = \begin{cases} \dfrac{\tau_j^\alpha \eta_j^\beta}{\sum_{k \in K_i} \tau_k^\alpha \eta_k^\beta} & \text{if } j \in K_i \\ 0 & \text{otherwise} \end{cases}$$

Where τ_j and η_j are the pheromone level and visibility respectively at pixel j, α and β are the weightings for these two components, and K_i is the set of pixels neighbouring pixel i. To prevent the ants converging on the strong edges outlining the leaf instead of the venation, the visibility for all background pixels (again calculated using Otsu's method) and all pixels within a short distance of the background (in this case, a distance of 10 pixels) is set to 0. After all the ants have performed one move, the pheromone levels are updated:

$$\tau_{i+1} = (1 - \rho)\tau_i + \delta a_i \eta_i$$

Where ρ is the rate at which pheromones evaporate, δ is the update rate, and a_i is the number of ants at pixel i. There is a risk that ants will simply move between the same small set of pixels, building up pheromone levels until it is highly unlikely for them to escape. This is prevented by keeping a list of the last 10 pixels visited by each ant, and forbidding the ant from re-visiting any of these pixels. After a set number of moves have taken place, the pheromone map is thresholded to produce a binary vein classification.

4.5.3 Results And Comparison Of Methods

Figure 4.13 contains examples of typical results obtained using this method. For each leaf the algorithm was run for 500 steps, using 2000 ants. The pheromone map was then thresholded at 2% of the maximum pheromone level. These values were chosen as they appeared to give the best qualitative results. The results

differ from those obtained using the evolved classifiers in a number of ways. Firstly, due to the use of only the edge gradients to guide the ants across the image, the results contained only the hollow outline of the venation, whereas the other method extracts the full vein. One advantage of using ants is that it helps in extracting continuous venation, whilst the evolved classifiers extract veins with many small gaps in them. On the downside, when a vein contains a section with only a low edge magnitude, the ants are unable to continue to extract the rest of that vein as the pixel-by-pixel evolved classifiers are able to do. The effects of this can be seen near the top of the first image in Figure 4.13, where a large section of venation is completely absent. Furthermore, whereas much of the false positive results from the first method are isolated pixels that can be easily removed, the ants produce larger, connected areas of noise, that may be harder to distinguish from the actual venation.

Figure 4.13: Results using the ant colony algorithm.

By applying morphological closing, the hollow vein centres can be filled in (Figure 4.14). From these, quantitative results can be calculated, as shown in Figure 4.11 (dashed line). It can be see that the ant algorithm still performs worse than the evolved classifiers, except when the true positive rate falls below approximately 0.63.

Figure 4.14: Results after morphological closing

4.6 Summary

This chapter has provided techniques for the extraction of many of the key components of the plant leaves (primarily the venation, margins and texture), providing appropriate descriptors which can be used in the automated comparison and classification of species. Further to this, a comparative study has been carried out of the most popular techniques used for the analysis of leaf shape, demonstrating that for the purposes of classification, both elliptic Fourier descriptors and shape-features perform well, dependent on the characteristics of the leaf.

Chapter 5

Machine Learning for Plant Leaf Analysis

Chapter contributions:

- Methods for classification of the leaf lamina and margins that incorporate intra-species variance to improve results.
- Probabilistic classification combining multiple leaf feature-sets.
- A technique for the automatic selection of which feature-sets to use, on a leaf-by-leaf basis.

5.1 Incorporating Intra-Species Variation into Plant Classification

One of the key challenges to automated analysis of plant leaves lies in the large amount of possible variation, even within a single species, as was illustrated in Figure 2.1. As well as the natural variation one can expect from any organic

object, variation in leaves can come from a number of sources. Much variation comes from the age and developmental stage of the leaf. In terms of shape, growth in a young leaf often primarily occurs length-wise, becoming broader later in development. In some lobed species, the lobes may not be apparent until a certain point. Furthermore, some species, such as certain Eucalyptus, may feature different types of leaves on young and mature shoots. As with lobes, margin characteristics, such as teeth, may not develop until the leaf has reached its full size, often appearing first near the apex, then gradually growing further back towards the insertion point. Pigmentation may also alter as the leaf develops.

Another source of variation is from damage that may occur to the leaf, particularily as a result of disease or attack from insects. Disease most commonly affects the surface of the leaf, ranging discolouration to disctinct markings, whilst insect damage often alters the leaf shape, where parts of it have been eaten, but there are many exceptions to both cases.

Much variation can also occur in the image capture process. Lighting conditions can play a large role here. Many leaves feature waxy surfaces which may reflect light differently depending on the relative position of the lighting source, and the amount of light that's allowed to pass through the leaf may affect the visibility of features such as the finest venation. Indeed, some leaf data sets have been created using a specific backlighting system. Camera focus and resolution will also affect the level of texture information available. Moreover, cameras are not the only devices that have been used to capture leaf images, with other examples including flat-bed scanners, x-ray devices and even electron microscopes.

This section explores methods for increasing the reliability of leaf classification by taking account of the intra-species variation that may be present.

5.1.1 Utilizing The Hungarian Algorithm For Improved Classification Of Leaf Laminas

For the leaf macro-texture, data of the type generated in Section 4.2.1 - where the leaf to be classified is described by a distribution of points within a feature space - can be classified using a number of existing methods.

When described using histograms, the difference between two probability density functions (pdfs) can be calculated using bin-by-bin methods, such as the Jeffrey-divergence metric, however these methods encounter problems when the data has a high dimensionality, where a large number of bins makes the calculation expensive, whilst the sparse population of bins produces poor results. The earth mover's distance (EMD) [119] deals with this by using signatures, and provides an accurate and intuitive measurement. These 'signatures' are weighted points within the feature space. This is akin to clustering data points drawn from the distribution, and weighting each cluster centroid by the number of points in the cluster. Another method is to use kernel density estimation [105] to estimate a probability density function using points sampled from a distribution, and then to use this estimation to predict the probability of another sampling of points belonging to the same distribution. More recently, 'bag-of-words' methods have enjoyed increasing usage for this problem, particularly in the guise of 'bag-of-visual-words' [128] for image retrieval.

To overcome the problems inherent to the leaf's macro-texture, presented here are two different methods which utilize information generated in the calculation of the earth mover's distance in order to allow for more robust classification of pdfs, particularly when there is high intra-class variation. The first of these methods combines this with the strengths of the 'bag-of-words' method, whilst the second uses this information more directly in order to try to model the intra-class variation.

5.1.1.1 Background

The Hungarian Algorithm and The Earth Mover's Distance. The earth mover's distance (EMD) [119] is a measure of the difference between two pdfs. The analogy is that, to reform one mound of earth as another, the effort required would depend on the sum of the distances that each unit of dirt must be moved. Whilst bin-by-bin methods only consider the amount of 'earth' in each location, the EMD considers how far it must be moved. There are two forms of pdf descriptions that allow the EMD to be calculated: histogram binning and the aforementioned signatures. Since the binning is analogous to using evenly spaced signatures, only the latter needs to be considered.

Whilst there may be many ways of reforming one pdf into another, the EMD is calculated as being the one that requires the minimum total movement (the sum of the distances that each unit of 'earth' is moved). See Figure 5.1 for an example of how one set of data may be mapped to another in this manner. The standard way of determining this is to model it as the transportation problem – the assignment of sources to destinations subject to a set of transportation costs. There are a number of methods for solving the transportation problem, but by reforming the data so that each signature has an equal weight, it becomes equivalent to the simpler assignment problem, which can be solved using the Hungarian algorithm [74]. Whilst the original Hungarian algorithm was $O(n^4)$, an $O(n^3)$ version has since been found by [35].

The EMD only uses the minimum cost calculated by the Hungarian algorithm, but in the usage here the corresponding mapping between signatures will also be recorded, as it provides not only a measurement of the difference between the pdfs, but also information about in what way they are different. The EMD normally uses the Euclidean distance as the cost of moving 'earth' between two points, but here the squared Euclidean distance is used, as this helps to preserve

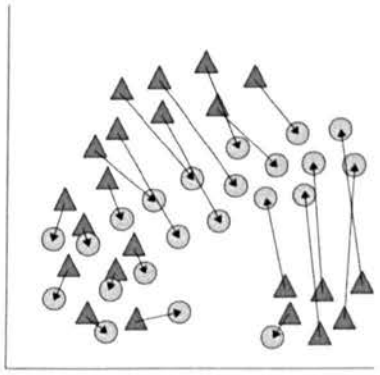
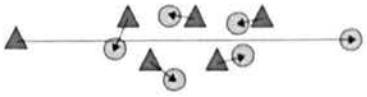
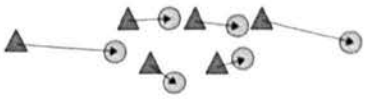


Figure 5.1: The mapping via the Hungarian algorithm between two sets of points.

the topology/ordering of the points (Figure 5.2), since the pairing of points over increasing distances is penalised.



(a) Euclidean distance



(b) Squared Euclidean distance

Figure 5.2: Using the squared Euclidean distance as the cost function preserves the topology.

The Bag-Of-Words Model. The 'bag-of-words' model was originally used for the retrieval of text documents [131]. The idea was to represent documents as the frequency of occurrence of different words, and to find similar documents by comparing these frequencies. In recent years this concept has

been extended to allow for the classification of more general forms of data. Typically, a large number of points are sampled from the training distributions and then a clustering is performed on these. The cluster centroids are used as the 'codewords' in a 'dictionary' used to perform a quantization of the data, by assigning each data-point to its nearest 'codeword'. A set of points from a distribution can then be described as the frequency of occurrence of each 'codeword'. This concept has seen much use recently in the field of computer vision, for tasks such as image retrieval [128, 135, 20] and texture analysis [78, 137].

5.1.1.2 Notation

The problem is defined as follows. A leaf is described by a set of n data points, $X = \{\bar{x}_1, \bar{x}_2, ...\bar{x}_n\}$, windows sampled from the leaf's surface. Each data point $\bar{x}$ is a feature vector, $\bar{x} = [x_1, x_2, ..., x_d]$, where d is the number of features. Given a number of different species, where species i is described by another set of n data points, $C_i = \{\bar{y}_1, \bar{y}_2, ...\bar{y}_n\}$, randomly sampled from all leaves in the training set that belong to the species, the wish is to determine the species to which the leaf described by X most likely belongs. This is calculated using Bayes theorem:

$$c^* = \arg\max_{C_i} P(C_i|X) \tag{5.1}$$

$$= \arg\max_{C_i} P(X|C_i)P(C_i)/P(X) \tag{5.2}$$

$$= \arg\max_{C_i} P(X|C_i)P(C_i) \tag{5.3}$$

The term $P(X)$ in Equation 5.2 is discarded as it is constant for all i.

5.1.1.3 Data-Point-Mapped Bag-Of-Words

The method involves first generating a set of codewords from the training set, suitable for representing the data. All points in the training leaf and species objects are assigned to their nearest codeword. A mapping is calculated between the data points in each training leaf object and its corresponding species object. For each species, the joint distribution is calculated for a training object point assigned to one particular codeword being mapped to species object point that is assigned to a second codeword. That is, for each pair of codewords and each species, the probability is calculated of a mapping having its training object point assigned to the first of these codewords and its species point assigned to the second. For classification, the same codeword assignments and mappings are performed, and the previously calculated probabilities are used to determine the species which the leaf belongs to.

Generating A Dictionary. Within the literature there has been much discussion on the appropriate methods for generating, and ideal size of, the codeword dictionary [127, 67, 69]. The simplest approach is choose points evenly distributed throughout the feature space. The main disadvantage of this is that large portions of the space may not be used, resulting in redundant codewords, whilst other, more useful areas may receive inadequate representation. Another simple method is to use randomly selected points from the training data as the codewords. This largely eradicates the above problems, although using the centroids from a clustering performed on the training data normally provides a better representation. Another approach is to perform a separate clustering for each class and combining the generated codewords. This ensures that each class has some appropriate codewords, but may result in very similar codewords in the combined dictionary. It was found that a k-means clustering of the whole training set produces an appropriate dictionary for this method.

There is no consensus on the size of a dictionary, with suggestions varying greatly, but for this method it was found that, with objects described using 1024 points, a dictionary of size 256 produced good results, with larger dictionaries providing little or no improvement. D_i is the i^{th} codeword in the dictionary.

Producing The Class Models. For each species, a species object is produced by randomly selecting n points from the species's examples in the training set. For each training leaf, a mapping is found from its data points to those of its species object using the Hungarian algorithm. This mapping pairs the points in one object to those in the other, such that the sum of the squared Euclidean distances between paired points is minimised (see Figure 5.1). The point in the species object C_i to which point $\bar{x}$ is paired is defined as $M(\bar{x}, C_i)$.

Each point in the training data is assigned to its nearest codeword. For each species i, for each pair of codewords, (D_a, D_b), the conditional probability is calculated of a point $\bar{x}$ in that species's training data being assigned to codeword D_a, given that the corresponding point in the species object has been assigned to D_b. This is calculated as follows:

$$P(W(\bar{x}) = D_a | W(M(\bar{x}, C_i)) = D_b) \tag{5.4}$$

$$= \frac{P(W(\bar{x}) = D_a, W(M(\bar{x}, C_i)) = D_b)}{P(W(M(\bar{x}, C_i)) = D_b)} \tag{5.5}$$

where

$$P(W(\bar{x}) = D_a, W(M(\bar{x}, C_i)) = D_b) = \sum_{\substack{W(T_{ij}) = D_a \\ W(M(T_{ij}, C_i)) = D_b}} \frac{1}{|T_i|} \tag{5.6}$$

$$P(W(M(\bar{x}, C_i)) = D_b) = \sum_{d=0}^{|D|} P(W(\bar{x}) = D_d, W(M(\bar{x}, C_i)) = D_b) \qquad (5.7)$$

where T_{ij} is the j^{th} point and $|T_i|$ is the total number of points in the training data for species i, $|D|$ is the number of codewords, and $W(\bar{x}) = D_a$ indicates that point $\bar{x}$ has been assigned to codeword D_a (likewise, $W(M(\bar{x}, C_i)) = D_b$ indicates that the point which $\bar{x}$ is paired with is assigned to codeword D_b).

Equation (5.6) calculates the probability of a point in D_a being mapped to a point in D_b as the fraction of training points for a species C_i for which this occurs. The probability of a point, from any codeword, being mapped to one in D_b is then the sum of these for all codewords (Equation (5.7)).

Performing The Classification. To classify a leaf, again all of the leaf's data points are assigned to determine their nearest codewords. The object is mapped using the Hungarian algorithm to each of the species objects. The species to which the leaf most likely belongs can then be determined using a Bayesian classifier.

$$c^* = \arg\max_{C_i} P(X|C_i)P(C_i) \qquad (5.8)$$

$$P(C_i) = \frac{|T_i|}{\sum_j |T_j|} \qquad (5.9)$$

$$P(X|C_i) = \prod_{W(\bar{x})=X} P(W(\bar{x}) = D_a | W(M(\bar{x}, C_i)) = D_b) \qquad (5.10)$$

5.1.1.4 Intra-Class Variation Models

The second method attempts to improve reliability by modelling the variation within each species. Each species object's data points are separated into a number of clusters. We model the movement (in the transformation from one pdf to another) within each of these clusters under the mapping between the species object and its training examples (Figure 5.3). This essentially aims to describe how each portion of the distribution typically varies for that species. These models are then used to determine to which species another leaf most likely belongs.

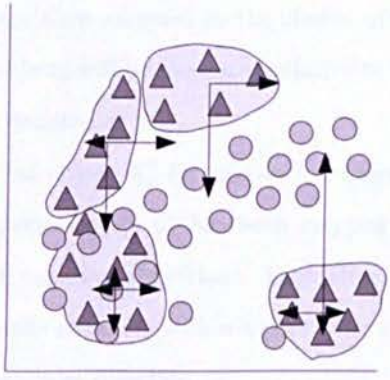


Figure 5.3: Descriptors are generated to model the movement between the class object and another object in terms of each cluster

Training The Classifier. For each species, a species object is created as before. Next a small number, k, of cluster centroids are found, for each species, using the clustering algorithm described in Figure 5.4. This method of clustering creates clusters of equal size, and thereby helps to ensure the centroids are appropriately spread thoughout the distribution, with centroid density approximately proportional to the density of the data points. Any

Figure 5.4 Clustering algorithm

```
Initialise cluster centroids at randomly picked data points
repeat
  for all clusters do
    Sort points according to distance from centroid
  end for
  repeat
    for all clusters do
      Assign next nearest unassigned point to cluster
    end for
  until all points assigned to clusters
  for all clusters do
    Calculate centroid as mean of points in cluster
  end for
until sufficiently converged, or max iterations reached
```

clustering algorithm with similar properties could also be used. All the points in the species object are then assigned to the cluster of their nearest centroid. The change between objects will be measured relative to these clusters. The j^{th} cluster for species i is denoted as K_{ij}.

For each training leaf object X_i^t for species i, a movement descriptor F_i^t is generated, after the species object C_i has been mapped to the training object (as before, using the Hungarian algorithm). Each element in the descriptor is the sum of the movements of points within a particular cluster, for a particular dimension and in a particular direction.

$$F_i^t = \{\bar{f}_{ab}^t | 0 < a < k; 0 < b < d\} \tag{5.11}$$

$$\bar{f}_{ab}^t = [f_{ab+}^t, f_{ab-}^t] \tag{5.12}$$

$$f_{ab+}^t = \frac{k}{|X_i^t|} \sum_{\substack{\bar{x} \in X_i^t \\ x_b - M(\bar{x}, C_i)_b > 0 \\ M(\bar{x}, C_i) \in K_{ia}}} x_b - M(\bar{x}, C_i)_b \tag{5.13}$$

$$f_{ab-}^t = \frac{k}{|X_i^t|} \sum_{\substack{\bar{x} \in X_i^t \\ x_b - M(\bar{x}, C_i)_b < 0 \\ M(\bar{x}, C_i) \in K_{ia}}} x_b - M(\bar{x}, C_i)_b \tag{5.14}$$

where d is the number of dimensions, k is the number of clusters, and t is the training instance. $\bar{x} \in K_{ia}$ indicates that point $\bar{x}$ is assigned to the a^{th} cluster for species i, and x_b refers to the value in $\bar{x}$ corresponding to the b^{th} feature (likewise for $M(\bar{x}, C_i)_b$).

Equations (5.13) and (5.14) calculate the elements of the descriptor for cluster a, in the positive ($x_b - M(\bar{x}, C_i)_b > 0$) and negative ($x_b - M(\bar{x}, C_i)_b < 0$) directions, respectively, along dimension b. These are calculated as being the sum of the distances between training points and their mapped species points, where the mapped point is in the given cluster ($M(\bar{x}, C_i) \in K_{ia}$). These are normalized by multiplying by the number of clusters, k, divided by the number of points in the training object ($|X_i^t|$).

Classification. To classify a leaf X, for each potential species, the mapping and generation of a movement descriptor, F_i, is performed as per the training stage. We then use a Parzen window method [105] with a Gaussian kernel to calculate the likelihoods for each species, and determine the classification.

$$c^* = \arg\max_i P(X|C_i)P(C_i) \tag{5.15}$$

$$P(X|C_i) = P(F_i|C_i) \tag{5.16}$$

$$= \prod_{a=0}^{a<d} \prod_{b=0}^{b<k} P(\bar{f}_{ab}|C_i) \tag{5.17}$$

$$P(\bar{f}_{ab}|C_i) = \frac{1}{T_i} \sum_{t=0}^{T_i} P(\bar{f}_{ab}|\bar{f}_{ab}^t) \tag{5.18}$$

$$P(\bar{f}_{ab}|\bar{f}_{ab}^t) = \phi(||\bar{f}_{ab}^t - \bar{f}_{ab}||) \tag{5.19}$$

where T_i is the number of training examples for species i and $\phi(x)$ is a normal distribution function with mean, $\mu = 0$ and standard deviation, $\sigma = 0.002$.

5.1.1.5 Experiments

In this section the new algorithms are empirically evaluated by comparing it to a selection of other techniques. For these experiment we have 32 different species, with 16 examples of each, performing a 16-fold cross validation. Each example's object has 1024 data points, generated as described in Section 4.2.1. For the first method (Section 5.1.1.3) we use dictionaries of up to 256 codewords, and for the second method (Section 5.1.1.4) we use up to 64 clusters for each species. Whilst results are show to improve up to these values, no significant improvement was found in using higher values.

Methods For Comparison. The three methods we use for comparison are kernel density estimation, the earth mover's distance, and a bag-of-words method using a Naive-Bayes classifier.

- Kernel Density Estimation:

  Kernel density estimation is used to predict the probability density function for each species. This estimate of the pdf is then used to calculate the likelihood of the leaf belonging to that species.

$$P(X|C_i) = \prod_{\bar{x} \in X} P(\bar{x}|C_i)$$
$$= \prod_{\bar{x} \in X} \sum_{\bar{y} \in C_i} \frac{\phi(||\bar{y} - \bar{x}||)}{|C_i|}$$

  where $\phi(x)$ is a normal distribution function with mean, $\mu = 0$ and standard deviation, $\sigma = 0.1$. This kernel function was used as it appeared to give the best results for the dataset.

- Earth Mover's Distance:

  For this we use the pure value calculated by the earth mover's distance instead of utilizing the mapping between objects. Each leaf is classified as belonging to the species whose object is closest to it according to the EMD metric.

- Naive-Bayesian Bag-of-Words:

  For the bag of words method, we use the same codeword dictionary as for the new method, to allow fairer comparison. We use a Naive-Bayes classifier, as it is both one of the most common classifiers (along with SVMs) used for bag-of-words [27], and is similar to that used in the proposed method.

Results. Tables 5.1a, 5.1b, and 5.1c give the results, respectively, for the first proposed method (Section 5.1.1.3), bag-of-words method, and second proposed method (Section 5.1.1.4), used different numbers of data points, and different dictionary sizes/numbers of clusters. The exact same dictionaries were used for both the first method and bag-of-words method. The overall results of the experiments are given in Table 5.2.

	$n = 256$	$n = 512$	$n = 1024$
$\|D\| = 16$	67.97	73.05	75.39
$\|D\| = 32$	75.39	80.66	81.64
$\|D\| = 64$	84.77	85.35	88.09
$\|D\| = 128$	86.13	90.04	90.06
$\|D\| = 256$	90.02	91.02	92.97

(a) Method 1, data-point-mapped bag-of-words, varying object and dictionary size (in %)

	$n = 256$	$n = 512$	$n = 1024$
$\|D\| = 16$	57.03	63.28	63.28
$\|D\| = 32$	62.70	65.82	67.19
$\|D\| = 64$	69.73	74.02	74.02
$\|D\| = 128$	74.41	76.76	77.54
$\|D\| = 256$	77.15	79.30	80.27

(b) Bag-of-words method, varying object and dictionary size (in %)

	$n = 256$	$n = 512$	$n = 1024$
$k = 8$	69.73	80.08	86.33
$k = 16$	83.79	90.63	92.97
$k = 32$	91.21	93.75	98.05
$k = 64$	94.73	96.48	98.05

(c) Method 2, intra-class variation models, varying object size and number of clusters (in %)

Table 5.1: Results for each method.

As the results show, the new methods both performed far better than the standard bag-of-words method. This is because when the difference between pdfs means that points are assigned to different codewords, the standard method considers only that these points are no longer assigned to the same codeword,

Method	$n = 256$	$n = 512$	$n = 1024$
First Proposed Method	90.02	91.02	92.97
Second Proposed Method	94.73	96.48	98.05
Kernel Denisty Estimation	69.73	73.83	77.73
Earth Mover's Distance	73.83	79.88	85.35
Bag-of-Words	77.15	79.30	80.27

Table 5.2: Overall results, using best parameter values for each method (in %).

whereas the new methods both consider where in the feature space those points may exist, given that particular class. The kernel density estimation and earth mover's distance methods both performed worse than the other methods. These methods both directly compare samplings from distributions, and so are susceptible to noise produced by the sampling. The bag-of-words methods eliminate much of this noise, by quantisation via assignment to codewords, as does the second new method, by using the behaviour of different parts of the distribution.

Of these two methods, the second method performs better for plant leaf classification, at 98.05% of leaves correctly classified versus 92.97%. This is likely because this method deals better with the variation within each species, which for this dataset may be quite high, due to varying levels of damage or disease present on the leaves, and slight differences in lighting conditions. For other data where there is either less intra-class variation, or it is less quantifiable, it is possible that the first method may still perform best.

Given that the EMD must be calculated in performing the new methods, it may be possible to improve the results by incorporating the EMD metric. In this case, however, doing so produced no change in results. As would be expected, increasing the number of points used to describe objects increases the quality of the classification, but the new methods still perform better than the other methods when a smaller number of points are used, making them particularly suitable when larger samplings are not practical.

Due to $O(n^3)$ nature of the Hungarian algorithm, the method presented here

can be quite slow compared to some other methods, requiring approximately 6 seconds per leaf in these tests, with $n = 1024$. Despite this, for many applications the additional time required is entirely acceptable given the improvement in accuracy, and the cost can be mitigated to some extent, for example, by using a faster, less reliable method to eliminate the least likely classes first. Furthermore, these methods still performed better than the other methods tested here when $n = 256$, greatly reducing the time required, and allowing accurate classification even when less data is available.

5.1.2 Comparing Leaf Margins Using Dynamic Time Warping

With the leaf margins, there is again much variation within some species, largely due to the different size and coverage of teeth as the leaf develops. In order to compare two margins, a common starting point on the margin must first be selected. The obvious candidates for this are the apices and insertion points.

Whilst these have been identified (in Section 4.4), it is not known which is which. When performing a comparison, all four combinations (possible pairings) for sequence start points are therefore used. Some species' leaves have a degree of asymmetry. Whilst the details along the margin in these cases will be similar on either side of the leaf, the distance between insertion point and apex may be quite different. To account for this, the margin signatures (generated as per described in Section 4.3) are oriented to always procede along the shortest side first. The DTW algorithm (Section 4.4.1) is applied for all four configurations, and the smallest measurement is selected as being the difference between the two leaf margins.

Following the assumption in Section 4.4 that the maximum difference between the lengths of the two sides of the leaf will be $2w$, where $w = \frac{n}{8}$, the value

of k used in Equation (4.6) is also set to $\frac{n}{8}$, as this is the point in the sequences where the most distortion is expected to occur.

5.1.2.1 Results

The method is evaluated on a dataset containing 16 leaves from each of 100 different species. A 16-fold cross-validation is performed, such that one leaf from each species is used each time in the testing set, whilst the remaining leaves are used as the training set. Classification is performed using the k-nearest-neighbour technique, with $k = 5$, with this value chosen as it was found to produce the best results. For comparison, two other techniques are also used on the same data:

- Cross-correlation:

  For two sequences, $\mathbf{a} = \langle a_1, ..., a_m \rangle$, $\mathbf{b} = \langle b_1, ..., b_m \rangle$, the distance between the two is calculated for every possible offset of one sequence against the other. The lowest distance calculated is used for the classification.

$$distance = \min_{0 \leq i < n} \sum_{j=0}^{n} \|a_j - b_{j+i \bmod n}\|$$

- Bag-of-Words:

  A large number of feature vectors are sampled from the entire training set, and a k-means clustering is performed on these. The cluster centroids are used as the codewords in a dictionary used to perform a quantization of the data, by assigning each data-point to its nearest codeword. A margin sequence can then be described as the frequency of occurrence of each 'codeword' [27]. For classification, the distance between two sequences is then calculated as the Jeffrey-divergence metric for their two histograms.

The results for the three methods are shown in Table 5.3 as the rates of correct classification.

Method	Result (%)
Cross-Correlation	57.12
Bag-of-Words	74.51
Proposed Method	91.32

Table 5.3: Results for the three methods.

As the results show, the proposed method performed significantly better than the other two. The cross-correlation method conserves the order of the sequence but is too rigid to account for the variations that occur in leaves, for example in the exact positions of the tips of lobes, which appear as peaks in the signature. By ignoring the order of the sequence, the bag-of-words method describes only the content of the margin, and loses valuable information. By using the DTW algorithm, the proposed method can utilize the order of the sequence, whilst having enough flexibility to deal with the variation inherent to natural data.

5.2 Combining Different Leaf Features

Whilst it has been shown previously that high classification accuracy can be achieved using single aspects of plant leaves, it seems apparent that greater accuracy could be achieved by effectively combining multiple leaf-components. As dataset sizes increase, certain aspects for certain species will become unsuitable for distinguishing them, due to both high intra-species and low inter-species variation causing overlap within the search space. For example, the leaves on a large number of different species have a similar oval shape (Figure 5.5). In these cases, whilst shape by itself may be insufficient, combining it with texture information may prove adequate for accurate classification.

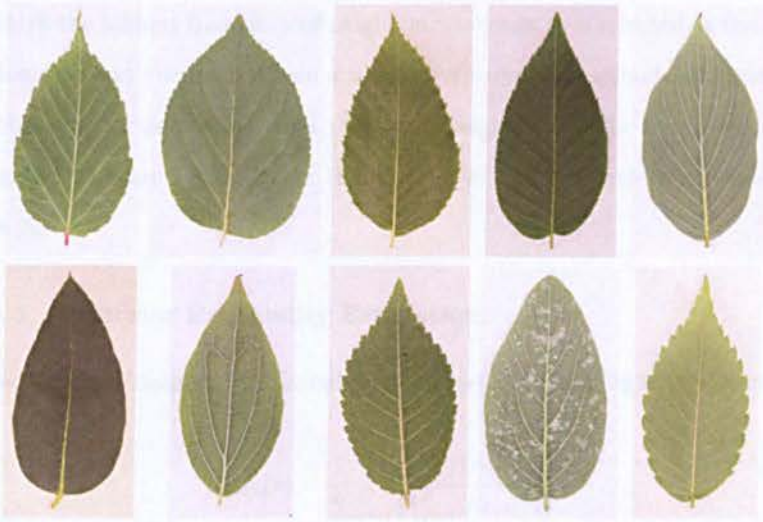


Figure 5.5: Many species have similar leaf shapes

As well as desiring a robust method of performing classification by combining different leaf feature-sets, it would be useful to be able to assess the utility of each feature-set, not just in terms of general classification accuracy, but in terms of how likely it is to be a useful feature-set for classifying a particular leaf. Given the computational requirements of the extraction and comparison, it must be assessed as to whether there is value in using all of the leaf components and methods available. Furthermore, it is plausible that some feature-sets are correlated with each other, reducing the value in using both together. Another concept worthy of investigation, is whether it is best to employ a small selection of highly rigorous, but computationally expensive methods, or to use a larger number of simple yet quick methods.

5.2.1 Probabilistic Classification From K-Nearest-Neighbour

Here, the use of k-NN classifiers for producing probabilistic classifications from multiple leaf feature-sets is explored. Typically with k-NN classifiers, the class

for which the highest frequency of neighbours belongs to is selected as the classification, however there have been a number of proposed methods for producing a probability for each class from the set of neighbours. The two methods investigated here are Fukunaga & Hostetler's [40], and an extended version by Atiya [5].

5.2.1.1 Posterior Probability Estimation

For each species/class, c_i, the probability of a leaf, x, belonging to it is calculated as

$$P(c_i|x) = \frac{\prod_{f=1}^{F} P_f(c_i|x)}{\sum_{i=0}^{C} \prod_{f=1}^{F} P_f(c_i|x)}$$

where $P_f(c_i|x)$ is the probability of class c_i for the leaf feature-set f, and F is the total number of feature-sets to be used.

Fukunaga's method. To calculate the probability from feature-set f, two different techniques are used. The first is that of Fukunaga,

$$P_f(c_i|x) = \frac{K_i}{K}$$

where K is the total number of neighbours being used, and K_i is the number of those neighbours belonging to class c_i.

Atiya's method. Atiya extended this to include weights, calculated from the training set.

$$P_f(c_i|x) = \sum_{j=0}^{K} v_j B_{ij}$$

where v_j is the weight for the j^{th} neighbour, and B is a matrix with $K + 1$ columns, and C rows, with B_{ij} set to 1 if the j^{th} neighbour is from class i, and 0 otherwise. The elements in the final column are all set to $\frac{1}{C}$.

The weights v_j, $j = 1..K$ are calculated as

$$v_j = \frac{e^{w_j}}{\sum_{k=1} K e^{w_j}}$$

with w_j determined by maximising the likelihood of the data. Each value is initialised to be equal, and then updated by

$$w_j = w_j + \eta \left[\sum_{n=1}^{N} \frac{B_{c_n j}(n) e^{w_j}}{\sum_{i=1}^{K+1} B_{c_n i}(n) e^{w_i}} - \frac{N e^{w_j}}{\sum_{i=1}^{K+1} e^{w_i}} \right]$$

where $B(n)$ and c_n are the B matrix and class, respectively, for the n^{th} training sample, and η is the step size. This update is repeated until the change in weights becomes negligible.

5.2.1.2 Experiments

To evaluate these two methods, 4 different leaf feature-sets are used:

1. Shape features - set of 8 features as described in Section 4.1.1.1.
2. Elliptic Fourier descriptors - as per Section 4.1.1.3. Used as well as the shape features, due to there ability to distinguish between species that the shape features cannot.
3. Margin histogram - a 32-bin histogram was generated by quantizing the data generated in Section 4.3.
4. Lamina histogram - the histograms generated in Section 4.2.1 were used.

Due to the superior results previously seen in using the lamina and margin, compared to the shape, smaller dictionary and sampling sizes were used for the data here, to better show the value in combining multiple feature-sets. A 16-fold cross-validation was performed on the 100-species dataset (16 samples per

species). Results were generated for every combination of the four leaf feature-sets. These results are shown is Table 5.4. The accuracy stated is the fraction of leaves correctly classified using that method and combination. The deviation, is the standard deviation for the accuracy between different species. A high deviation shows that the combination performs much better for some species than for otherwise, whilst a low deviation shows that it works similarily for all species.

Method	Shape	EFD	Lamina	Margin	Accuracy	Deviation
Fukunaga			✓		0.5456	0.2739
	✓				0.5987	0.3108
		✓			0.6056	0.2897
				✓	0.6825	0.2796
	✓	✓			0.7531	0.2609
		✓		✓	0.8625	0.1709
		✓	✓		0.8731	0.1359
	✓		✓		0.8756	0.1315
	✓			✓	0.8831	0.1626
	✓	✓		✓	0.8937	0.1579
	✓	✓	✓		0.9093	0.1119
			✓	✓	0.9143	0.1048
		✓	✓	✓	0.9587	0.0656
	✓	✓	✓	✓	0.9625	0.0712
	✓		✓	✓	0.9643	0.0601
Atiya			✓		0.6437	0.2344
		✓			0.6593	0.2577
	✓				0.6643	0.2779
				✓	0.7212	0.2345
	✓	✓			0.7887	0.2249
		✓		✓	0.8762	0.1533
	✓		✓		0.8925	0.1182
	✓			✓	0.8981	0.1437
	✓	✓		✓	0.9025	0.1458
		✓	✓		0.9050	0.1184
	✓	✓	✓		0.9262	0.1098
			✓	✓	0.9337	0.0717
		✓	✓	✓	0.9681	0.0602
	✓	✓	✓	✓	0.9681	0.0640
	✓		✓	✓	0.9688	0.0553

Table 5.4: Results for each combination of leaf feature-sets.

As would be expected, in both cases, the accuracy generally increased as the number of feature-sets used increased. Furthermore, Atiya's method slightly outperformed Fukunaga's, indicating the value in weighting the contribution of each neighbour. It is worth noting that when all the feature-sets except the EFD were used, the accuracy was slightly higher than for all four feature-sets. This shows that in some instances, the use of certain features may be detrimental to the result.

Of the single feature-set cases, use of the margin performed best, whilst the lamina performed the worst. Despite this, the three feature-set instances in which the lamina was used but only one of the shape-based feature-sets were, achieved significantly better results than when both shape-based feature-sets were used, illustrating the need to use a diverse set of features.

5.2.2 Automatic Feature Selection

As has been previously noted, different leaf feature-sets are better suited to classifying some species than others. With intra-species variation in many cases being greater than inter-species variation, certain features will not be adequate for distinguishing between species for which this is the case. Indeed, use of some features could be detrimental to the correct classification. For example, it is possible that, for a given feature, none of the nearest neighbours belong to the correct species, resulting in an incorrect classification, despite whatever value other features may be. As such, it may not always be best to use all of the features available. This can be seen in Table 5.4 where some three feature-set combinations performed better than using all four.

Further to this, as datasets, and the number of species therein, increase in size (it has been estimated that there may be in excess of 400,000 species of plant) the computational cost of identifying a species could become very great.

Given that every feature-set used to this end would add to the cost, there is further benefit in reducing the number of features which need to be used.

A large amount of work has been done in the feature selection field. Most techniques aim to find suitable subsets, either by searching through candidate subsets (Wrapper methods) or by using prior knowledge to predict the best features (Filter methods). Wrapper methods range from basic techniques such as forward selection and backward elimination [71] to more modern methods such as those of Chen et al [19] and Rashedi et al [112]. Filters methods include correlation-based selection techniques [149] and Markov blanket filters [72].

Since some leaf feature-sets have been shown to work well for indentifying some species, whilst others perform better for others, it may be beneficial to dynamically select which feature-sets to use based on the leaf in question, rather than using a predetermined set of features. Furthermore, it may prove possible to evaluate the utility of a particular component for classifying a particular leaf without needing to generate a full set of features for it. Here, a method and a number of metrics are explored for dynamically selecting feature-sets on a leaf-by-leaf basis, along with an evaluation of their effectiveness.

5.2.2.1 Metrics For Feature Utility

Given a vector for a particular leaf component, there are a number of different metrics which could be used to estimate that component's utility for classifying the leaf, prior to performing the classification. The metrics explored here are based upon the neighbourhoods the sets of nearest neighbours) used in the previous section for classifying the leaves. These metrics are calculated for all examples in the training set. When a new vector is presented, the methods to be described in Section 5.2.2.2 use these past calculations to estimate the value for the new vector.

1. *Same vs K* - The fraction of the K nearest neighbours which belong to a training examples true species - K_t/K, where K_t is the number of neighbours from the same species as the training example. This metric reflects Fukunaga's method in Section 5.2.1.1.

2. *Same vs Next Highest* - The number of neighbours from the correct class divided by the number of neighbours for the second highest scoring class. This gives a measure of the likelihood that the vector would have been correctly classified.

3. *Neighbouring Classes* - 1 divided by the total number of different classes represented within the neighbourhood. If all the neighbours are from different classes, the vector is less likely to classify correctly than if they are all from the same class.

4. *Entropy* - Provides a measure of uncertainty, with low values indicating a high level of predictability. Calculated as

$$E = 1 - \frac{\sum_{i=1}^{C} p(c_i) \log p(c_i)}{\log K}$$

where $p(c_i)$ is the fraction of neighbours belonging to class c_i. This metric has the advantage of giving the same value for a vector regardless of which species in came from, in relation to the neighbours.

5.2.2.2 Estimation of Feature Utility

For estimating the utility of a new vector, a feed-forward neural-network is trained via back-propagation on the training vectors, with the utility metric values for those training vectors as the expected output. The network consists of two hidden layers, each with twice as many nodes as the number of input nodes, and a single output node. The number of input nodes was dependent

on the leaf feature-set being used. Sigmoid functions ($f(x) = (1 + e^{-x})^{-1}$) are used at each hidden and output node. Training was performed by introducing all of the training vectors to the network in a random order. This process was repeated until the decrease in the average output error dropped below a small threshold. Once trained to sufficient convergence, the utility of a new vector can be estimated by inputting the vector to the trained network.

5.2.2.3 Classification With Leaf Feature Selection

Once an estimate of the utility of each of the four feature-sets has been acquired for a leaf by using the trained neural-network, this information can be used to minimise the number of feature-sets required to perform a classification. The leaf is first classified using the feature-set with the highest utility, and the probability for the top result is calculated as per Section 5.2.1. If this probability is greater than some predetermined threshold (to be discussed in Section 5.2.2.4), the result is accepted, else the feature-set with the next highest utility is selected and the probability recalculated, until the threshold is passed.

In cases in which all the available features appear to be necessary, the contribution from each feature-sets is weighted according to its estimated utility:

$$P(c_i|x) = \frac{\prod_{f=1}^{F} P_f(c_i|x)^{w_f}}{\sum_{i=0}^{C} \prod_{f=1}^{F} P_f(c_i|x)^{w_f}}$$

5.2.2.4 Experiments and Results

The method is tested using the same data as in Section 5.2.1. The networks used consisted of two hidden layers, with $2d$ nodes per hidden layer, where d is the size of the input vector.

The method was run for each of the metrics, using the four leaf feature-set described in Section 5.2.1.2, and with the total number of feature-set to use being varied from one to four. When less than four of the feature-set are used, they are selected in order of estimated untility (i.e. when three are used, the feature-set with the lowest estimated utility is ignored).

These results are shown in Table 5.5. When all four features are used, the results are naturally the same, but as the number of features used is reduced, an increasing improvement can be seen over using fixed sets of features. In Table 5.4, the highest performance from using a single feature-set was 0.7212, but by selecting which feature-set to use on a leaf-by-leaf basis, here an accuracy of 0.8037 was achieved. Of the four metrics, the fourth one, based on the entropy performed best. This metric seems the natural choice as it directly relates to the predictability of the result for a given part of the feature space.

	No. Features			
Metric	1	2	3	4
1	0.7575	0.8938	0.9406	0.9681
2	0.7738	0.8994	0.9275	0.9681
3	0.7763	0.9150	0.9456	0.9681
4	0.8037	0.9231	0.9538	0.9681

Table 5.5: Classification accuracy for each number of feature-set and metric.

Table 5.6 shows the frequency that each combination of features was used, and the corresponding accuracies. For example, when two feature-set were used, the combination of EFD and margin was used for 55.25% of the leaves. As would be expected, for most combinations the accuracy is higher than when applied to all the leaves in the test set, with those combinations that perform better in general being used with higher frequency. The combination of shape features and lamina achieved 100% accuracy, however the combination was only selected for a very small number of leaves The final two columns, P_{true} and P_{false}, show the average Fukunaga probability (used to determine the classification) when

the result is correct and false respectively. The Fukunaga probability can be seen as a measure of confidence that a leaf has been correctly classified. This value is typically far higher in the former case, when the classification is correct, suggesting it could be of use in determining when to stop increasing the number of feature-set used, as increasing the number of feature-set also tends to increase this value.

Shape	EFD	Lamina	Margin	Acc	Frequency	P_{true}	P_{false}
✓				0.9080	0.0544	0.4077	0.2887
	✓			0.8437	0.2719	0.4169	0.2817
		✓		0.8186	0.1412	0.4013	0.3048
			✓	0.7688	0.5325	0.3982	0.2678
✓	✓			0.9744	0.0487	0.9186	0.5677
✓		✓		1.0000	0.0144	0.8583	0.0000
✓			✓	0.9682	0.0981	0.9198	0.4298
	✓	✓		0.9746	0.0737	0.9044	0.4208
	✓		✓	0.9038	0.5525	0.8845	0.5401
		✓	✓	0.9176	0.2125	0.8502	0.4058
✓	✓	✓		0.9600	0.0312	0.9656	0.5826
✓	✓		✓	0.9383	0.3950	0.9702	0.5743
✓		✓	✓	0.9851	0.0419	0.9605	0.3027
	✓	✓	✓	0.9624	0.5319	0.9514	0.5593
✓	✓	✓	✓	0.9681	1.0000	0.9779	0.6706

Table 5.6: Frequency and accuracy of each combination.

In Table 5.7, the results are shown for the weighted and unweighted methods of selecting features, and for a method using the same features for each leaf, with the order of application predetermined, based on their individual performances (in this case, the order is margin (accuracy 0.7212), shape features (0.6643), EFA (0.6593) and then texture (0.6437)). Both the weighted and unweighted methods perform significantly better than the fixed method when a reduced number of features is used. When more than one feature-set is used, the weighted version is able to perform slightly better than the unweighted.

Finally, we examine whether the number of feature-sets required for accurate classification of a leaf can be determined from the calculated Fukunaga proba-

	No. Features			
Mode	1	2	3	4
Fixed	0.7212	0.8981	0.9025	0.9681
Unweighted	0.8037	0.9231	0.9538	0.9681
Weighted	0.8037	0.9256	0.9575	0.9700

Table 5.7: Results for the three methods of selecting features.

bility. This will allow the minimisation of the number of feature-sets used. The threshold on this probability is varied from 0 to 0.9, with additional feature-sets being added until this threshold is passed. Once the threshold has been passed the leaf is classifed as the species with the highest probability. Figures 5.6 and 5.7 show the frequency for which each number of features is required at each threshold, for the unweighted and weighted forms respectively. Here, a five indicates that the threshold was not met even when using all four features. In most cases, only two features were required to meet this target. In all cases, the modal number of features needed, before the maximum accuracy was reached, was two or less. In the weighted case, higher thresholds (above 0.5) required more features to be used, but did not result in an improvement in accuracy.

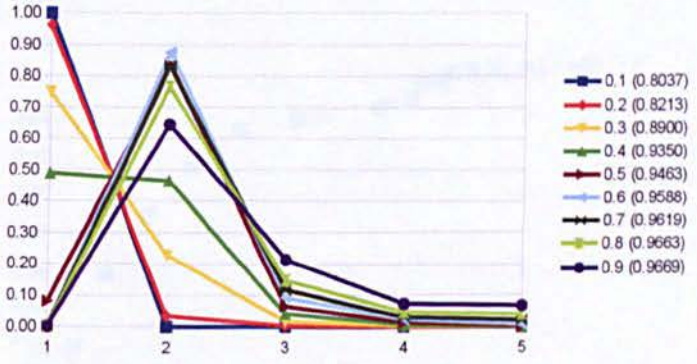


Figure 5.6: Frequency (y-axis) of number of features (x-axis) required to meet probability threshold, unweighted case. Number in brackets indicates the accuracy achieved at that threshold. Five indicates threshold was not met when using all features.

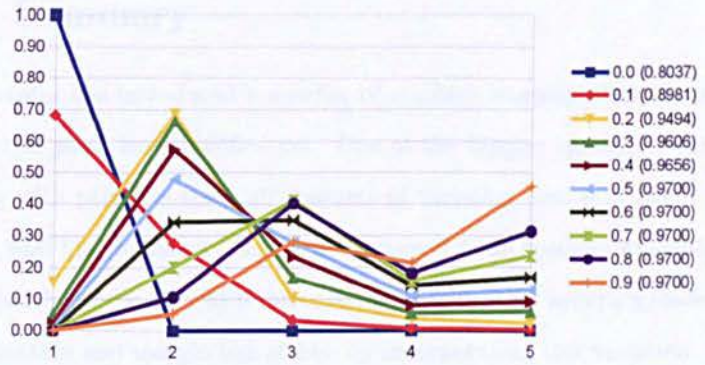


Figure 5.7: Frequency of number of features required to meet probability threshold, weighted case.

Figure 5.8 show the accuracy achieved at each threshold, in relation to the average number of features required. As can be seen, the accuracy increases as the average number of feature-sets is increased, but there is little improvement in the results beyond the use of a threshold which achieves an average of two feature-sets being used. Furthermore, in the weighted case the maximum accuracy required an average of three, rather than the previous four, features.

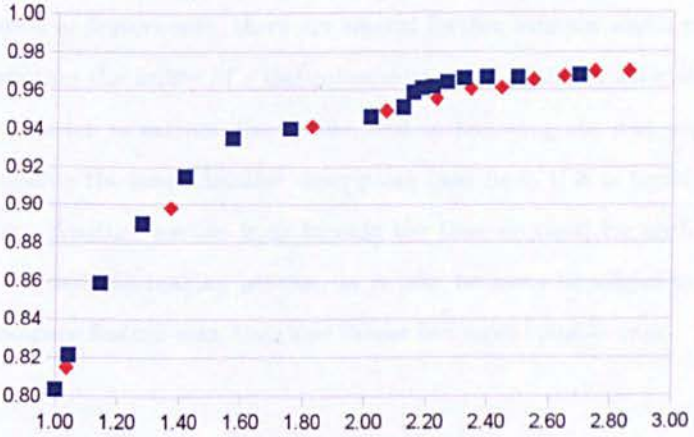


Figure 5.8: Accuracy (y-axis) when different average number of features (x-axis) used. Blue - unweighted; Orange - weighted.

5.3 Summary

This chapter has introduced a number of machine learning algorithms suitable for used in plant leaf classification. One of the biggest challenges involved in working with plants is the high amounts of variation that may occur within a species, and high similarity that exists between some species. Algorithms have been described here for which increase the classification accuracy, based on the macro-texture and margin signatures, by incorporating this variation.

Furthermore, a framework has been provided for combining the classification from multiple sources of information. This has included the leaf-dependent selection of suitable feature-sets, in order to improve the classification whilst simultaneously decreasing computational costs, but automatically eliminating detrimental feature-sets.

There is still much work that can be done in this area. Continuing the incorporation of intra-species variation into the classification, it would be useful to have a similar such method based on the leaf shape, where there can be a large difference between leaves even on the same plant. For the selection and combination of feature-sets, there are several further avenues worth exploring. One is whether the utility of a leaf component can be reliably estimated using simplified, faster to extract descriptors, and so removing the cost associated with compiling the more detailed descriptors used here, if it is found to be of low utility. Another avenue is to include the time required for each feature-set into the decision making process, as it may be more beneficial to use two fast-to-compare feature-sets, than one slower but more reliable one.

Chapter 6

Botanists' Vision

Chapter contributions:

- Study of the difference in eye-movements between botanists and non-botanists when viewing leaf images, using eye-tracker data.
- Preliminary work towards replicating a botanist's observation points, based on this data.

Given that professional botanists have received extensive training and experience in studying and identifying plants, it may be possible to improve upon computational methods, but making use of information regarding how they view leaf images. These data can be captured through the use of eye-tracking technology, since a botanist may not in fact be aware of the precise process they are performing.

When viewing any detailed image, such as an advertisement, website or some particular object, the attention of the human visual system is attracted to certain features, known as salient regions. This process of observation is to a large extent innate and subconscious, although can become less so through prior-knowledge of the observed image, or experience in viewing particular types of

image. Research into eye-movement is involved in a several fields, including and beyond the study of perceptual systems. The study of eye-fixation points and saccades (fast eye movements between points of interest/stimuli) can provide insight into cognitive processes such as written language comprehension, memory, mental imagery and decision making [114]. Eye movement research is of great interest in the study of neuroscience and psychiatry, as well as ergonomics, advertising and design [141]. Since eye movements can be controlled, to some degree, voluntarily, and detected and recorded by modern technology with great speed and precision, they can now be used as a powerful input device for many practical applications in human-computer interactions [115].

Wearable eye-tracking devices allow collection of eye-movement information for natural scenes, involving the use of generally unconstrained eye, head, and hand movements. The most commonly sought eye-tracking metrics include the number, duration and location of fixations, both across the entire scene and within set areas of interest, and the sequence of movements between them, among many others [92, 64]. Longer fixation periods generally indicate greater cognitive processing of the fixated region, possibly due to a higher level of detail or a lower scale feature of interest, and the percentage of total fixation dedicated to a particular area may indicate its saliency [34, 121].

With sufficient knowledge and experience, an expert in a particular field can become highly efficient at analysing certain types of images. This could be a physician searching for anomalies in images produced by medical scanners, a botanist studying images of leaves to determine a plant's species, or a security personnel identifying suspicious behaviour in CCTV footage. Using advanced eye-tracking technology, we can capture and analyse in great depth the process through which a human expert analyses such images. This chiefly involves identifying their fixations, and analysing the sequence in which these fixations

are visited. Through this it may be possible to enable a computer system to more accurately replicate the human expert's fixation process. This could lead to advances in the use of computer vision techniques to perform such tasks, as it would allow more efficient processing of the images, and may reveal additional information which current techniques are overlooking.

In terms of plant classification, this could aid in the identification of which parts of the leaf are most important for the task, or inspire new processes that may not yet have been considered.

In this chapter, eye-tracker data is used to perform preliminary work towards understanding how botanists study leaf images, and for replicating a botanist's observation points when performing a leaf recognition task.

6.1 Comparing The Eye-Movements Of Botanists and Non-Botanists

Before the eye-tracker data can be utilized in this manner, it is important to first establish that the knowledge and experience acquired by botanists does indeed have an effect on their fixation points, and the sequences thereof, whilst viewing leaves. Here, a pilot study is conducted, demonstrating the difference in eye-movements between botanists and non-botanists. In the process, initial data will also be gathered for use towards the aforementioned aims of replicating the botanists observations.

The experiments performed involve subjects performing a simple leaf recognition task. Subjects are shown an image of a leaf for a short period of time. Afterwards, they are allowed to view images of leaves from eight different species, one of which is from the same species as the initial leaf. The subject is tasked with identifying which of the eight leaves is the one from the same species.

Sets of species are chosen such that the leaves portray similar visual qualities (for example, similar colouration or shape), and thereby making the task non-trivial. This was carried out at two different display intervals for the initial images, 1500ms and 4000ms. Each of these intervals was used eight times, with a different set of leaves being used each time.

During this task, the subject was wearing a head-mounted eye-tracking device. This involves a camera capturing the view in front of the subject, a second camera capturing a video of one of their eyes, and software capable of calculating, from the eye-movements, precisely where in their field of view they were looking. These observation points are recorded and translated back onto the original leaf image. Before each set of tests, the tracker was calibrated, and the subject was asked to try minimise head movement during the test. Between each set of the tests, the subjects were allowed a short period of rest.

The task was performed by nine volunteers - four botanists and five non-botanists.

6.1.1 Results and Analysis

The heatmaps - visualisations of where on an image the subject looked - from a selection of the leaf images used are shown in figures 6.1 and 6.2, for the 1500ms and 4000ms cases respectively. These show the areas on which the subject fixated, with the brightest red representing the regions where their gaze remained for the longest. These were generated as a Gaussian mixture model of each point the eye-tracker detected the subject looking at. They are normalised such that the point on which the subject's eyes spent the most time is shown as pure red. To begin with, it can be observed that in the 1500ms case, the botanists typically viewed smaller, more select regions of the leaves than the non-botanists. This effect was diminished in the 4000ms case, when more time

was available to look over the entire leaves. Another observation is that the focal points fell more often outside the surface of the leaves for the non-botanists than the botanists, suggesting that the non-botanists are using more the details of the shape of the leaf, than its interior characteristics.

In Figure 6.3, the histograms of the fixation lengths - the length of time spent fixating on a single point - for both groups are presented. It can be seen that the botanists' fixations tended to be longer (at an average of 0.272 seconds versus 0.209 at 1500ms, and 0.353 versus 0.284 at 4000ms) indicating that areas were observed in greater detail. In the 1500ms tests, the non-botanists had over half of their fixation in the 0.12 second interval, indicating that they did not concentrate for very long on any one part of the leaf, instead opting to cover a larger portion of the leaf in the time permitted.

The saccade amplitudes (eye-movement sizes) are shown in Figure 6.4. Here there appears to be little significant difference between the two groups. The largest portion of saccades in all cases were around 3°, equating to approximately one tenth of a leaf's length, indicating a tendency to move from one region to a another relatively close by, rather than hopping from one side of the leaf to the other.

Table 6.1 gives the average fixation lengths, saccade amplitudes and densities for each of the subjects. Here the density refers to a measure of the portion of a leaf viewed, with lower scores indicating that the subject concentrated on a smaller area of the leaf. Confirming the difference in fixation lengths, the botanists average fixations were longer than all but one of the non-botanists at 1500ms, and longer than all of them at 4000ms. For saccades, there was quite a high variation between individuals, with the average amplitudes ranging from 3.954° to 6.642°, but no particular trend between the groups. In terms of the densities, when the viewing time was limited to 1500ms, all botanists

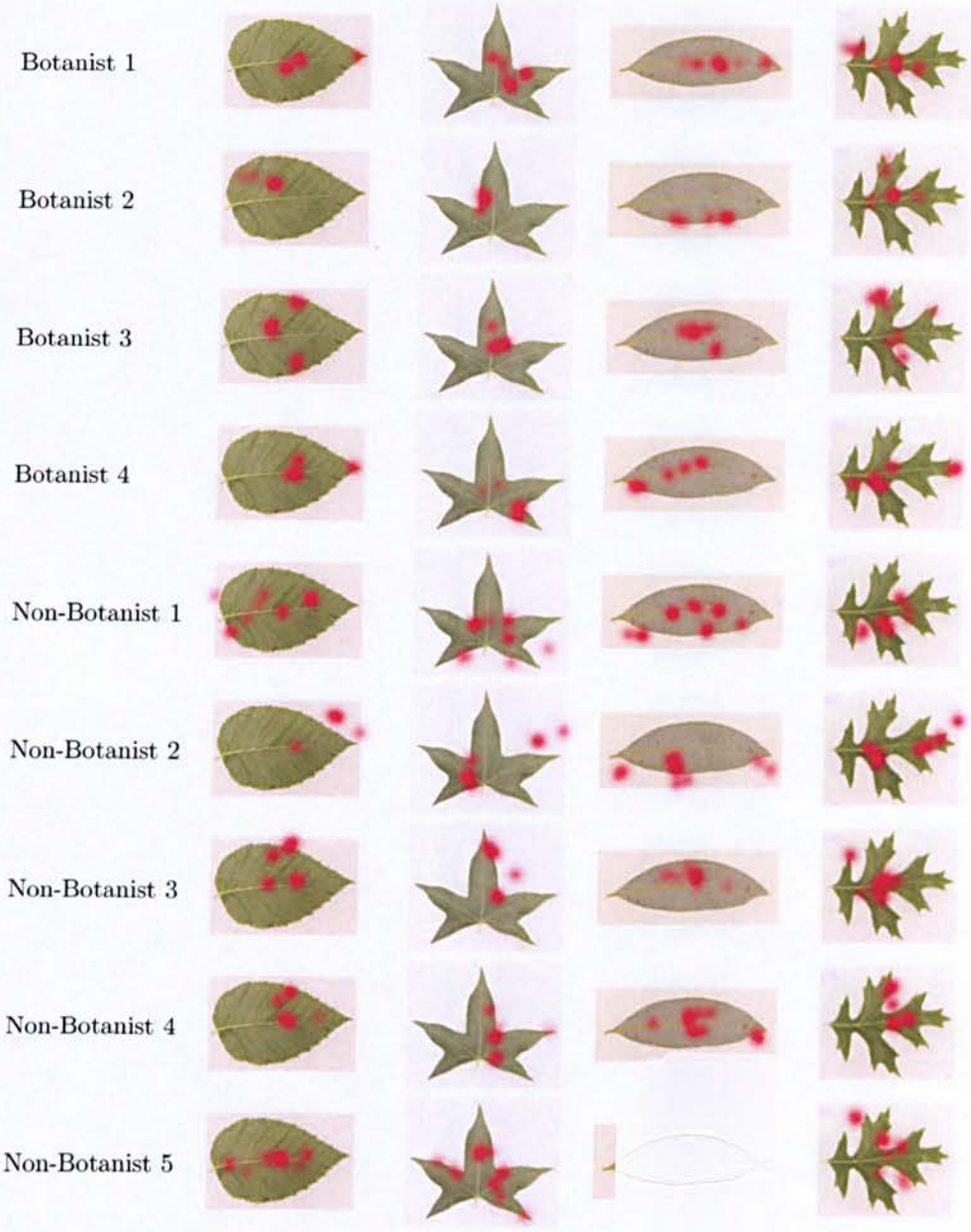


Figure 6.1: Example heatmaps from when the subjects were shown the leaves for 1500ms.

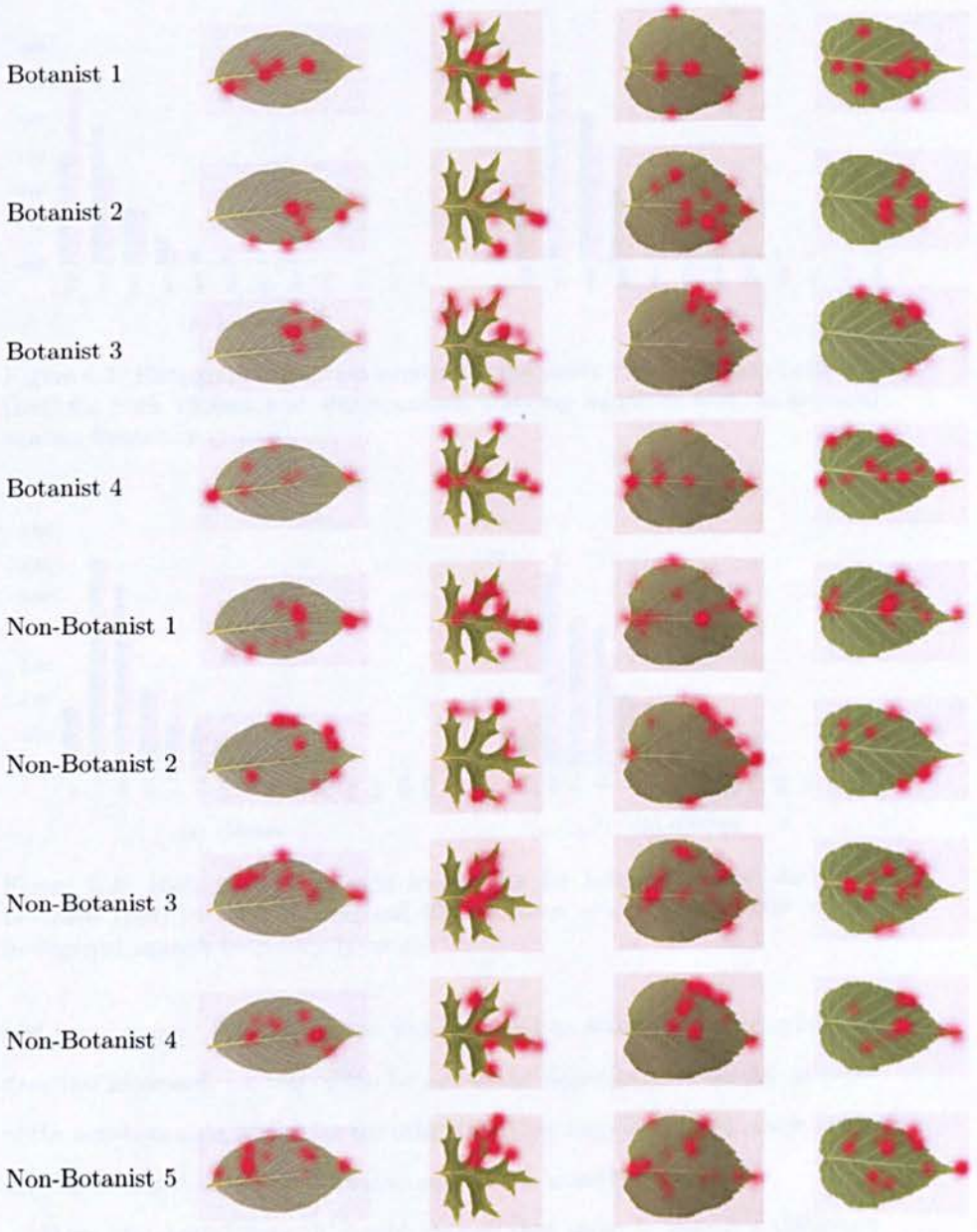
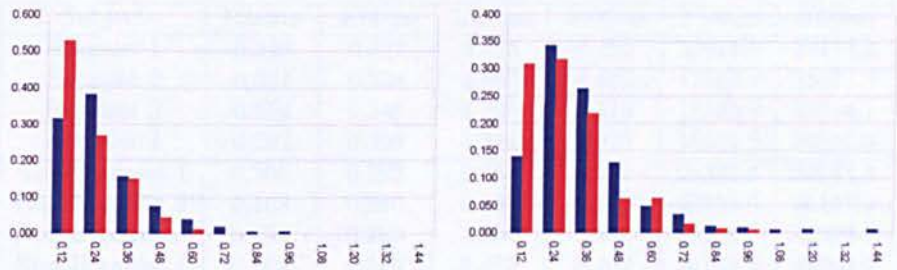


Figure 6.2: Example heatmaps from when the subjects were shown the leaves for 4000ms.



(a) 1500ms (b) 4000ms

Figure 6.3: Histogram of fixation lengths for botanists (blue) and non-botanists (red) for both 1500ms and 4000ms cases, showing length (x-axis, in seconds) against frequency (y-axis).

(a) 1500ms (b) 4000ms

Figure 6.4: Histogram of saccade amplitudes for botanists (blue) and non-botanists (red) for both 1500ms and 4000ms cases, showing amplitude (x-axis, in degrees) against frequency (y-axis).

had lower scores. When the time was increased to 400ms, however, whilst all densities increased, for two of the botanists the densities were similar to those of the non-botanists, whilst for the other two, they remained significantly lower, demonstrating that not all botanists exhibit the same behaviour.

From this initial data, it would appear that there is indeed a difference between how trained botanists and non-botanists view leaf images, during a recognition task. The main quantifiable difference is that the botanists preferred to study small parts of the leaf in high detail, whilst the non-botanists attempted

	Fixations		Saccades		Densities	
Subject	1500ms	4000ms	1500ms	4000ms	1500ms	4000ms
Botanist 1	0.249	0.327	5.495	6.492	17218.0	35174.4
Botanist 2	0.301	0.338	4.557	4.890	17660.8	35174.4
Botanist 3	0.254	0.348	4.293	5.516	18585.1	23744.1
Botanist 4	0.282	0.399	5.714	5.910	18401.2	26539.0
Non-Botanist 1	0.206	0.262	6.254	6.072	25323.6	34644.9
Non-Botanist 2	0.168	0.280	6.642	6.252	22752.0	36157.9
Non-Botanist 3	0.194	0.263	3.954	4.668	21155.7	34299.7
Non-Botanist 4	0.261	0.319	5.427	5.762	20735.3	33704.9
Non-Botanist 5	0.218	0.295	5.105	4.913	22283.4	33422.8

Table 6.1: Average statistics for each subject.

to acquire information from a larger portion of the leaf, and consequently in less detail. There is also suggestion that the non-botanists relied more on the shape-related features than on internal features.

6.2 Reverse Engineering Expert Visual Observations

Here, a first step is presented towards utilizing this type of eye-tracking data for computer vision purposes, concentrating on its use in the study of the classification of plant leaves, from the perspective of the expert in plant systematics, which uses tools based on morphology for identification and is one of the principal branches of study in plant biology. Plant systematists are responsible for the organisation and accessibility of plant diversity data which is underpinned by accurate identification and naming. Figure 6.5 illustrates the typical sequences of fixations when an expert in plant systematics studies a leaf. In the approach here, neural-gas algorithms [89] are applied for filter parameter learning, to discover a set of filters which are particularly well suited for identifying the fixation points on an image of a leaf.

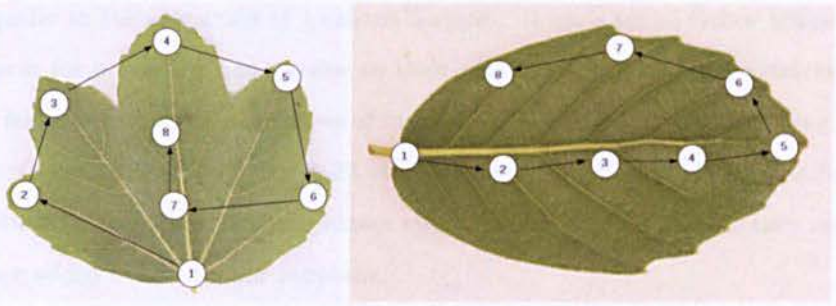


Figure 6.5: Synthetic examples of typical sequences of fixation for eye-tracker data from an expert botanist.

6.2.1 Related Work

In Filter Parameter Learning (FPL) [11, 55, 76], a set of image filters are described by some parameters whose values change dynamically through the course of some learning process. There have been numerous approaches to this problem. In [55], Heidemann presents an object recognition architecture based on feature extraction by Gabor filter kernels, and performs feature classification by an artificial neural network. The parameters of the Gabor filters are optimized to the specific problem by minimizing an energy function. These Gabor filters can then be used to extract features that can be more easily classified by a neural network. Alain and Shigeru in [11] used a discriminative feature extraction method applied to a bank of filters for the modelling of speech. A method proposed by Koray et al. [68] automatically learns the feature extractors in an unsupervised fashion by simultaneously learning the filters and the pooling units that combine multiple filter outputs. The method generates topographic maps of similar filters that extract features of orientations, scales, and positions. By doing this, locally-invariant outputs are produced. In [43], Gautama et al. force the filters to partition the input space in an equitable manner: each filter is tuned to a different frequency region and contributes

equally to the extraction of localized features. Here, a set of Gabor filters is learnt for processing images, due to their well-known properties in extraction of features from their parameters of frequencies, orientations, and smoothing of the Gaussian envelope [111, 48, 22, 80]. Furthermore, links have been identified between Gabor filters and the human visual system [30], and as such they may have added benefit for our purposes.

In the field of neural networks many different architectures and training rules exist, from the perceptrons (from single-unit to multilayer versions), Hopfield-type recurrent networks (including probabilistic versions strongly related to statistical physics and Gibbs distributions) and the Self Organizing Map (SOM), among others [37, 53]. In a self-organising map, the network being trained has a fixed topology throughout, however there exist several variants where, based on errors within the network, elements of the network are added or removed. The neural-gas algorithm [89] is one such variant. It uses a fixed number of nodes which are initially distributed either randomly or uniformly throughout the input space. Connections between these nodes/neurons are added or removed so that for every input pattern, the two closest nodes are connected in the final network. In short, the organization of neurons, according to their distance to the input pattern, and subsequent modification of its reference vector, produces the neuron expansion within the input space. The neurons' positions and their connections become configured to accurately represent the data distribution. Subsequently, by adding and deleting edges, a triangulation between different processing elements is provided. An extension to this, the growing-neural-gas algorithm (GNG) [38] is initialised with just two nodes, and adds more over time. Furthermore, it removes any nodes which have become separated from the network in an unused area of the space. This removes the requirement for *a priori* knowledge about the topological dimension of the space of input

vectors [93]. The method here is based on this form of the neural-gas algorithm.

6.2.2 Methodology

The approach here finds a set of image filters that can be used to efficiently identify possible fixation points on an image of a plant leaf. Firstly, data was collected for where such fixation points lie by using an eye-tracking device to capture a botanist's eye movements as they study a series a leaf images. Each leaf was shown to the botanist for a set period of time, during which they were asked to verbally give as much information as possible about the leaf. This was not recorded, but was done to ensure the manner in which the leaves were studied was realistic and relevant, as it meant the botanist had to look at areas of the leaf which would provide the most useful information. The fixations which have been discovered are used as input into an algorithm which attempts to find a set of filters which give high responses to fixation windows (Section 6.2.2.1). The filters learnt are based upon the Gabor model (Section 4.2.2.1). The learning is performed using a variant of the growing-neural-gas algorithm (Section 6.2.2.3).

6.2.2.1 Fixations and Filter Responses

A *fixation point* is defined as being a point on an image where a person focuses their attention a short amount of time (typically more than 100ms [12]). Using eye-tracking technology, where these fixation points are can be identified, as a series of images are shown to a subject. If the images are each only shown for a particularly short amount of time (no more than a few seconds), the fixation points found may correspond to the most salient parts of the image [88]. If, however, the expert is allowed to study an image for a longer period of time, the fixation points discovered will indicate the most important parts of the image required for the expert to analyse it. Furthermore, the time that the

expert spends concentrating on each fixation point, and the order in which they move their vision between them, can provide important information and insight into the experts' processes.

In analysing and searching for fixation points, a *fixation window* is defined as a square region centred around a fixation point. The size of this window should correspond to the scale of the feature which the expert is studying, which may correspond to the time which the expert spends looking at that feature. At this stage, however, it has been chosen to fix this at 100 pixels in width.

The method described in here is intended to discover filters which will be useful for identifying fixation points. To achieve this the response is calculated for a particular filter being applied to a particular fixation point as being the sum of the absolute values of the convolution between filter and image at each pixel within the fixation window. The algorithm will search for a set of filters which produce high responses to fixation points.

6.2.2.2 Gabor Filters

The aim is to find a set of n filters, $F = \{f_1, f_2, ..f_n\}$, that can be used to efficiently identify fixations. For this purpose, Gabor filters have been chosen. These have been applied to a large range of computer vision problems including image segmentation [124] and face detection [59]. Gabor filters have been used in models of the human visual system, therefore are expected to prove useful here [29]. Gabor filters have been described in Section 4.2.2.1.

It is possible to produce a wide variety of different filters, through the use of only a small set of parameters. To this end, the following parameter ranges were used: $\theta \in [0, \frac{\pi}{2}]$, $\gamma = 0.6$, $\sigma \in [1, 10]$, $\lambda \in [\sigma, 8\sigma]$ and $\psi \in \{0, \frac{\pi}{2}\}$.

6.2.2.3 Learning The Filters

In order to learn the set of filters, a neural gas algorithm is used. Fixations are chosen one at a time at random from the training set, and a filter is found which gives a high response for that fixation. This filter is then used as an input pattern for the neural gas algorithm. Filters are selected by testing a set number of filters sampled from the portion of the input space occupied by the neural gas (see Algorithm 1 for details), and then chosing the one that provided the highest response. The advantage of this is that it helps speed convergence, and avoids wasting computation time by only testing filters that are likely to prove useful.

The particular neural gas algorithm used here is a modification of the growing-neural-gas algorithm [38]. The original algorithm was initialised with two neurons, and grown by adding a new neuron every set number of iterations. Since it is desirable, for our purposes, to only find a minimal number of filters (using a large number of filters would reduce efficiency when searching for fixations), the algorithm is instead started with the maximum required number of neurons, and only adds new neurons whenever the algorithm removes a neuron which has become separated from the network, thus the number of neurons remains constant. The advantage of using this approach over the standard neural-gas algorithm is that it replaces neurons that appear less useful, thereby aiding convergence. Furthermore, the removal of connections allows the gas to separate if discrete regions of the space need to be occupied. Once the algorithm has converged, post-pruning [17] is applied to further improve the final set of filters. The post-pruning algorithm removes clusters of neurons from unused parts of the space, and adjusts the positions of others, to achieve better final results.

The algorithm can be summarised as follows:

1. Initialise the neural-gas:

- Neurons are uniformly distributed through filter-space, within predetermined bounds for each parameter.
- Connections are created between neighbouring neurons. Each connection has an age, which is initially set to 0.

2. Until stopping criteria is met, repeat:
   (a) Select a fixation from the training set.
   (b) Generate k filters, drawn from the distribution of neurons, as per Algorithm 1, which generates random filters returning the first which matches the criteria.
   (c) Calculate the response for each of the filters being applied to the training fixation.
   (d) Apply one step of the neural-gas algorithm (see Section 6.2.2.3), using as the input pattern the parameters of the filter with the highest response in the previous step.
3. Apply post-pruning [17] to the final neural-gas.

Algorithm 1 Kernel density estimation algorithm for selecting filters

repeat
  $\xi \leftarrow$ random filter vector
  $x \leftarrow 0$
  for all neurons $f_i \in F$ **do**
    $x \leftarrow x + \frac{1}{\sqrt{2\pi\sigma^2}} \exp^{-\frac{\|\xi - f_i\|^2}{2\sigma^2}}$
  end for
  $y \leftarrow$ random value in range $[0, \frac{|F|}{\sqrt{2\pi\sigma^2}}]$
until $y \leq$x
return ξ

A Modified Growing-Neural-Gas Algorithm At each step the modified growing-neural-gas algorithm calculates the new positions of its neurons according to an input pattern ξ:

1. Find the two neurons, f_1, f_2 closest (by Euclidean distance) to the input pattern ξ.

2. Increment the age of all connections between f_1 and its neighbours.

3. Increase f_1's accumulated error by $||f_1 - \xi||^2$.

4. Move f_1 and its connected neurons towards ξ, by fractions ϵ_b,ϵ_c (here 0.2 and 0.1) respectively:

$$f_1 = f_1 + \epsilon_b(\xi - f_1)$$

$$f_c = f_c + \epsilon_c(\xi - f_c) \text{for all direct neighbours } c \text{ of } f_1$$

5. If no connection exists between f_1, f_2, create a new connection, else reset the connections age to 0.

6. Remove any connection with an age above some threshold.

7. Remove any neurons which have become disconnected from all other neurons.

8. For each neuron removed in step 7, insert a new neuron as follows:

   (a) Find the neuron, f_i, with the largest accumulated error (from step 3), and the neuron, f_j, with the highest accumulated error of all f_i's neighbours.

   (b) Insert a new neuron, f_k, between f_i and f_j:

$$f_k = \frac{f_i - f_k}{2}$$

   (c) Replace the connection between f_i, f_j with new connections between f_i, f_k and f_j, f_k

(d) Decrease the accumulated errors of f_i, f_j, by multiplying them by some constant. Set accumulated error of f_k equal to that of f_i.

9. Decrease all error variables by multiplying them by some constant.

6.2.3 Evaluation

Due to the nature of the data acquired by an eye-tracker, quantitative results are difficult to obtain. This is because there are no definitive negative testing examples. For example, when studying a leaf, a botanist may only need to look at the margin on one side of the leaf to obtain the information they require from it. If the margin on the other side of the leaf goes un-viewed, this does not mean it is any less relevant, since the decision to use one particular side may have been arbitrary, due to leaf symmetry, and may be different on a second viewing of the same leaf. Because of this, instead of trying to identify all possible fixation points on a leaf image, the evaluation method instead tries to locate a couple of different leaf features, and treat all other areas of the leaf as negative examples.

From the data collection using a professional botanist, it has been identified that a leaf's insertion point (where the petiole (stem) joins the leaf) and apex (the 'tip' of the leaf) are fixation points on most leaves (Figure 6.6). The filters which have been learnt are used to identify these points from a set of points randomly taken from some leaf images. In the first experiment (Section 6.2.3) we use a nearest-neighbour classifier to label image windows as either an insertion point, apex or other. In the second (Section 6.2.3), the points on each leaf that are most likely to be the apex and the insertion point are found. The results are compared to those using filters similar to the popular Leung-Malik and Root Filter Set filter banks [78, 45]. Results are discussed in Section 6.2.3.

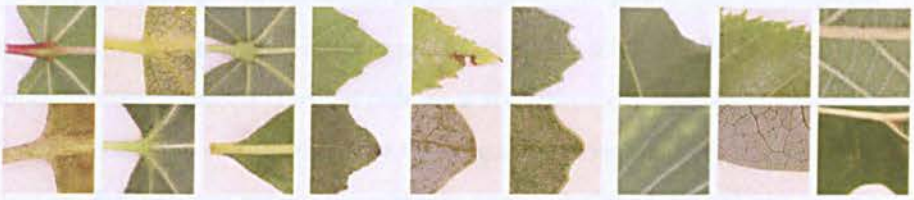
The images used were automatically oriented according to the leaf's primary axis, and were scaled so that each leaf had an area within the image of approx-

imately 2^{18} pixels. A fixation window width of 100 pixels was used, as this allowed for an appropriate size region around the apices and insertion points.

Experiment 1. In the first experiment, the suitability of the filters that have been learnt for identifying whether a given window is fixation window or not is analysed. The training set used consists of the windows centred around the insertion points and apices of 240 leaves from a total of 30 different species. The testing set is then comprised of windows centred around the insertion point, apex and four other points on a different set of 240 leaves (see Figure 6.7 for some examples). Having learnt a set of filters using the training set, each fixation in the training set is processed using the learnt filters to produce an n-dimensional vector of filter responses that can be used to describe the fixations. Filter response vectors are then generated for the fixations in the test set. A nearest-neighbour algorithm is used to classify the test fixations, whereby the nearest training vector to a test vector is found, and if the distance between them is less than some threshold, the test fixation is assigned the class of the training fixation, else it is classified as not bring a fixation.

For comparison, a set of filters based on the Leung-Malik and Root Filter Set filter banks is also used. These filters are evenly distributed throughout the parameter space, using the same parameter ranges as for the new method. The results of these with 16, 36 and 128 filters are given in Table 6.2, with

Figure 6.6: Heatmaps from eye-tracker data, indicating the insertion points and apices as fixations.



(a) Insertion points (b) Apices (c) Other windows

Figure 6.7: Examples of the three classes

values indicated the percentage of windows correctly classified. The ROC curve shown in Figure 6.8 was produced by varying the threshold used to perform the classification, for the sets of 16 filters. The accuracies given in Table 6.2 correspond to the threshold value which gave the best results for each set of filters.

Table 6.2: Results for experiment 1

Method	16 filters	36 filters	128 filters
New method	93.68	93.26	93.61
Filter bank	80.97	89.58	93.68

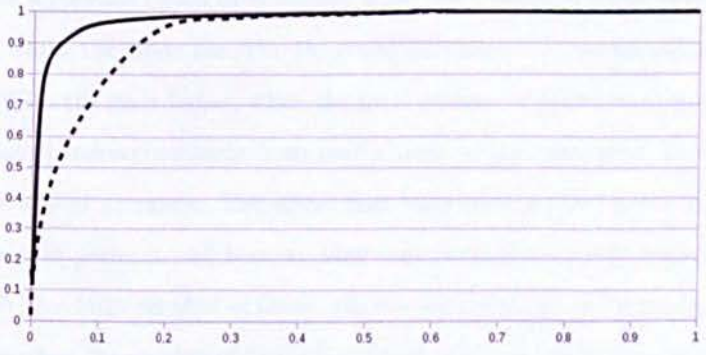


Figure 6.8: ROC curve for experiment 1. Solid line = our method, dashed line = standard filter bank.

Experiment 2. In our second experiment, we attempt to find the insertion point and apex on leaf from amongst 16 possible windows. This experiment more accurately reflects the final task, as it attempts to discover the most likely fixation points on each image. The training is performed as per experiment 1. To test, for each leaf image, the response vectors for windows centred around the insertion point, apex, and 14 other randomly selected points on the leaf are compared to the response vectors for the training set. The windows closest to an insertion point and apex from the training set are selected as being these respective points on the new leaf. Table 6.3 shows the results for this experiment.

Table 6.3: Results for experiment 2 (%)

Method	16 filters	36 filters	128 filters
Our method	91.25	95.00	95.08
Filter bank	77.92	91.76	94.58

Results. The results show that whilst both sets of filters achieve a similar, high level of accuracy when using a large number of filters, the set of filters learnt using the algorithm retain their quality when the number of filters is reduced to just 16, whilst the filters based on the standard filter banks perform significantly worse. With the filter banks, when the total number of filters is reduced, filters are removed indescriminately from useful areas of the parameter space, which causes a loss of accuracy. The filters that have been learnt by our algorithm, however, still perform well because they only occupy the useful regions in the space. With a large number of filters, there is a redundancy in these areas of the space, so when the number of filters is reduced,only this redundancy is removed, and do not lose any quality. This allows for the same level of accuracy whilst only requiring a fraction of the amount of processing.

6.2.4 Summary

For many practical applications, speed and efficiency are particularly important for a method for discovering the fixation points in an image, because large areas of the image need to be processed. By developing an algorithm that allows the learning of a small set of filters that is capable of distinguishing fixation points with as high accuracy as a larger filter bank, a step has been taken towards this. Further work in this area could explore how efficiency can be further improved by intelligently selecting only the best subset of our filter to use on any given part of the image. It may also be profitable to investigate the use of different methods of classification for improving the accuracy.

Beyond this, how the sequences for the fixations can be discovered need to be considered. One possibility may be the use of hidden Markov models, involving spatial and temporal information, as well as data generated by the filter set. With a system for accurate estimation and replication of the methods in which a human expert studies an image, the hope is to able to improve the efficiency and robustness of automated computer vision systems for performing such tasks, although further investigation will be needed into whether such information can indeed allow for more accurate leaf classification. Further to this, it is hoped that we may be able to discover new knowledge about how human expert achieve this task.

Chapter 7

Conclusions

This thesis has presented work covering a range of aspects associated with the computer-assisted identification and analysis of plant leaves. In Chapter 2, a comprehensive review was carried out of the work previously carried out in this field, identifying the challenges involved in this task, and those areas in which progress still needed to be made. Chapter 4 presented a number of new methods for the extraction and description of various components of the leaf. A comparative study of the most commonly used shape analysis techniques was also performed, demonstrating the strengths and weaknesses of each method. One of the challenges in this field is in coping with the high intra-species variation present in leaves, and in Chapter 5, new techniques are presented for doing so, in regards to the leaf macro-texture and margins. Also in Chapter 5, methods of combining the different leaf feature-sets were presented, including a method for automatically selecting which feature-sets are most appropriate to use when trying to classify a particular leaf. Chapter 6 offers a preliminary study into how botanists view leaves in contrast to non-botanists, including a look at how this information might be used to improve automated techniques.

7.1 Achievements

This work has made several achievements, with regards to the challenges involved, making great progress in this field.

Extraction of information. New methods have been presented for the extraction and description of the leaf macro- and micro-texture, margin characters, and the venation patterns, showing strong performance in comparison to existing techniques. Though there has been much work previously carried out on the analysis of leaf shape, there has been little in the way of comparison of the different techniques. Here, the merits of the most popular techniques have been highlighted and contrasted.

Dealing with intra-class variation. One of the key challenges associated with the identification of plant species relates to the often high intra-species variation. Here, methods have been developed for solving this problem, in regards to the leaf macro-texture and margins. The methods presented for the former of these could also be applied to many other machine-learning problems where intra-class variation is a significant issue.

Combining modalities. With many different components of the leaf available for use for classification, it was important to develop an appropriate method for combining the information. In particular, due to the wide variety of forms many leaf features can take, and the low inter-species variation found for some leaf features between some sets of species, there was a desire to be able to automatically select the best subset of modalities to use on a leaf-by-leaf basis. A method has been presented here, which achieves improved accuracy and reduced computational requirements by predicting the value of each feature-set when classifying a given leaf.

Understanding how botanists view leaves. Much can be learnt from the behaviour of experts which can be applied to the task here. Eye tracking technology has been used to establish the differences between how botanists and non-botanists view leaf images whilst performing a leaf recognition task. Further to this, preliminary work has been performed towards replicating the botanists' observation points, so that this information could be utilized in an automated system.

7.2 Future Work

The work presented in this thesis makes several advancements towards the goal of automated classification and analysis of leaves, however, there are still some areas in which further progess could be made. Whilst a relatively accurate method for the extraction of venation patterns was put forward in Chapter 4, as yet there is no suitable method for describing and comparing venations, allowing them to be used alongside the other leaf components when performing an identification. Again, the method would need to take into account the variation that is present, particularly since the number of secondary and tertiary veins will vary from leaf to leaf within a species.

The methods for leaf-shape analysis discussed in Section 4.1 are existing methods for general shape and morphometric analysis, rather than being entirely tailored to the leaf-specific task. Future work could include the development a shape analysis method that is more suitable to the subject matter here.

In Chapter 5, a framework was proposed for the selection and combination of different leaf feature-sets. This framework managed to improve the classification accuracy whilst reducing the number of feature-sets required, however there remain a number of extensions worthy of exploration. One such thing is whether simple, fast-to-extract descriptors can be reliably used to predict the value in

using the more complex descriptors, which require more processing to extract. This could greatly reduce the computational requirements. Further to this, it may be desirable to factor in each extraction/comparison's computational needs, when determining the feature-sets to use. From this, it may emerge that there is greater value in using many simple but fast methods, than the more advanced but costly ones, despite them producing better results when used individually.

Bibliography

[1] P. Abry, P. Goncalves, and J. L. Vehel. *Scaling, Fractals and Wavelets.* Wiley, Jan. 2009. 23

[2] D. C. Adams, F. J. Rohlf, and D. E. Slice. Geometric morphometrics: ten years of progress following the "revolution". *Italian Journal of Zoology*, 71(1):5–16, 2004. 20

[3] G. Agarwal, P. Belhumeur, S. Feiner, D. Jacobs, W. J. Kress, R. Ramamoorthi, N. A. Bourg, N. Dixit, H. Ling, and D. Mahajan. First steps toward an electronic field guide for plants. *Taxon*, 55(3):597, 2006. 10, 33, 37

[4] M. Andrade, S. J. Mayo, D. Kirkup, and C. Van Den Berg. Comparative morphology of population of *Monstera* Adans. (*Araceae*) from natural forest fragments in northeast Brazil using elliptic Fourier analysis of leaf outlines. *Kew Bulletin*, 63:193–211, 2008. 17, 52

[5] A. F. Atiya. Estimating the posterior probabilities using the k-nearest neighbor rule. *Neural computation*, 17(3):731–740, 2005. 101

[6] A. R. Backes and O. M. Bruno. Plant leaf identification using multi-scale fractal dimension. In *International Conference On Image Analysis And Processing*, pages 143–150. Springer Berlin / Heidelberg, 2009. 23, 28

[7] A. R. Backes, J. J. de Mesquita Sá Junior, R. M. Kolb, and O. M. Bruno. Plant species identification using multi-scale fractal dimension applied to images of adaxial surface epidermis. *Computer Analysis Of Images And Patterns*, 5702:680–688, 2009. 28

[8] A. R. Backes, W. N. Gonçalves, A. S. Martinez, and O. M. Bruno. Texture analysis and classification using deterministic tourist walk. *Pattern Recognition*, 43:685–694, 2010. 28

[9] P. Belhumeur, D. Chen, S. Feiner, D. Jacobs, W. Kress, H. Ling, I. Lopez, R. Ramamoorthi, S. Sheorey, S. White, and L. Zhang. Searching the world's herbaria: A system for visual identification of plant species. In *Computer Vision - ECCV 2008*, pages 116–129. Springer, 2008. 33

[10] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 509–522, 2002. 33

[11] A. Biem and S. Katagiri. Filter bank design based on discriminative feature extraction. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 485–488, 1994. 122

[12] P. Blignaut. Fixation identification: The optimum threshold for a dispersion algorithm. *Attention, Perception and Psychophysics*, 71(4):881–895, 2009. 124

[13] F. L. Bookstein. Size and shape spaces for landmark data in two dimensions. *Statistical Science*, 1(2):181–222, May 1986. 20

[14] O. M. Bruno, R. de Oliveira Plotze, M. Falvo, and M. de Castro. Fractal dimension applied to plant identification. *Information Sciences*, 178(12):2722–2733, 2008. 24

[15] M. Budka, B. Gabrys, and K. Musial. On accuracy of PDF divergence estimators and their applicability to representative data sampling. *Entropy*, 13:1229–1266, 2011. 61

[16] X. P. Burgos-Artizzu, A. Ribeiro, A. Tellaeche, G. Pajares, and C. Fernández-Quintanilla. Analysis of natural images processing for the extraction of agricultural elements. *Image and Vision Computing*, 28(1):138–149, Jan. 2010. 34, 36

[17] F. Canales and M. Chacón. Modification of the growing neural gas algorithm for cluster analysis. *Progress in Pattern Recognition, Image Analysis and Applications*, 4756:684–693, 2007. 126, 127

[18] D. Casanova, J. J. de Mesquita Sá Junior, and O. M. Bruno. Plant leaf identification using Gabor wavelets. *International Journal of Imaging Systems and Technology*, 19:236–243, 2009. 28

[19] L. Chen, B. Chen, and Y. Chen. Image feature selection based on ant colony optimization. *Advances in Artificial Intelligence*, pages 580–589, 2011. 105

[20] X. Chen, X. Hu, and X. Shen. Spatial weighting for bag-of-visual-words and its application in content-based image retrieval. *Advances in Knowledge Discovery and Data Mining*, 5476:867–874, 2009. 86

[21] S. C. Cheng, J. J. Jhou, and B. H. Liou. PDA plant search system based on the characteristics of leaves using fuzzy function. In *New Trends in Applied Artificial Intelligence*, number 4570 in LNAI, pages 834–844. Springer, 2007. 34

[22] Z. Chi, L. Houqiang, and W. Chao. Plant species recognition based on bark patterns using novel Gabor filter banks. In *Neural Networks and*

Signal Processing, 2003. Proceedings of the 2003 International Conference on, volume 2, pages 1035–1038, dec. 2003. 123

[23] J. Y. Clark. Plant identification from characters and measurements using artificial neural networks. In N. MacLeod, editor, *Automated taxon identification in systematics: theory, approaches and applications*, pages 207–224. CRC, 2007. 27

[24] J. Y. Clark. Neural networks and cluster analysis for unsupervised classification of cultivated species of *Tilia* (Malvaceae). *Botanical Journal of the Linnean Society*, 159:300–314, 2009. 27, 29, 30

[25] J. Clarke, S. Barman, P. Remagnino, K. Bailey, D. Kirkup, S. Mayo, and P. Wilkin. Venation pattern analysis of leaf images. *Lecture Notes In Computer Science*, 4292:427–436, 2006. 25

[26] P. Comon. Independent component analysis: A new concept? *Signal Processing*, 36:287–314, 1994. 25

[27] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *Workshop on Statistical Learning in Computer Vision, ECCV*, pages 1–22, 2004. 94, 98

[28] M. Das, R. Manmatha, and E. M. Riseman. Indexing flower patent images using domain knowledge. *IEEE Intelligent Systems*, 14(5):24–33, 1999. 30

[29] J. G. Daugman. Two-dimensional spectral analysis of cortical receptive field profiles. *Vision Research*, 20:847–856, 1980. 59, 125

[30] J. G. Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *J Opt Soc Am A*, 2:1160–1169, 1985. 58, 123

[31] T. A. Dickinson, W. H. Parker, and R. E. Strauss. Another approach to leaf shape comparisons. *Taxon*, pages 1–20, 1987. 37

[32] J.-X. Du, D.-S. Huang, X.-F. Wang, and X. Gu. Computer-aided plant species identification (CAPSI) based on leaf shape matching technique. *Transactions Institute Of Measurement And Control*, 28:275–284, 2006. 24, 33

[33] J.-X. Du, X.-F. Wang, and G.-J. Zhang. Leaf shape based plant species recognition. *Applied Mathematics and Computation*, 185:883–893, 2007. 22

[34] A. T. Duchowski. *Eye Tracking Methodology: Theory and Practice.* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007. 114

[35] J. Edmonds and R. M. Karp. Theoretical improvements in algorithmic efficiency for network flow problems. *Journal Of The ACM*, 19:248–264, 1972. 84

[36] B. Ellis, D. C. Daly, L. J. Hickey, K. R. Johnson, J. D. Mitchell, P. Wilf, and S. L. Wing. *Manual Of Leaf Architecture.* Cornell University Press, 2009. 27

[37] L. Feldkamp. Neural networks: Current applications, book review. *Proceedings of the IEEE*, 84(1):87, jan 1996. 123

[38] B. Fritzke. A growing neural gas network learns topologies. In *Advances in Neural Information Processing Systems*, pages 625–632. MIT Press, 1995. 123, 126

[39] H. Fu and Z. Chi. Combined thresholding and neural network approach for vein pattern extraction from leaf images. In *IEE Proceedings. Vision*

Image And Signal Processing, volume 153, pages 881–892. Institution of Electrical Engineers, 2006. 25

[40] K. Fukunaga and L. Hostetler. K-nearest-neighbor bayes-risk estimation. *Information Theory, IEEE Transactions on*, 21(3):285–293, 1975. 101

[41] N. Furuta, S. Ninomiya, N. Takahashi, H. Ohmori, and Y. Ukai. Quantitative evaluation of soybean leaflet shape by principal component scores based on elliptic Fourier descriptor. *Breeding Science*, 45:315–320, 1995. 17, 52

[42] E. Gage and P. Wilkin. A morphometric study of species delimitation in *sternbergia lutea* (Alliaceae, Amaryllidoideae) and its allies *s. sicula* and *s. greuteriana*. *Botanical Journal of the Linnean Society*, 158:460–469, 2008. 30, 37

[43] T. Gautama and M. M. V. Hulle. Self-organized feature extraction achieved with a parameterized filterbank. *Neural Processing Letters*, 10(2):131–137, 1999. 122

[44] S. Gebhardt, J. Schellberg, R. Lock, and W. Khbauch. Identification of broad-leaved dock (*Rumex obtusifolius* L.) on grassland by means of digital image processing. *Precision Agriculture*, 7(3):165–178, July 2006. 10, 35

[45] J. Geusebroek, A. Smeulders, and J. van de Weijer. Fast anisotropic gauss filtering. *IEEE Transactions on Image Processing*, 12:2003, 2002. 129

[46] C. Goodall. Procrustes methods in the statistical analysis of shape. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 285–339, 1991. 17

[47] R. Govaerts, P. Wilkin, L. Raz, and O. Téllez-Valdés. World checklist of Dioscoreaceae. The Board of Trustees of the Royal Botanic Gardens, Kew., 2010. 13

[48] S. Grigorescu, N. Petkov, and P. Kruizinga. Comparison of texture features based on Gabor filters. *Image Processing, IEEE Transactions on*, 11(10):1160–1167, oct 2002. 123

[49] X. Gu, J.-X. Du, and X.-F. Wang. Leaf recognition based on the combination of wavelet transform and Gaussian interpolation. *Advances In Intelligent Computing*, 3644:253–262, 2005. 28

[50] S. Gubatz, V. J. Dercksen, C. Brüss, W. Weschke, and U. Wobus. Analysis of barley (*Hordeum vulgare*) grain development using three-dimensional digital models. *The Plant Journal*, 52(4):779–790, 2007. 31

[51] A. Haigh, P. Wilkin, and F. Rakotonasolo. A new species of *Dioscorea* L. (Dioscoreaceae) from Western Madagascar and its distribution and conservation status. *Kew Bulletin*, 60:273–281, 2005. 20, 48

[52] R. M. Haralick, I. Dinstein, and K. Shanmugam. Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-3:610–621, 1973. 60, 62

[53] S. Haykin. *Neural Networks and Learning Machines*. Prentice Hall, 2009. 123

[54] D. J. Hearn. Shape analysis for the automated identification of plants from images of leaves. *Taxon*, 58:934–954, 2009. 12, 16, 17

[55] G. Heidemann. A neural 3-d object recognition architecture using optimized gabor filters. In *Proceedings of the International Conference on*

Pattern Recognition (ICPR 96), pages 70–74, Washington DC USA, 1996. IEEE Computer Society. 122

[56] J. Hemming and T. Rath. PA-Precision agriculture:: Computer-Vision-based weed identification under field conditions using controlled lighting. *Journal of Agricultural Engineering Research*, 78(3):233243, 2001. 35

[57] A. Hong, G. Chen, J. L. Li, Z. R. Chi, and D. Zhang. A flower image retrieval method based on ROI feature. *Journal of Zhejiang University-Science*, 5(7):764–772, 2004. 30

[58] M. Hu. Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, 8(2):179–187, 1962. 22

[59] L.-L. Huang, A. Shimizu, and H. Kobatake. Robust face detection using gabor filter features. *Pattern Recognition Letters*, 26:1641–1649, 2005. 59, 125

[60] Q. Huang, A. Jain, G. Stockman, and A. Smucker. Automatic image analysis of plant root structures. In *Proceedings 11th IAPR International Conference on Pattern Recognition, 1992. Vol.II. Conference B: Pattern Recognition Methodology and Systems*, pages 569–572, 1992. 31

[61] Z.-K. Huang, H. De-Shuang, J.-X. Du, Z.-H. Quan, and S.-B. Guo. Bark classification based on Gabor filter features using RBPNN neural network. In *International Conference on Neural Information Processing*, pages 80–87. Springer, 2006. 30

[62] P. M. Huff, P. Wilf, and E. J. Azumah. Digital future for paleoclimate estimation from fossil leaves? Preliminary results. *Palaios*, 18(3):266–274, June 2003. 36

[63] C. Im, H. Nishida, and T. L. Kunii. Recognizing plant species by normalized leaf shapes. In *Vision Interface*, volume 99, pages 19–21, 1999. 24

[64] R. Jacob and K. Karn. Eye tracking in human-computer interaction and usability research: Ready to deliver the promises. *The Mindś Eye: Cognitive and Applied Aspects of Eye Movement Research*, pages 573–603, 2003. 114

[65] R. J. Jensen, K. M. Ciofani, and L. C. Miramontes. Lines, outlines, and landmarks: Morphometric analyses of leaves of *Acer rubrum*, *Acer saccharinum* (Aceraceae) and their hybrid. *Taxon*, 51(3):475–492, 2002. 20, 48

[66] L. Journaux, M.-F. Destain, J. Miteran, A. Piron, and F. Cointault. Texture classifaction with generalized Fourier descriptors in dimensionality reduction context: An overview exploration. *Artificial Neural Networks In Pattern Recognition*, 5064:280–291, 2008. 62

[67] F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. In *International Conference on Computer Vision*, 2005. 87

[68] K. Kavukcuoglu, M. Ranzato, R. Fergus, and Y. LeCun. Learning invariant features through topographic filter maps. In *Proc. International Conference on Computer Vision and Pattern Recognition (CVPR'09)*, pages 1605–1612. IEEE, 2009. 122

[69] A. V. Ken Chatfield, Victor Lempitsky and A. Zisserman. The devil is in the details: an evaluation of recent feature encoding methods. In *British Machine Vision Conference*, pages 76.1–76.12, 2011. 87

[70] N. Kirchgessner, H. Scharr, and U. Schurr. Robust vein extraction on plant leaf images. In *2nd IASTED International Conference Visualisation, Imaging And Image Processing*, 2002. 26

[71] J. Kittler. Pattern recognition and signal processing. pages 41–60, 1978. 105

[72] D. Koller and M. Sahami. Toward optimal feature selection. In *International Conference on Machine Learning*, page 284292, 1996. 105

[73] F. P. Kuhl and C. R. Giardina. Elliptic Fourier features of a closed contour. *Computer graphics and image processing*, 18(3):236–258, 1982. 15

[74] H. W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2:83–97, 1955. 84

[75] N. Kumar, P. N. Belhumeur, A. Biswas, D. Jacobs, W. J. Kress, I. Lopez, and J. V. B. Soares. Leafsnap: A computer vision system for automatic plant species identification. In *Proceedings of the European Conference in Computer Vision (ECCV)*, pages 497–504, 2012. 10, 12, 34

[76] Y. Kurosawa. Incremental learning for feature extraction filter mask used in similar pattern classification. In *International Joint Conference on Neural Networks (IJCNN)*, pages 497–504, 2008. 122

[77] C. Lee and S. Chen. Classification of leaf images. *International Journal of Imaging Systems and Technology*, 16(1):15–23, 2006. 22

[78] T. Leung and J. Malik. Representing and recognising the visual appearance of materials using three-dimensional textons. *International Journal Of Computer Vision*, 43:7–27, 2001. 86, 129

[79] C. Lexer, J. Joseph, M. van Loo, G. Prenner, B. Heinze, M. W. Chase, and D. Kirkup. The use of digital image-based morphometrics to study the

phenotypic mosaic in taxa with porous genomes. *Taxon*, 58(2):349–364, 2009. 17, 52

[80] W. Li, K. Mao, H. Zhang, and T. Chai. Designing compact Gabor filter banks for efficient texture feature extraction. In *Control Automation Robotics Vision (ICARCV), 2010 11th International Conference on*, pages 1193 –1197, dec. 2010. 123

[81] Y. Li, Z. Chi, and D. D. Feng. Leaf vein extraction using independent component analysis. In *IEEE International Conference on Systems, Man and Cybernetics*, pages 3890–3984. IEEE, 2006. 25

[82] H. Ling and D. W. Jacobs. Shape classification using the inner distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29:286–299, 2007. 20

[83] M. Lipske. New electronic field guide uses leaf shapes to identify plant species. *Inside Smithsonian Research*, Winter, 2008. 34

[84] J. Liu and Y. Y. Tang. Adaptive image segmentation with distributed behavior-based agents. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(6):544–551, 1999. 71

[85] J. Liu, S. Zhang, and S. Deng. A method of plant classification based on wavelet transforms and support vector machines. *Emerging Intelligent Computing Technology and Applications*, 5754:253–260, 2009. 28

[86] W. Ma, H. Zha, J. Liu, X. Zhang, and B. Xiang. Image-based plant modeling by knowing leaves from their apexes. In *19th International Conference on Pattern Recognition, 2008*, pages 1–4, 2008. 29

[87] N. MacLeod. Generalizing and extending the eigenshape method of shape space visualization and analysis. *Paleobiology*, pages 107–138, 1999. 17

[88] S. Mannan, C. Kennard, and M. Husain. The role of visual salience in directing eye movements in visual object agnosia. *Current Biology*, 19:247–248, 2009. 124

[89] T. Martinetz and K. Shulten. A neural-gas network learns topologies. *Artificial Neural Networks*, pages 397–402, 1991. 121, 123

[90] T. McLellan and J. A. Endler. The relative success of some methods for measuring and describing the shape of complex objects. *Systematic Biology*, 47:264–281, 1998. 16, 23, 24, 27

[91] C. Meade and J. Parnell. Multivariate analysis of leaf shape patterns in Asian species of the *Uvaria* group (Annonaceae). *Botanical Journal Of The Linnean Society*, 143:231–242, 2003. 18, 51

[92] E. Megaw and J. Richardson. Eye movements and industrial inspection. *Applied Ergonomics*, 10:145–154, 1979. 114

[93] R. Mendona Ernesto Rego, A. Araujo, and F. de Lima Neto. Growing self-reconstruction maps. *Neural Networks, IEEE Transactions on*, 21(2):211 –223, feb. 2010. 124

[94] F. Mokhtarian. Silhoutte based isolated object recognition through curvature scale space. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(5):539–544, 1995. 19

[95] F. Mokhtarian and S. Abbasi. Matching shapes with self-intersection: application to leaf classification. *IEEE Transactions Image Processing*, 13:653–661, 2004. 19

[96] R. Mullen, D. Monekosso, S. Barman, P. Remagnino, and P. Wilkin. Artificial ants to extract leaf outlines and primary venation patterns. *Lecture Notes In Computer Science*, 5217:251–258, 2008. 25

[97] Y. Nam, E. Hwang, and D. Kim. CLOVER: a mobile content-based leaf image retrieval system. In *Digital Libraries: Implementing Strategies and Sharing Experiences*, number 3815 in LNCS, pages 139–148. Springer Berlin / Heidelberg, 2005. 34

[98] Y. Nam, E. Hwang, and D. Kim. A similarity-based leaf image retrieval scheme: Joining shape and venation features. *Computer Vision And Image Understanding*, 110:245–259, 2008. 26

[99] J. Neto, G. Meyer, D. Jones, and A. Samal. Plant species identification using elliptic Fourier leaf shape analysis. *Computers And Electronics In Agriculture*, 50:121–134, 2006. 17

[100] M. E. Nilsback and A. Zisserman. Delving into the whorl of flower segmentation. In *Proceedings of the British Machine Vision Conference*, volume 1, pages 570–579, 2007. 30

[101] D. Oakley and H. J. Falcon-Lang. Morphometric analysis of cretaceous (Cenomanian) angiosperm woods from the Czech Republic. *Review of Palaeobotany and Palynology*, 153(3-4):375–385, 2009. 31

[102] N. Otsu. A threshold selection method from gray level histograms. *IEEE Trans. Systems, Man and Cybernetics*, 9:62–66, 1979. 57, 72

[103] J. Pan and Y. He. Recognition of plants by leaves digital image and neural network. In *2008 International Conference on Computer Science and Software Engineering*, pages 906–910, Wuhan, China, 2008. 36

[104] J. Park, E. Hwang, and Y. Nam. Utilizing venation features for efficient leaf image retrieval. *Journal of Systems and Software*, 81(1):71–82, 2008. 26

[105] E. Parzen. On estimation of a probability density funstion and mode. *Annals Of Mathematical Statistics*, 33:1065–1076, 1962. 83, 92

[106] C. Patterson. Morphological characters and homology. In K. Joysey and A. Friday, editors, *Problems of phylogenetic reconstruction*, volume 576, pages 21–74. Academic Press, 1982. 21

[107] E. J. Pauwels, P. M. de Zeeum, and E. B. Ranguelova. Computer-assisted tree taxonomy by automated image recognition. *Engineering Applications Of Artificial Intelligence*, 22(1):26–31, Feb. 2009. 22

[108] R. d. Plotze, M. Falvo, J. G. Padua, L. C. Bernacci, M. L. C. Vieira, G. C. X. Oliveira, and O. Martinez. Leaf shape analysis using the multiscale Minkowski fractal dimension, a new morphometric method: A study with *Passiflora* (Passifloraceae). *Canadian Journal Of Botany*, 83(3):287–301, 2005. 23, 24, 26

[109] L. Qi, Q. Yang, G. Bao, Y. Xun, and L. Zhang. A dynamic threshold segmentation algorithm for cucumber identification in greenhouse. In *2nd International Congress on Image and Signal Processing, 2009*, pages 1–4, 2009. 36

[110] E. Ramos and D. S. Fernandez. Classification of leaf epidermis microphotographs using texture features. *Ecological Informatics*, 4:177–181, 2009. 28

[111] T. Randen and J. Husoy. Filtering for texture classification: a comparative study. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(4):291–310, apr 1999. 123

[112] E. Rashedi, H. Nezamabadi, and S. Saryazdi. A simultaneous feature adaptation and feature selection method for content-based image retrieval systems. *Knowledge-Based Systems*, 39:85–94, 2013. 105

[113] T. S. Ray. Landmark eigenshape analysis: homologous contours: leaf shape in syngonium (Araceae). *American Journal of Botany*, 79:69–76, 1992. 17

[114] L. W. Renniger, P. Verhese, and J. Coughlan. Where to look next? eye movements reduce local uncertainty. *Journal of Vision*, 7, 2007. 114

[115] D. C. Richardson, M. J. Spivey, and W. G. *Eye-Tracking: Characteristics and Methods Eye-Tracking: Research Areas and Applications.* Informa Healthcare, 2004. 114

[116] D. L. Royer, J. C. McElwain, J. M. Adams, and P. Wilf. Sensitivity of leaf size and shape to climate within *Acer rubrum* and *Quercus kelloggii*. *New Phytologist*, 179(3):808–817, 2008. 11

[117] D. L. Royer and P. Wilf. Why do toothed leaves correlate with cold climates? Gas exchange at leaf margins provides new insights into a classic paleotemperature proxy. *International Journal of Plant Sciences*, 167(1):11–18, Jan. 2006. 27

[118] D. L. Royer, P. Wilf, D. A. Janesko, E. A. Kowalski, and D. L. Dilcher. Correlations of climate and plant ecology to leaf size and shape: potential proxies for the fossil record. *American Journal of Botany*, 92(7):1141–1151, July 2005. 26, 36

[119] Y. Rubner, L. J. Guibas, and C. Tomasi. The earth mover's distance, multi-dimensional scaling, and color-based image retrieval. In *ARPA Image Understanding Workshop*, pages 661–668, 1997. 83, 84

[120] K. Rumpunen and I. V. Bartish. Comparison of differentiation estimates based on morphometric and molecular data, exemplified by various leaf shape descriptors and RAPDs in the genus *Chaenomeles*. *Taxon*, 51:69–82, 2002. 27

[121] W. J. Ryan, A. T. Duchowski, E. A. Vincent, and D. Battisto. Match-moving for area-based analysis of eye movements in natural tasks. In *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*, ETRA 10, pages 235–242, New York NY USA, 2010. ACM. 114

[122] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions On Acoustics, Speech and Signal Processing*, 26(1):43–49, 1978. 67, 68

[123] S. Salvador and P. Chan. Toward accurate dynamic time warping in linear time and space. *Intelligent Data Analysis*, 11(5):561–580, 2007. 68, 69

[124] R. Sandler and M. Lindenbaum. Gabor filter analysis for texture segmentation. In *Computer Vision And Pattern Recognition Workshop*, page 178. IEEE, 2006. 59, 125

[125] D. Šeatović. A segmentation approach in novel real time 3D plant recognition system. In *Proceedings of the 6th international conference on Computer vision systems*, pages 363–372, 2008. 10, 35

[126] S. Sempena, N. U. Maulidevi, and P. R. Aryan. Human action recognition using dynamic time warping. In *International Conference on Electrical Engineering and Informatics*. IEEE, 2011. 67

[127] K. Sikka, T. Wu, J. Susskind, and M. Bartlett. Exploring bag of words architectures in the facial expression domain. *European Conference on Computer Vision*, pages 250–259, 2012. 87

[128] J. Sivic and A. Zisserman. Google video: A text retrieval approach to object matching in videos. In *International Conference On Computer Vision*, volume 2, pages 1470–1477, 2003. 83, 86

[129] P. Soille. Morphological image analysis applied to crop field mapping. *Image and Vision Computing*, 18(13):1025–1032, Oct. 2000. 35

[130] Y. Song, R. Wilson, R. Edmondson, and N. Parsons. Surface modelling of plants from stereo images. In *Sixth International Conference on 3-D Digital Imaging and Modeling, 2007. 3DIM '07.*, pages 312–319, 2007. 29

[131] K. Sparck-Jones and R. M. Needham. Automatic term classifications and retrieval. *Information Storage And Retrieval*, 4:91–100, 1968. 85

[132] M. Teague. Image analysis via the general theory of moments. *J. Opt. Soc. Am*, 70(8):920–930, 1980. 22

[133] C. H. Teng, Y. T. Kuo, and Y. S. Chen. Leaf segmentation, its 3D position estimation and leaf classification from a few images with very close viewpoints. *Image Analysis and Recognition*, pages 937–946, 2009. 29

[134] K. Thiele. The holy grail of the perfect character: the cladistic treatment of morphometric data. *Cladistics*, 9:275–304, 1993. 21

[135] P. Tirilly, V. Claveau, and P. Gros. Language modeling for bag-of-visual words image categorization. In *International Conference on Content-based Image and Video Retrieval (CIVR)*, pages 249–258. ACM, 2008. 86

[136] M. Turkan, B. Dulek, I. Onaran, and A. Cetin. Human face detection in video using edge projections. In *Visual Information Processing*. SPIE, 2006. 67

[137] M. Varma and A. Zisserman. A statistical approach to material classification using image patch exemplars. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31:2032–2047, 2009. 86

[138] X. Wang, D. Huang, J. Du, H. Xu, and L. Heutte. Classification of plant leaf images with complicated background. *Applied Mathematics and Computation*, 205(2):916–926, Nov. 2008. 22

[139] Z. Wang, Z. Chi, and F. Dagan. Shape based leaf image retrieval. *Vision, Image And Signal Processing*, 150:34–43, 2003. 18, 27

[140] Z. Wang, Z. Chi, F. Dagan, and Q. Wang. Leaf image retrieval with shape features. *Advances In Visual Information Systems*, 1929:41–52, 2000. 18

[141] M. Wedel and R. Pieters. A review of eye-tracking research in marketing. *Review of Marketing Research*, 4:123–147, 2008. 114

[142] R. White, H. C. Prentice, and T. Verwist. Automated image acquisition and morphometric description. *Canadian Journal of Botany*, 66:450–459, 1988. 16

[143] S. White, S. Feiner, and J. Kopylec. Virtual vouchers: Prototyping a mobile augmented reality user interface for botanical species identification. In *IEEE Symposium on 3D User Interfaces*, pages 119–126, 2006. 10, 34

[144] P. Wilkin. A morphometric study of *Dioscorea quartiniana* (Dioscoreaceae). *Kew Bulletin*, 54:1–18, 1999. 30, 37

[145] S. G. Wu, F. S. Bao, E. Y. Xu, Y.-X. Wang, Y.-F. Chang, and Q.-L. Xiang. A leaf recognition algorithm for plant classification using probabilistic neural network. In *IEEE International Symposium on Signal Processing and Information Technology*. IEEE, 2007. 22

[146] L. Ye and E. Keogh. Time series shapelets: A new primitive for data mining. In *IEEE International Conference on Knowledge Discovery and Data Mining*, pages 947–956. ACM, 2009. 18

[147] Y. Yoshioka, H. Iwata, R. Ohsawa, and S. Ninomiya. Anaylsis of petal shape variation of *Pimula sieboldii* by elliptic Fourier descriptors and principal component analysis. *Annals of Botany*, 94:1657-1664, 2004. 30, 52

[148] J. P. Young, T. A. Dickinson, and N. G. Dengler. A morphometric analysis of heterophyllous leaf development in *Ranunculus flabellaris*. *International Journal Of Plant Species*, 156:590–602, Sept. 1995. 20

[149] L. Yu and H. Liu. Efficient feature selection via analysis of relevance and redundancy. *Journal of Machine Learning Research*, 5:1205–1224, 2004. 105

[150] M. Zelditch, D. Swiderski, and W. Fink. Discovery of phylogenetic characters in morphological data. In J. Wiens, editor, *Phylogenetic Analysis of Morphological Data*, pages 37–83. Smithsonian Institution Press, Washington and London, 2000. 21

[151] G. Zeng, S. T. Birchfield, and C. E. Wells. Rapid automated detection of roots in minirhizotron images. *Machine Vision and Applications*, 21(3):309–317, 2008. 31