Computationally Efficient Features for Leaf Image Classification

A Preprint

Jayani P.G. Lakshika [*]
Department of Statistics
University of Sri Jayewardenepura
Nugegoda, Sri Lanka
jayanilakshika76@gmail.com

Thiyanga S. Talagala
Department of Statistics
University of Sri Jayewardenepura
Nugegoda, Sri Lanka
ttalagala@sjp.ac.lk

March 22, 2021

Abstract

Enter the text of your abstract here.

***K*eywords** scagnostics · image processing · bloo · these are optional and can be removed

1 Introduction

Leaf identification is becoming very popular in classifying plant species. Leaf contains significant features that can help people to identify and classify the plant species by looking at it. In medical perspective, medicinal plants are usually identified by practitioners based on years of experience through sensory or olfactory senses. The other method of recognizing these plants involves laboratory-based testing, which requires trained skills, data interpretation which is costly and time-intensive. Automatic ways to identify medicinal plants are useful especially those that are lacking experience in medicinal plant recognition. In this process, image processing, and feature extraction have more weight, because they are the initial step in identification.

Image processing is a crucial step in plant classification. The main aim of image processing is to improve the leaf image by removing undesired distortion (Waldchen and Mader 2018). Therefore processed leaf image can be used to extract features. Image segmentation (Anantrasirichai, Hannuna, and Canagarajah 2017), image orientation, cropping, grey scaling, binary thresholding, noise removal, contrast stretching, threshold inversion, image normalization (Aakif and Khan 2015), and edge recognition (Azlah et al. 2019; Goyal, Kapil, and Kumar 2018; Kalyoncu and Toygar 2014) are some of image processing techniques applied in recent research. Image processing steps can be applied parallel or individually or several times until the quality of leaf image reaches satisfactory level.

Most challenging part is to extract distinctive leaf features. Therefore most of the time research more focused on neural network models like CNN (Wu et al. 2007; Azlah et al. 2019; Herdiyeni and Wahyuni 2012) which are complicated and hard to understand what happening inside the algorithm. Feature extraction refers to taking measurements, geometric or otherwise, of possibly segmented, meaningful regions in the image (Waldchen and Mader 2018). A digital image is merely a collection of pixels represented as large matrices of integers corresponding to intensities of colors at different positions of the image (Gonzalez and Woods 2006). The main aim of feature extraction is to reduce dimensionality of this information by obtaining patterns of leaf images. In generally shape, colors, and texture contain these patterns. In this paper other than existing features, we are introducing scagnostic features, correlation of cartesian coordinate, and number of minimum and maximum points as new features.

[*] Use footnote for providing further information about author (webpage, alternative address)—*not* for acknowledging funding agencies. Optional.

The paper is organized as follows. In section 2, we briefly review the works on features of leaves that use in leaf classification, and image processing techniques that have to follow before extract the features. The next section, 3, discuss the datasets that we use to do the experiment. Steps of image processing of leaf images describes in section 4. In section 4, discuss about feature extraction in in-detail because features are highly influenced by the plant species to be classified. The next section, 5, show the experimental results for two existing datasets. The details about software and packages that use to extract the features are discussed in section 6. In section 7, discuss about visualization of leaf images in the feature space. Some discussion about the outputs and concluding remarks are given in last section.

2 Literature Review

3 Data

There are two leaf datasets of two different countries are used for the experiment, because to generalize the features we have to have leaf images collected by different countries in different methods like scans, pseudo-scans and photography.

3.1 Flavia Leaf Image Dataset

The Flavia dataset contains 1907 leaf images. There are 32 different species and each have 50-77 images. Scanners and digital cameras are used to acquire the leaf images on plain background. The isolated leaf images contain blades only, without petiole. These leaf images are collected from the most common plants in Yangtze, Delta, China (Waldchen and Mader 2018). Those leaves were sampled on the campus of the Nanjing University and the Sun Yat-Sen arboretum, Nanking, China (Waldchen and Mader 2018). (`https://sourceforge.net/projects/flavia/files/Leaf%2520Image%2520Dataset/`)

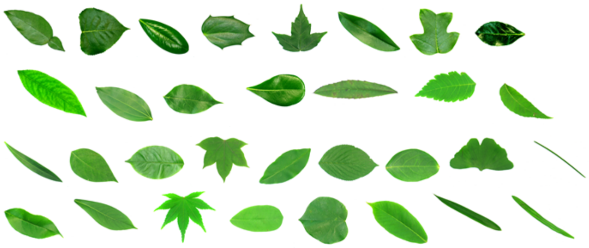


Figure 1: Sample of Flavia dataset

3.2 Swedish Leaf Image Dataset

The Swedish dataset contains 1125 images. The images of isolated leaf scans on a plain background of 15 Swedish tree species, with 75 leaves per species. This dataset has been captured as part of a joined leaf classification project between the Linkoping University and the Swedish Museum of Natural History (Waldchen and Mader 2018). (`https://www.cvl.isy.liu.se/en/research/datasets/swedish-leaf/`)

4 Image Processing

4.1 Introduction

The image processing receives an image as input and generates a modified image as an output which is suitable for better morphological analysis (Goyal, Kapil, and Kumar 2018), feature extraction. Image processing is an essential step to reduce noise, background subtraction and content enhancement in the identification process (Goyal, Kapil, and Kumar 2018).

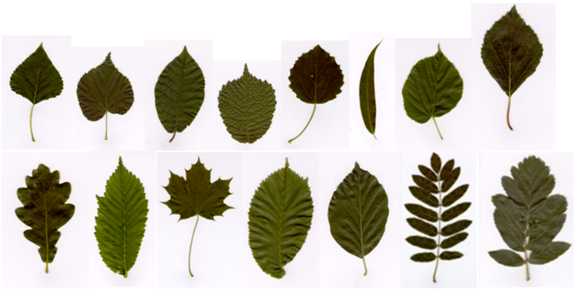


Figure 2: Sample of Swedish leaf dataset

4.2 Image Processing Workflow

In identifying a plant species, leaves are considered as a dominant feature. Image preprocessing plays a vital role, because features have to be clearly found after preprocessing. The main idea of preprocessing is to remove any kind of external noises present in an image.

To improve the leaf image, a series of operations are followed. This research includes operations like converting BGR image to RGB, gray scaling, Gaussian filtering, binary thresholding, remove stalk, close holes, and image resizing. Some of these steps can be applied for the necessary cases like applying remove stalk to the leaf images which only have a stalk.

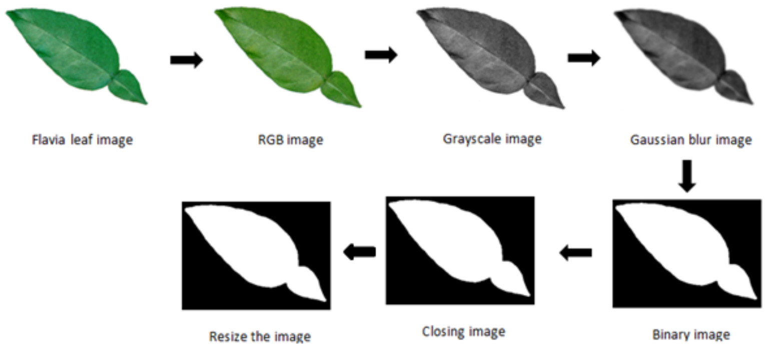


Figure 3: Image processing

For the development of the system proposed the focus was on the leaf which has a simple arrangement.

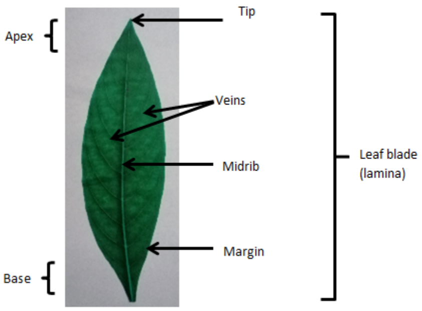


Figure 4: Leaf with simple arrangement

Dataset	Original Image Step	Number of leaf images
Flavia dataset	Step 1	1907
Swedish Leaf Image Dataset	Step 1	975

Table 1: Number of leaf images used in the algorithm

4.2.1 Step 1: Converting BGR Image to RGB

BGR (Blue-Green-Red) and RGB(Red-Green-Blue) are conventions for the order of the different colour channels. They are not colour spaces. When converting BGR image to RGB, there is no any computations, just switches around the order. There is a difference in OpenCV and Matplotlib in pixel ordering. OpenCV follows BGR order while Matplotlib follows RGB order. Therefore when we want to display an image which loaded from OpenCV using Matplotlib functions, have to convert it to RGB mode.

4.2.2 Step 2: Grayscaling

Grayscaling is the process of converting an image to shades of gray from other colour spaces like RGB. Gray conversion of the image is implemented to optimize the contrast and intensity of images (Goyal, Kapil, and Kumar 2018). There are some importance of grayscaling:

- Dimension reduction

As an example, there are three colour channels in RGB images and has three dimensions. But in gray scaled images only have one dimension.

- For other algorithms to work

There are many algorithm customized to work only on gray scaled images. Eg: Haralick texture features calculation works only on gray scaled images.

- Reduced model complexity

Consider an example of training neural article on RGB images of 10x10x3 pixel. The input layer will have 300 input nodes. Whereas for gray scaled images the same neural network will need only 100 input node.

4.2.3 Step 3: Image Smoothing

Image smoothing is also known as image blurring in OpenCV. Image smoothing refers to making the image less clear or distinct. Image smoothing is done with the help of various low pass filter kernels.

Advantages of Smoothing

- Smoothing techniques help in noise removal. Noise of an image is considered as high pass signal which can be occurred because of the source (camera sensor). Therefore restrict noise by using the low pass kernel.
- Helps in smoothing image.
- To remove low intensity edges.
- Helps in hiding the details when necessary. E:g:- To hide the face of the victim in police cases.

There are many image smoothing techniques that are available in OpenCV. For our research we used Gaussian filtering as the image smoothing technique.

4.2.4 Step 4: Gaussian Filtering (Gaussian Blurring/Gaussian Smoothing)

Gaussian function is used to blur the image. It is a linear filter which is done by using the functions in OpenCV. By specifying the width and height of the Gaussian kernel that must be positive and odd, and specifying the kernel standard deviation along x and y-axis, Gaussian smoothing is established in OpenCV. When the kernel standard deviation along x-axis is specified, kernel standard deviation along y-axis is taken as equal to the the kernel standard deviation along x-axis. But if both kernel standard deviation are given as zeros, they are calculated by using the kernel size. In our research the width and height of the kernel is defined as 55 and the kernel standard deviation along x-axis is assigned as zero.

4.2.5 Step 5: Binary Thresholding

Thresholding is a segmentation technique that is used to separate foreground from its background. In thresholding technique, the pixel values are assigned by using the threshold value. By comparing each pixel value with the threshold value, the thresholding technique is worked.If the pixel value is smaller than the threshold value, the pixel value is set as 0, and if not the pixel value is set to a maximum value which is generally 255. Thersholding technique is done on grayscale images in Computer Vision.

We used Otsu's binarization which is an adaptive thresholding after Gaussian filtering to convert colour images to the binary images. In Otsu's method, the threshold value is determined automatically. The algorithm of Otsu's method finds the optimal threshold value which is chosen arbitrary.

Otsu's Thresholding

Nobuyuki Otsu is the investor of Otsu's method which is defined for a gray scale histogram h_I of an input image I. To segment an image I into two subsets of pixels Otsu's method calculates an optimal threshold τ. Image I is defined on a regular carrier Ω containing $|\Omega|$ pixel locations.

The algorithm maximizes the variance σ^2 between the two subsets (Within-class-variance) to find the threshold τ.

$$\sigma^2 = P_1(\mu_1 - \mu)^2 + P_2(\mu_2 - \mu)^2 = P_1 P_2(\mu_1 - \mu_2)^2$$

where μ is the mean of the histogram, μ_1 and μ_2 are the mean values of first and second subset respectively, P_1 and P_2 are the corresponding probabilities of the two clusters, defined by

$$P_1 = \frac{\sum_{i=0}^{u} h_I(i)}{|\Omega|}$$

$$P_2 = \frac{\sum_{i=u+1}^{255} h_I(i)}{|\Omega|}$$

Where u is the candidate threshold and the maximum gray level (G_{max}) is assumed as 255. To find optimal threshold τ for segmenting image I, all candidate thresholds are evaluated this way.

The algorithm of Otsu's method is defined as follows,

```
Compute the histogram of the grayscale image
Set the histogram variance $S_{max} = 0$
while $u < G_{max}$ do
    Compute $\sigma^2 = P_1 P_2(\mu_1 - \mu_2)^2$
    if $\sigma^2 > S_{max}$ then
        $S_{max} = \sigma^2$
        $\tau = u$
    end if
    Set $u = u + 1$
end while
```

Table 2: Otsu's method

4.2.6 Step 6: Image Resizing

There are two different leaf image datasets (Flavia, Swedish) which have different sizes. To compare the results on different datasets, to improve the memory storage capacity and to reduce computational complexity the leaf images are resized to a fixed resolution. In our study, the leaf images have been resized to [1600 x 1200px] which is the size of Flavia leaf images.

Other than the main image processing techniques, the following two techniques are applied in some or all cases as an image processing techniques after image thresholding.

*) **Remove Stalk**

Remove the petiole (stalk) of leaf image is another version of thresholding process. Thresholding is applied after finding the sure foreground area. To find the sure foreground area, distance transform technique is used. Binary image is used as the input of distance transform technique. In distance transform technique, image is created by assigning a number for each object pixel that corresponds to the distance to the nearest background pixel. The distance is calculated using the euclidean distance (1). After finding the sure foreground area, Otsu's binarization is applied again as the thresholding technique.

$$Euclideandistance = \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2} \tag{1}$$

where n = n-space

q_i, p_i = Euclidean vectors, starting from the origin of the space

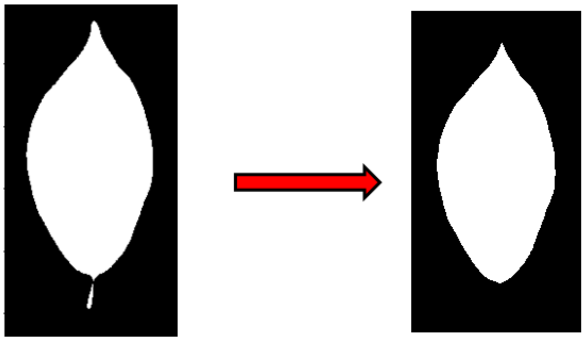


Figure 5: Remove stalk

*) **Closing Holes**

Closing holes is followed by noise removal technique which demonstrates the closing effect. Closing effect is used to remove small holes inside the foreground objects. Closing holes is a result of morphological transformation. morphological transformation are the operations based on image shape. Binary image is used as one input in morphological transformation. Kernel or structuring element which decides the nature of the operation is used as the second input. There are two morphological operators as Erosion and Dilation. Closing is a variant form of morphological operators which is used in closing holes process. Closing is also know as Dilation followed by Erosion.

Erosion

The basic idea of Erosion is that erodes away the boundaries of foreground object. Since the input is binary image, a pixels in the original image is either 1 or 0. If all the pixels under the kernel is 1, a pixel of original image is considered as 1, otherwise made to zero (eroded). Which means that depending upon the size of the kernel all pixels near boundary will be discarded. Therefore the thickness or size of the foreground object decreases (White region of the image decreases).

Dilation

Opposite of erosion is defined as Dilation. If at least one pixel under the kernel is 1, the pixel element is 1 in Dilation. It tends to increase the foreground of the image or the white region of the object.

Noise removal is that a technique of erosion is followed by Dilation. In erosion white noise is removed and shrinks the object (dilate) which doesn't come back. But in closing holes approach the white area is increased.

5 Features

In identification of plant species by using leaf images, features of the leaves play a main role, because each leaf posses unique feature that it make different from other. In previous research (Azlah et al. 2019; Jeon and

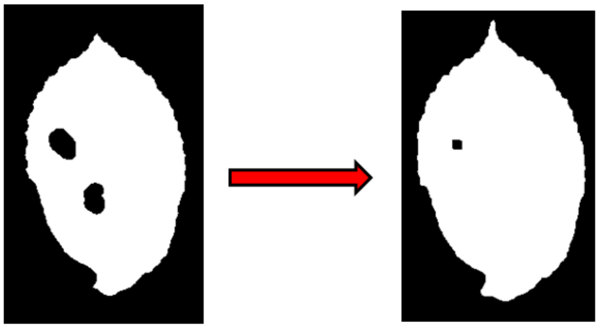


Figure 6: Closing holes

Rhee 2017), let the algorithm like CNN to extract features by itself and do the classification. Therefore it is so hard to interpret and generalize the features. We introduced pre-calculate features which can be easy to interpret and generalize. They are also computational efficient. Mainly we focused on four types of features of leaf images as Shape features, Texture features, Color features and Scagnostics features. We identified altogether 52 features.

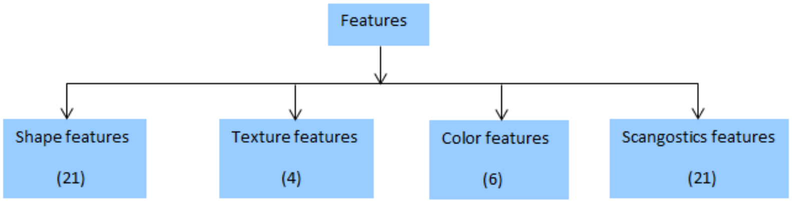


Figure 7: Example of geometric transformation

5.1 Shape Features

When identifying real-world objects, the shape is known as an essential sign for humans. A shape measure is a quantity, which relates to a particular shape characteristic of an object in general (Waldchen and Mader 2018).

The main geometric transformation are rotation reflection, scaling and translation (see figure 8). In my research, the defined shape descriptors (shape features) are invariant to the rotation and reflection. But some limitations are applied in translation and scaling.

The finding contour function doesn't work when the center of the leaf image is not in the contour. Therefore if the translation is applied away from the contour 9, then function can't identify the contour. Inappropriate scaling (see figure 10) also arises problems in the calculation. If the leaf image is really small, the function also hard to recognize the contour. Therefore taking the closest photo of the leaf images by keeping them in the center of the white paper is more suitable.

5.2 Contours

Simply contour (see figure 11) is a curve joining all the continuous points (along the boundary), having the same colour or intensity.

Shape descriptors can be classified into two main categories as contour-based and region-based. Contour-based descriptors extract shape features solely from the contour of a shape (Waldchen and Mader 2018). Whereas, region-based descriptors obtain shape features from the whole region of a shape (Waldchen and

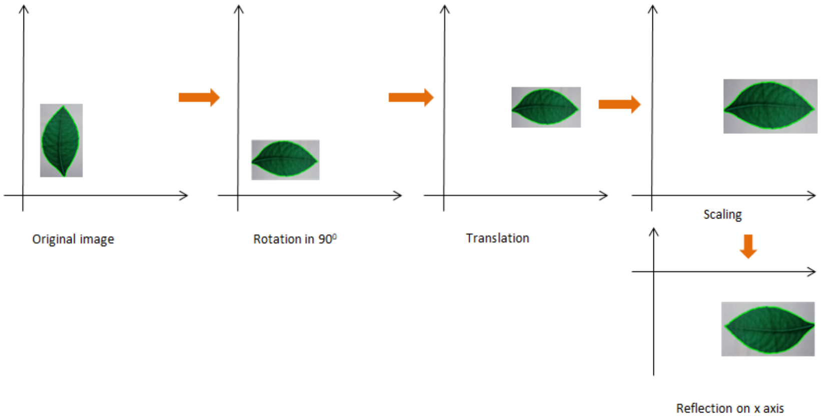


Figure 8: Example of geometric transformation

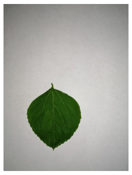


Figure 9: Inappropriate translation

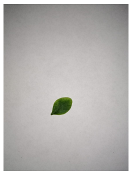


Figure 10: Inappropriate scaling

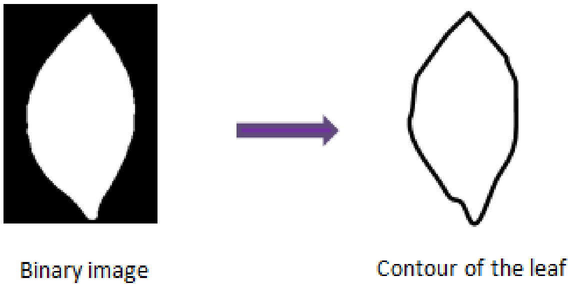


Figure 11: Extract contour of the leaf image

Mader 2018). In addition, there also exist some methods, which cannot be classified as either contour-based or region-based (see figure 12).

Through this research, we restricted our research to the following shape features to identify leaf images (see 3).

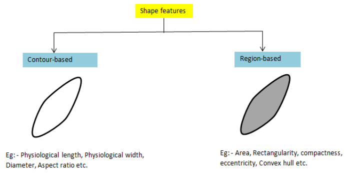

Shape features
Contour-based
Region-based
Eg:- Physiological length, Physiological width, Diameter, Aspect ratio etc.
Eg:- Area, Rectangularity, compactness, eccentricity, Convex hull etc.

Figure 12: Categorization of shape features

5.3 Diameter

Diameter is defined as the longest distance between any two points on the margin of the leaf (Waldchen and Mader 2018).

To calculate the diameter of the leaf image, firstly we have to find the contour of the leaf image. Then we have to select all pair of contour points and measure the Euclidean distance (2) between the two points separately. Finally have to find the maximum distance among the calculated distances. (see figure 17)

$$d\left(A,B\right)=\sqrt{\sum_{i=1}^{n}\left(q_i-p_i\right)^2} \tag{2}$$

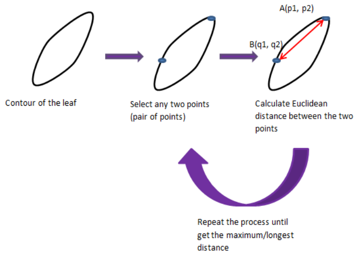

A(p1, p2)
B(q1, q2)
Contour of the leaf
Select any two points (pair of points)
Calculate Euclidean distance between the two points
Repeat the process until get the maximum/longest distance

Figure 13: Logic behind calculation of diameter

5.4 Physiological length and Physiological width

There are horizontal, vertical and rotated leaf images in the datasets (Flavia, Swedish, Actual and Kaggle). Kaggle leaf image dataset is only consist of horizontal and vertical leaf images. Therefore straight bounding rectangle is enough to extract Physiological length and Physiological width of leaf images.

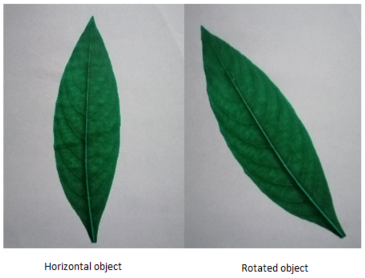


Figure 14: Straight(Horizontal or vertical) and rotated leaf image in Actual leaf image dataset

But straight bounding rectangle of rotated image doesn't give the correct values for Physiological length and Physiological width of leaf images.

To solve this problem, we considered rotated rectangle rather than bounded rectangle in computing shape features of angled images.

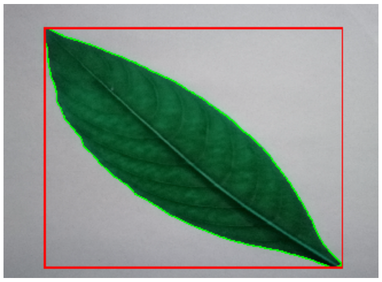


Figure 15: Bounded rectangle of rotated leaf image in Actual leaf image dataset

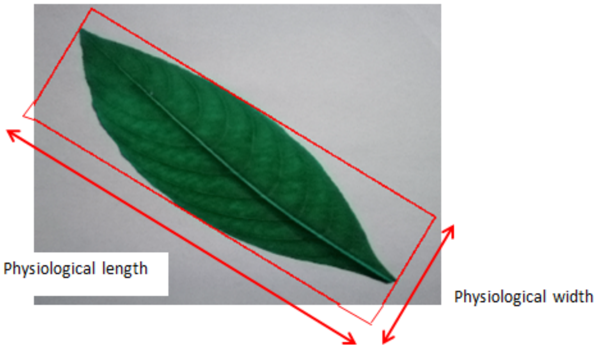


Figure 16: Rotated rectangle of angled leaf image in Actual leaf image dataset

There are two types of bounding rectangles.

- Straight Bounding Rectangle

This is a straight rectangle which doesn't consider the rotation of the object. Therefore area of the bounding rectangle doesn't minimize.

- Rotated Rectangle

This bounding rectangle is drawn with minimum area. Therefore the rotation of the object is also considered.

5.5 Area

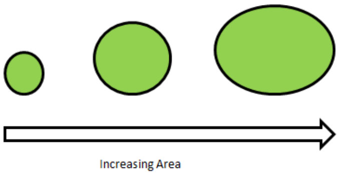


Figure 17: Area

5.6 Roundness/ Circularity

5.7 Compactness

5.8 Eccentricity

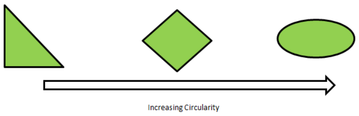


Figure 18: Circularity

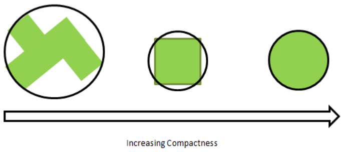


Figure 19: Compactness

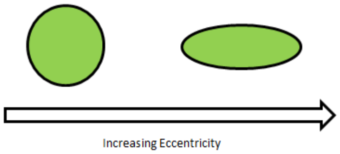


Figure 20: Eccentricity

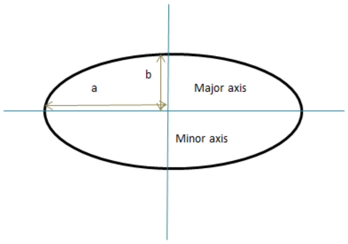


Figure 21: Ellipse

$$Eccentricity = \sqrt{1 - \frac{b^2}{a^2}} \tag{3}$$

5.9 Convexity

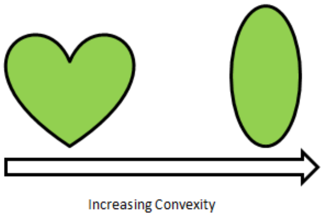


Figure 22: Convexity

5.10 Convex hull

The more details about convex hull can be found in the convex hull under scagnostics features.

5.11 Number of Minimum and Maximum Points

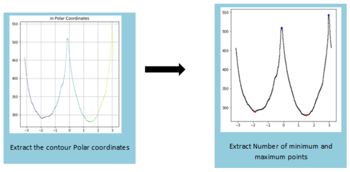
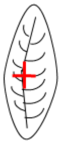
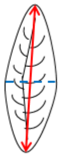
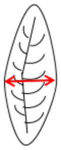
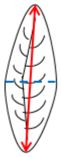


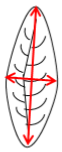
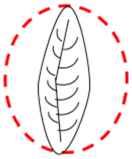
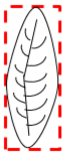
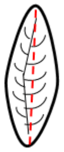
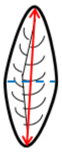
Figure 23: Minimum and Maximum Points

Table 3: Definitions of shape features

Shape feature	Description	Pictogram	Formula
Centroid	Represents the coordinates of the leaf's geometric center		
Physiological length/ Major axis length (L)	Line segment connecting the base and the tip of the leaf		
Physiological width/ Minor axis length (W)	Maximum width that is perpendicular to the major axis		
Diameter (D)	Longest distance between any two points on the margin of the origin		
Area (A)	Number of pixels in the region of the leaf		
Perimeter (P)	Summation of the distance between each adjoining pair of pixels around the border of the leaf		

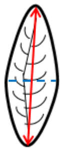
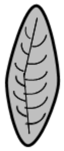
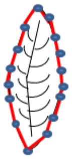
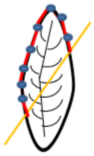
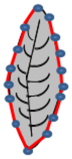
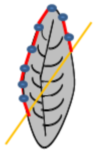
Continued on next page

Table 3 – continued from previous page

Shape feature	Description	Pictogram	Formula
Aspect ratio (AR)	Ratio of physiological length to physiological width		$AR = \frac{L}{W}$
Roundness/ Circularity (R)	Illustrate the difference between a leaf and a circle		$R = \frac{4\pi A}{P^2}$
Compactness	Ratio of the perimeter over the leaf's area		$C = \frac{P^2}{A}$
Rectangularity (N)	Represents how rectangle a shape is, i:e: how much it fits its minimum bounding rectangle		$N = \frac{A}{LW}$
Eccentricity (E)	Ratio of the distance between foci of the ellipse (f) and major axis length (a)		$E = \frac{f}{a}$
Narrow factor (NF)	Ratio of the diameter over the physiological length		$NF = \frac{D}{L}$
Perimeter ratio of diameter (P_D)	Ratio of the perimeter to the dimeter		$P_D = \frac{P}{D}$

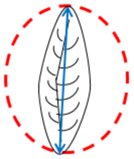
Continued on next page

Table 3 – continued from previous page

Shape feature	Description	Pictogram	Formula
Perimeter ratio of Major axis length (P_L)	Ratio of the perimeter to the physiological length		$P_L = \frac{P}{L}$
Perimeter ratio of Major axis length and Minor axis length (P_{LW})	Ratio of the leaf perimeter over the sum of the physiological length and the physiological width		$P_{LW} = \frac{P}{L+W}$
Number of convex points	Number of points to create a convex hull		
Perimeter convexity (P_C)	Ratio of the convex perimeter to the perimeter of the leaf		$P_C = \frac{P_{CH}}{P}$
Area convexity (A_{C1})	Normalized difference of the convex hull area and the leaf's area		$A_{C1} = \frac{(CH-A)}{A}$
Area ratio of convexity (A_{C2})	Ratio between leaf's area and area of the leaf's convex hull		$A_{C2} = \frac{A}{CH}$

Continued on next page

Table 3 – continued from previous page

Shape feature	Description	Pictogram	Formula
Equivalent diameter (D_E)	Diameter of a circle with the the same area as the leaf's area		$D_E = \sqrt{\frac{4 * A}{\pi}}$

5.12 Texture Features

Texture is the term used to describe the surface of a given object or appearance and is undoubtedly a main feature used in computer vision and pattern recognition (Waldchen and Mader 2018). Texture can only be assessed for a group of pixels whereas colour is usually a property of a pixel. Generally, texture is associated with the feel of various materials to human touch and texture image analysis is based on visual interpretation (Waldchen and Mader 2018) of this feeling. Leaf surface is a natural texture which has random persistent patterns and do not show detectable quasi-periodic structure (Waldchen and Mader 2018). Therefore, several authors claim fractal theory to be better suited than statistical, spectral, and structural approaches for describing these natural textures (Waldchen and Mader 2018).

The Haralick texture features (Boland 2000; **???**) are functions of the normalized GLCM (Gray Level co-occurrence Matrix - see matrix **??**) which is a common method to represent image texture.

$$GLCM = \begin{bmatrix} p(1,1) & p(1,2) & \cdot & \cdot & \cdot & p(1,N_g) \\ p(2,1) & p(2,2) & \cdot & \cdot & \cdot & p(2,N_g) \\ \cdot & \cdot & \cdot & & & \cdot \\ \cdot & \cdot & & \cdot & & \cdot \\ \cdot & \cdot & & & \cdot & \cdot \\ p(N_g,1) & p(N_g,2) & \cdot & \cdot & \cdot & p(N_g,N_g) \end{bmatrix}$$

The GLCM (**??**) is square with dimension N_g, where N_g is the number of gray levels in the image (Boland 2000). Element [i,j] of the matrix is generated by counting the number of times a pixel with value i is adjacent to a pixel with value j and then dividing the entire matrix by the total number of such comparisons made (Boland 2000). Therefore each entry is considered to be the probability (see figure 25) that a pixel with value i will be found adjacent to a pixel of value j (Boland 2000). Four such matrices can be calculated, because adjacency can be defined to occur in each of four directions in a 2D (see figure 26), square pixel image (horizontal, vertical, left and right diagonals - see equation 24) (Boland 2000).

The Haralick statistics are calculated for co-occurrence matrices generated using each of these directions (see figure 25) of adjacency (Boland 2000). Haralick then described 14 statistics that can be calculated from the co-occurrence matrix with the intent of describing the texture of the image. Through the research, we only used the following 4 statistics among 14 of them, because most of the researchers used these 4 statistics as texture features (see figure 4) of leaf images.

where p(i, j) = Probability density function of gray - level pairs

$$\mu_x = \sum_i i \sum_j p(i,j)$$

$$\mu_y = \sum_j j \sum_i p(i,j)$$

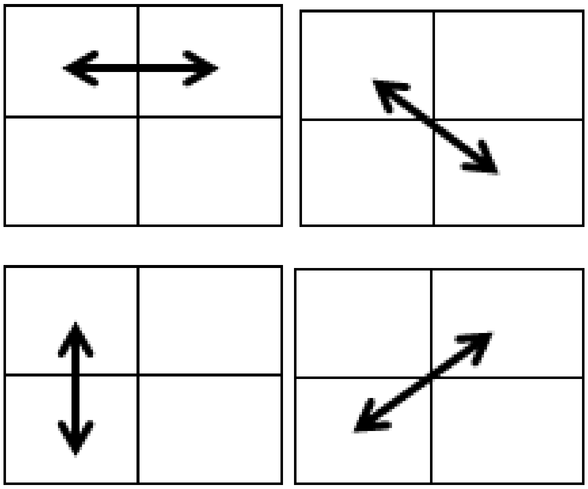


Figure 24: Four directions of adjacency as defined for calculation of the Haralick texture features

Texture feature	Description	Formula
Contrast	Measures the scale of difference in the Gray level Co-occurrence matrices.	$\sum_{n=0}^{N_g-1} n^2 \left\{ \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p(i,j) \right\}, \lvert i-j \rvert = n$
Correlation	Measures of how correlated a pixel is to its neighbour over the whole image in the Gay level Co-occurrences matrices.	$\frac{\sum_i \sum_j (ij)p(i,j)-\mu_x\mu_y}{\sigma_x\sigma_y}$
Entropy	Measures the randomness of the elements of the co-occurrence matrix.	$-\sum_i \sum_j p(i,j)log(p(i,j))$
Inverse Difference Moment	Measure of homogeneity. Measures the closeness of the distribution of elements in the GLCM to the GLCM diagonal.	$\sum_{i=1}^{N_g} \sum_{j=1}^{N_g} \frac{1}{1+(i-j)^2}p(i,j)$

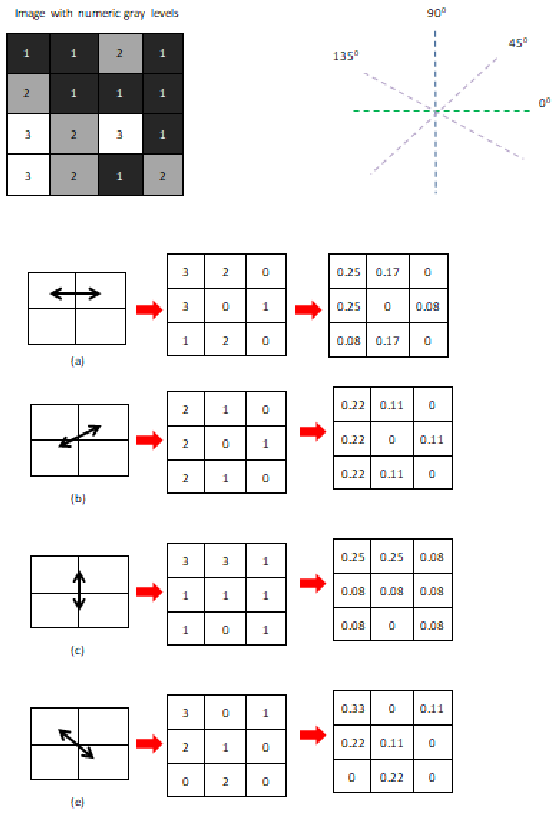
Table 4: Definitions of texture features

$$\sigma_x = \sum_i (i-\mu_x)^2 \sum_j p(i,j)$$

$$\sigma_y = \sum_j (j-\mu_y)^2 \sum_i p(i,j)$$

5.13 Color Features

Colour is an important feature of images (Waldchen and Mader 2018; Caglayan, Guclu, and Can 2013). Colour properties are defined within a particular colour space like red-green-blue (RGB) (Kodituwakku and S.Selvarajah 2010; Waldchen and Mader 2018). Colour properties can be extracted from images after a colour space is specified. In the field of image recognition, a number of general colour descriptors have been



1	1	2	1
2	1	1	1
3	2	3	1
3	2	1	2

(a)

3	2	0
3	0	1
1	2	0

0.25	0.17	0
0.25	0	0.08
0.08	0.17	0

(b)

2	1	0
2	0	1
2	1	0

0.22	0.11	0
0.22	0	0.11
0.22	0.11	0

(c)

3	3	1
1	1	1
1	0	1

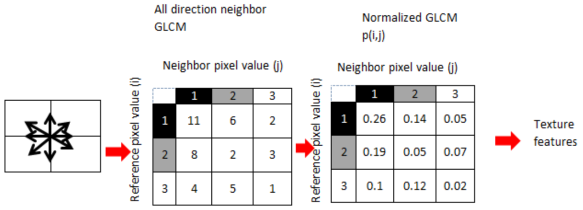
0.25	0.25	0.08
0.08	0.08	0.08
0.08	0	0.08

(e)

3	0	1
2	1	0
0	2	0

0.33	0	0.11
0.22	0.11	0
0	0.22	0

Figure 25: Computing the Haralick texture features from a 4 × 4 example image step by step



All direction neighbor GLCM

	1	2	3
1	11	6	2
2	8	2	3
3	4	5	1

Normalized GLCM p(i,j)

	1	2	3
1	0.26	0.14	0.05
2	0.19	0.05	0.07
3	0.1	0.12	0.02

Figure 26: Computing the Haralick texture features from a 4 × 4 example image with all direction

introduced. Colour moments (Kodituwakku and S.Selvarajah 2010; Waldchen and Mader 2018) are the simple descriptor among them. Mean, standard deviation skewness and kurtosis are the comment moments. Colour moments are used for characterizing planar colour patterns, irrespective of viewpoint or illumination conditions and without the need for object contour detection (Waldchen and Mader 2018). Colour moments are convenient for real-time applications because of its low dimension and low computational complexity.

Some of leaf images have very similar shape like Hathawariya (27) and Iramusu (27) in our experiment. Even though shapes are similar in some leaves, there are some differences in colours of leaf images. Therefore in addition to the shape features, we extracted colour based features of leaf images as well.

We used mean (μ) and standard deviation (σ) of intensity values of red, green and blue channels (Caglayan, Guclu, and Can 2013). Mean and standard deviation of each component are calculated as follows:

$$\mu = \frac{1}{MN}\sum_{x=1}^{M}\sum_{y=1}^{N} p_{xy} \tag{4}$$

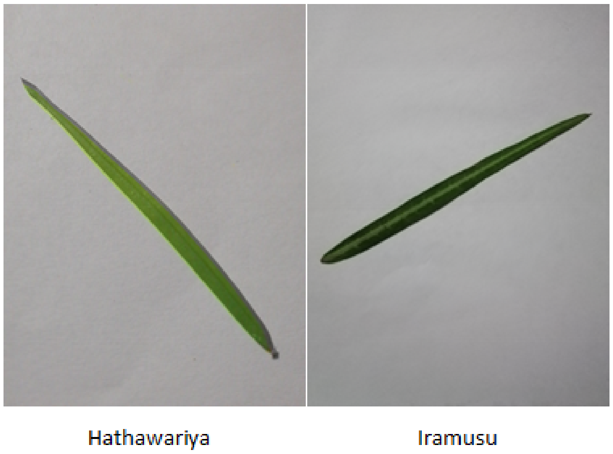


Figure 27:

$$\sigma = \frac{1}{MN}\sqrt{\sum_{x=1}^{M}\sum_{y=1}^{N}(p_{xy} - \mu)^2} \tag{5}$$

where M and N are dimensions of a leaf image and p_{xy} is the intensity value of pixel at (x, y) coordinate.

5.14 Scagnostic features

Scatterplot diagnostics is a term in Tukey neologism for scagnostics (Wilkinson, Anand, and Grossman 2005; Wilkinson and Wills 2008; Dang and Wilkinson 2014; Dang, Anand, and Wilkinson 2013). Scagnostics are characterizations of the 2D distributions of orthogonal pairwise projections of a set of points in multidimensional Euclidean space (Wilkinson, Anand, and Grossman 2005; Wilkinson and Wills 2008). Measures like density, skewness, shape, outliers, and texture are included in these characterizations.

There are two people who popular in the discussion about scagnostics. John and Paul Tukey, and Wilkinson et al. (Wilkinson, Anand, and Grossman 2005; Wilkinson and Wills 2008) introduce their ideas about scagnostics in two angles. In first place mid of 1980s, an exploratory graphical method called scagnostics was introduced by John and Paul Tukey. A set of measures characterizing a 2D scatterplot (Dang and Wilkinson 2014) was the base of this method. But John and Paul Tukey were never published their ideas.

After some years later, the details collected from the first author's recollection of Institute for Mathematics and its Applications (IMA) and some discussions with Paul Tukey, Wilkinson et al. (2005) developed nine scagnostics measures defined on planar proximity graphs (Wilkinson, Anand, and Grossman 2005; Wilkinson and Wills 2008). These measures were scalable to large datasets and therefore suitable for practical applications (Wilkinson, Anand, and Grossman 2005; Wilkinson and Wills 2008).

Wilkinson et al. was introduced nine scagnostics measures which determined the cells of scatter plot matrix (SPLOM) (Dang and Wilkinson 2014). SPLOM means that the organization of scatterplots in the layout of a covariance matrix. In here the scatterplots are of the scagnostics measures.

By characterizing a large collection of 2D scatterplots through a small number of measures like the area of the peeled convex hull (Tukey 1974) (Wilkinson, Anand, and Grossman 2005; Wilkinson and Wills 2008), the perimeter length of this hull, the area of closed 2D kernel density isolevel contours (Silverman 1986) (Wilkinson, Anand, and Grossman 2005; Wilkinson and Wills 2008), the perimeter length of these contours, and a nonlinearity measure of association based on principal curves (Hastie and Stuetzle 1989) (Wilkinson, Anand, and Grossman 2005; Wilkinson and Wills 2008) of the arrangement of points in these plots was proposed by Tukeys. This was a simple and powerful idea, but when implementing many details wanted to involve.

Wilkinson et al. proposed his method by including the criteria that should met by candidate scagnostics.

1. Distinguish many types of point distributions: multivariate normal, lognormal, multinomial, sparse, dense, convex, clustered, etc.

2. A small number of scagnostics characterizing these distributions.
3. Should have a common scale because want to compare them with each other.
4. Should have a comparable distribution because want to compare them to standard.
5. The intrinsic dimensionality of these scagnostics, when calculated over a large number of heterogeneous scatterplots, to be as large as possible.
6. To be efficiently computable because the scagnostics should be scalable to large numbers of points and dimensions

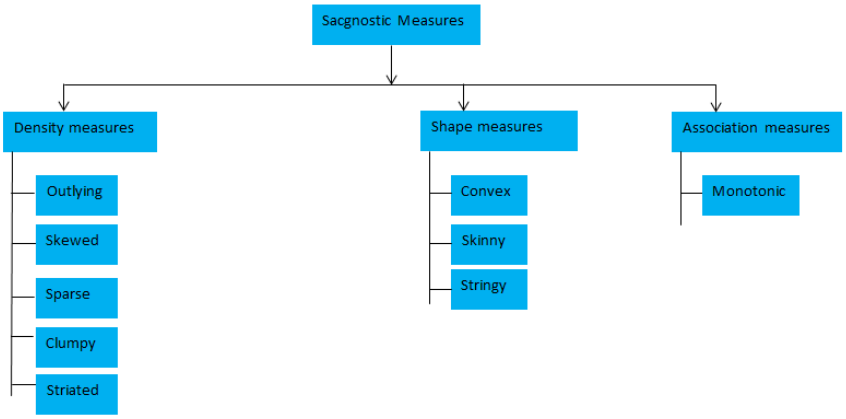


Figure 28: Hierarchy of Scagnostics

Scagnostic measures are based on following definitions.

5.14.1 Geometric Graphs

- Graph

A graph G = (V, E) is defined as a set V (called vertices) together with a relation on V induced by a set E (called edges). A pair of vertices is defined as an edge $e(\nu, \omega)$, with e $\in$ E and $\nu, \omega \in$ V.

- Geometric Graph

An embedding of a graph in a metric space S that maps vertices to points and edges to straight line segments connecting pairs of points is defined as a geometric graph G* = [f(V), g(E), S].

From several features of 2D Euclidean geometric graphs, Scagnostic measures are derived.

The Euclidean distance between vertices that connected to edge is defined as the length of an edge, length(e).

The sum of the lengths of edges in graph is known as the length of a graph, length(G).

A list of successively adjacent, distinct edges are known as a path. If first and last vertex are the same of the path, then the path is closed.

A region bounded by a closed path is known as a polygon (P). A polygon bounded by exactly one closed path that has no intersecting edges is known as a simple polygon.

The length of boundary of a simple polygon is known as the perimeter of a simple polygon. The area of interior of a simple polygon is known as the area of a simple polygon.

5.14.2 Minimum Spanning Tree (MST)

- Tree

A graph in which any two nodes are connected by exactly one path is known as a *tree*.

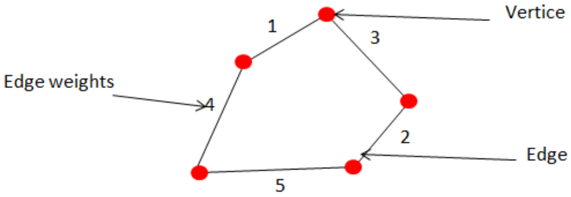


Figure 29: Graph with 5 vertices and 5 edges

- Spanning Tree

An undirected graph whose edges are structured as a tree is defines as a *Spanning Tree*.

Spanning Tree of Graph G is: G'(V',E')

$$V' = V$$

$$E' \subset E$$

$$E' = |V| - 1$$

The graph can have more than one spanning tree.

Spanning tree should not be disconnected and not contain any cycle. By removing one edge from the Spanning tree will make it disconnected. By adding one edge to the Spanning tree will create a loop. A complete (Each vertices connected with each other) undirected graph can have n^{n-2} number of spanning trees where n is the number of vertices. Every connected and undirected graph has at least one Spanning Tree. Disconnected graph doesn't have any spanning tree. From a complete graph by removing $max(edges - n + 1)$ edges we can construct a spanning tree.

- Minimum Spanning Tree

A spanning tree whose total length is least of all spanning trees on a given set of points is known as a Minimum Spanning Tree (MST).

If each edge has distinct weights then there will be only one and unique MST.

- Remark

The geometric MST computed from Euclidean distances between points in a 2D Euclidean geometric graph is the restriction.

5.14.3 Convex Hull

A collection of the boundaries of one or more simple polygons that have a subset of the points for their vertices and that collectively contain all the points, is defined as a hull of a set of points embedded in 2D Euclidean space.

If a hull contains all the straight line segments connecting any pair of points in its interior, is known as a convex hull. The convex hull bounds a single polygon. After deleting the points on the convex hull, a convex hull called peeled convex hull is computed.

5.14.4 Alpha Hull

Most of proximity graphs (neighborhood graph) represent the nonconvex shape of a set of points on the plane. A geometric graph whose edges are determined by an indicator function based on distances between a given set of points in a metric space, is known as a proximity graph. An open disk D is used to define the indicator function.

If a point is on the boundary of D then D *touches* a point and if a point is in D then D *contains* a point. An open disk of radius r is defined as $D(r)$.

An alpha shape (Dang and Wilkinson 2014) is a collection of one or more simple polygons (Wilkinson and Wills 2008; Wilkinson, Anand, and Grossman 2005). An edge exists between any pair of points that can be touched by an open disk $D(\alpha)$ containing no points, is defined as an alpha shape graph.

A value of α to be the average value of the edge lengths in the MST (Wilkinson and Wills 2008; Wilkinson, Anand, and Grossman 2005). The large values like 90th percentile of the MST edge lengths are used, because to reduce noise. If the percentile exceeds a tenth, clamp the value at one-tenth the width of a frame, because it prevents in including sparse or striated point sets in a single alpha graph.

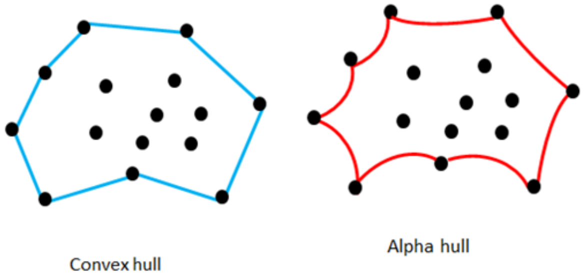


Figure 30: Convex hull and alpha hull

5.14.5 Preprocessing

To improve the performance of the algorithm and robustness of the measures, preprocessing techniques as binning and deleting outliers are used before computing geometric graphs.

- Binning

As the first step of binning, the data are normalized to the unit interval. Then use a 40 by 40 hexagonal grid to aggregate the points in each scatterplot. Reduce the bin size by half and rebin until no more than 250 non emty cells, if there are more than 250 non empty cells. By efficiency (too many bins slow down calculations of the geometric graphs) and sensitivity (too few bins obscure features in the scatterplots), the choice of bin size is constrained.

To improve the performance, hexagon binning is used. To manage the problem of having to many points that start to overlap, hexagon binning is used. The plots of hexagonal binning are density rather than points. To use hexagons instead of squares for binning a 2D surface as a plane, there are many reasons. Hexagons are more similar to circle than square.

To keep scagnostics orientation-independent this bias reduction is important. To attenuate the influence of binning, stabilizing transformation is used when computing scagnostics from binned data.

The weight function is defined as;

$$\omega = 0.7 + \frac{0.3}{1 + t^2} \tag{6}$$

where $t = \frac{n}{500}$. (n is the number of vertex)

If $n > 2000$ then this function is fairly constant. By using hex binning the shape and the parameters of the function is determined. In computing Sparse, Skewed and Convex scagnostics this weight function is used to adjust for bias.

- Deleting Outliers

To improve robustness of the scagnostics, deleting outliers can be used. A vertex whose adjacent edges in the MST all have a weight (length) greater than ω is defined as an outlier in this context. By considering nonparametric criterion for the simplicity and Tukey's idea choose the following weight calculation.

$$\omega = q_{75} + 1.5(q_{75} - q_{25}) \tag{7}$$

where q_{75} is the 75th percentile of the MST edge lengths and $(q_{75} - q_{25})$ is the Interquartile range of the edge lengths.

5.14.6 Degree of a Vertex

The degree of a vertex in an undirected graph is know as the number of edges associated with the vertex.

Eg:- Vertices of degree 2 There are 2 edges associated with each vertex.

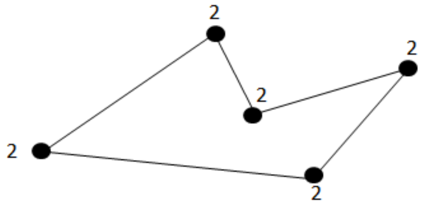


Figure 31: Vertices of degree 2

Geometric Graphs	Notation
Convex Hull	H
Alpha Hull	A
Minimum Spanning Tree	T

Table 5: Notations of Geometric Graphs

5.14.7 Density Measures

Detect different distributions of scattered points in density measures.

- Outlying

$$C_{outlying} = \frac{length(T_{outliers})}{length(T)} \tag{8}$$

The outlying measure calculate before deleting the outliers for the other measures. The proportion of the total edge length of the minimum spanning tree accounted for by the total length of edges adjacent to outlying points is used to calculate the outlying measure.

- Skewed

$$q_{skew} = \frac{q_{90} - q_{50}}{q_{90} - q_{10}} \tag{9}$$

$$C_{skew} = 1 - \omega(1 - q_{skew}) \tag{10}$$

where ω is the weight function (6).

The skewed measure is the first measure of relative density which is a relatively robust measure of skewness in the distribution of edge lengths. After adaptive binning skewed tends to decrease with n.

- Sparse

$$C_{sparse} = \omega q_{90} \tag{11}$$

where ω is the weight function (6) and q_{90} is the 90th percentile of the distribution of edge lengths in the MST.

The second relative density measure is Sparse measure that measures whether points in a 2D scatterplot are confined to a lattice or a small number of locations on the plane.

If the number of points is extremely small or tuples are produced by the product of categorical variables, then sparse can be happen.

$$q_{90} = \alpha statistic \tag{12}$$

The α statistic exceeds unity (e.g., when all points fall on either of the two diagonally opposing vertices of a square), clamp the value to 1 in the extremely rare event (Wilkinson and Wills 2008, @inproceedings44).

- Clumpy

$$C_{clumpy} = max_j[1 - \frac{max_k[length(e_k)]}{length(e_j)}] \tag{13}$$

Clustering points are not indicated by an extreme distribution of MST edge lengths. Therefore RUNT statistic (Wilkinson and Wills 2008, @inproceedings44) which is another measure based on the MST, is introduced. The smaller of the number of leaves of each of the two subtrees joined at that node isdefined as the runt size of a dendogram node. There is an association between runt size (r_j) each edge (e_j) in the MST because there is an isomorphism between a single-linkage dendrogram and the MST.

The smaller of the two subsets of edges that are still connected to each of the two vertices in e_j after deleting edges in the MST with lengths less than length(e_j), is known as the RUNT graph (R_j) (Wilkinson and Wills 2008, @inproceedings44).

The RUNT-based measure responds to clusters with small maximum intracluster distance relative to the length of their nearest-neighbor inter-cluster distance (Wilkinson and Wills 2008, @inproceedings44). In the formula j runs over all edges in MST and k runs over all edges in RUNT graph.

- Striated

$$C_{striated} = \frac{1}{|V|} \sum_{\nu \in V^{(2)}} I(\cos \theta_{e(\nu,a)e(\nu,b)} < -0.75) \tag{14}$$

where $V^{(2)} \subseteq V$ and $I()$ be an indicator function.

Striated define the coherence in a set of points as the presence of relatively smooth paths in the minimum spanning tree.

The measure is based on the number of adjacent edges whose cosine is less than minus 0.75.

5.14.8 Shape Measures

Both topological and geometric aspects of shape of a set of scattered points is considered. As an example, a set of scattered points on the plane appeared to be connected, convex and so forth, want to know under the shape measures. By definition scattered points are not like this. Therefore to make inferences additional machinery (based on geometric graphs) is needed. By measuring the aspects of the convex hull, the alpha hull, and the minimum spanning tree is determined.

- Convex

$$C_{convex} = \omega[area(A)/area(H)] \tag{15}$$

where ω is the weight function (6).

The ratio of the area of the alpha hall(A) and the area of the convex hull(H) is the base of measuring convexity.

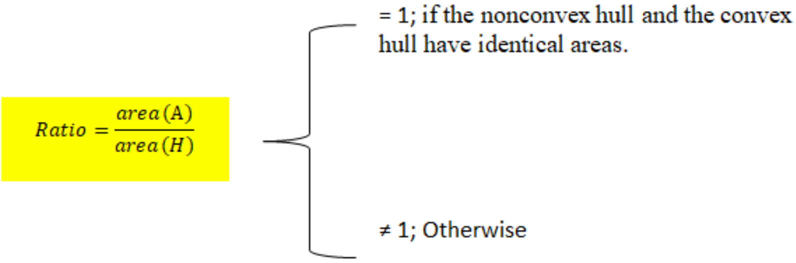

$Ratio = \frac{area(A)}{area(H)}$
= 1; if the nonconvex hull and the convex hull have identical areas.
≠ 1; Otherwise

Figure 32:

- Skinny

$$C_{skinny} = 1 - \frac{\sqrt{4\pi area(A)}}{perimeter(A)} \tag{16}$$

Roughly, the skinny is measured by using the corrected and normalized ratio of perimeter to area of a polygon measures.

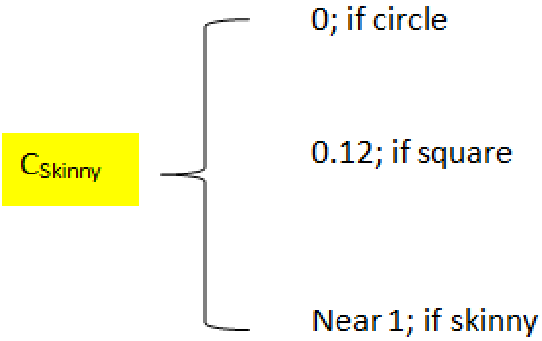


Figure 33:

- Stringy

$$C_{stringy} = \frac{|V^{(2)}|}{|V| - |V^{(1)}|} \tag{17}$$

where V is the number of vertices.

A skinny shape with no branches is known as a stringy shape. By counting the vertices of degree 2 in the minimum spanning tree and comparing them to the overall number of vertices minus the number of single-degree vertices, skinny measure is calculated.

To adjust for negative skew in its conditional distribution of n, cube the stringy measure.

5.14.9 Association Measure

Symmetric and relatively robust measure of association are interested.

- Monotonic

$$C_{monotonic} = r^2_{Spearman} \tag{18}$$

To assess the monotonicity in a scatter plot, the squared spearman correlation coefficient is used. This is the only coefficient not based on a subset of the Delaunay graph (Wilkinson and Wills 2008).

In calculating monotonicity, squared the coefficient because to consider the large values and to remove the distinction between positive and negative coefficients (Because assume that the investigators are more interested in strong relationships rather than negative or positive).

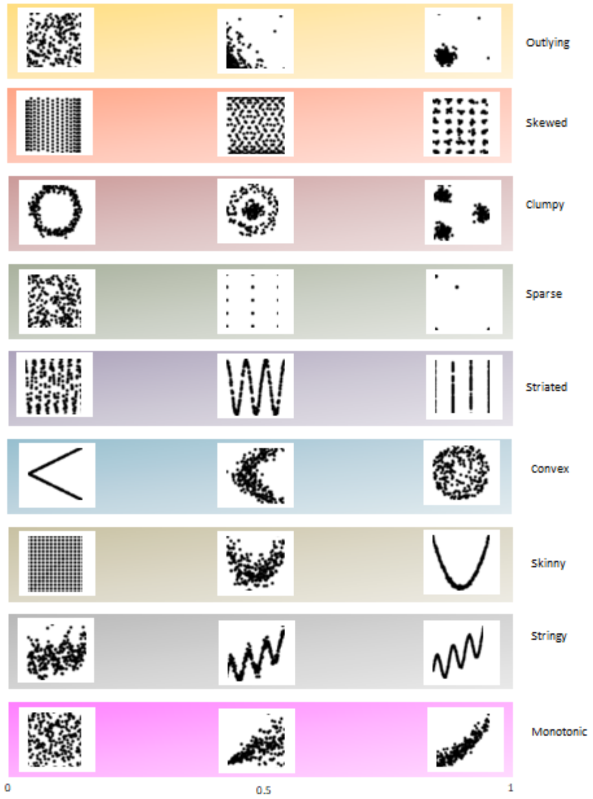


Figure 34: Exploring scatter plots by their scagnostics

We measured the scagnostic features for Cartesian and Polar coordinates separately.

6 Example

There are two color image leaf datasets (Flavia ,Swedish) are used. After passing through the required image processing steps, binary image is extracted which has a white foreground and black background.

The leaf images are taken as the closest ones. Therefore to find the best contour among several contours, can use the contour which contains the center of leaf image. Identify the best contour is really important when extracting shape features.

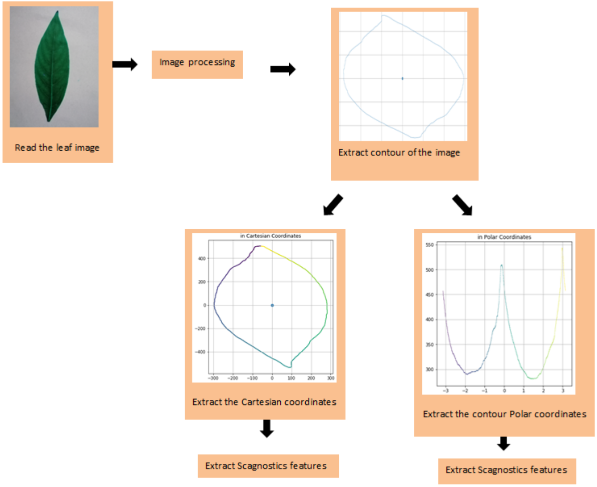

Read the leaf image
Image processing
Extract contour of the image
in Cartesian Coordinates
400
200
0
−200
−400
−300
−200
−100
0
100
200
300
Extract the Cartesian coordinates
in Polar Coordinates
550
500
450
400
350
300
−3
−2
−1
0
1
2
3
Extract the contour Polar coordinates
Extract Scagnostics features
Extract Scagnostics features

Figure 35: Preprocessing for Scagnostics

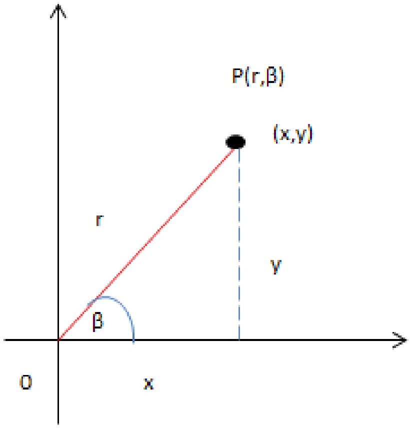


Figure 36: Polar coordinate

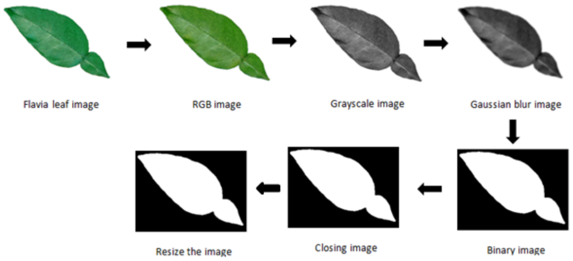


Figure 37: Image processing steps of Flavia dataset

7 Software

8 Visualization of Leaf Images in the Feature Space

8.1 Swedish Dataset

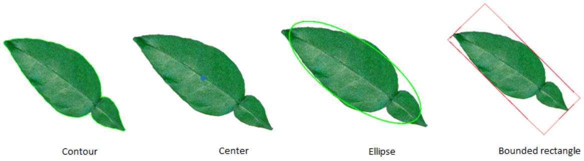
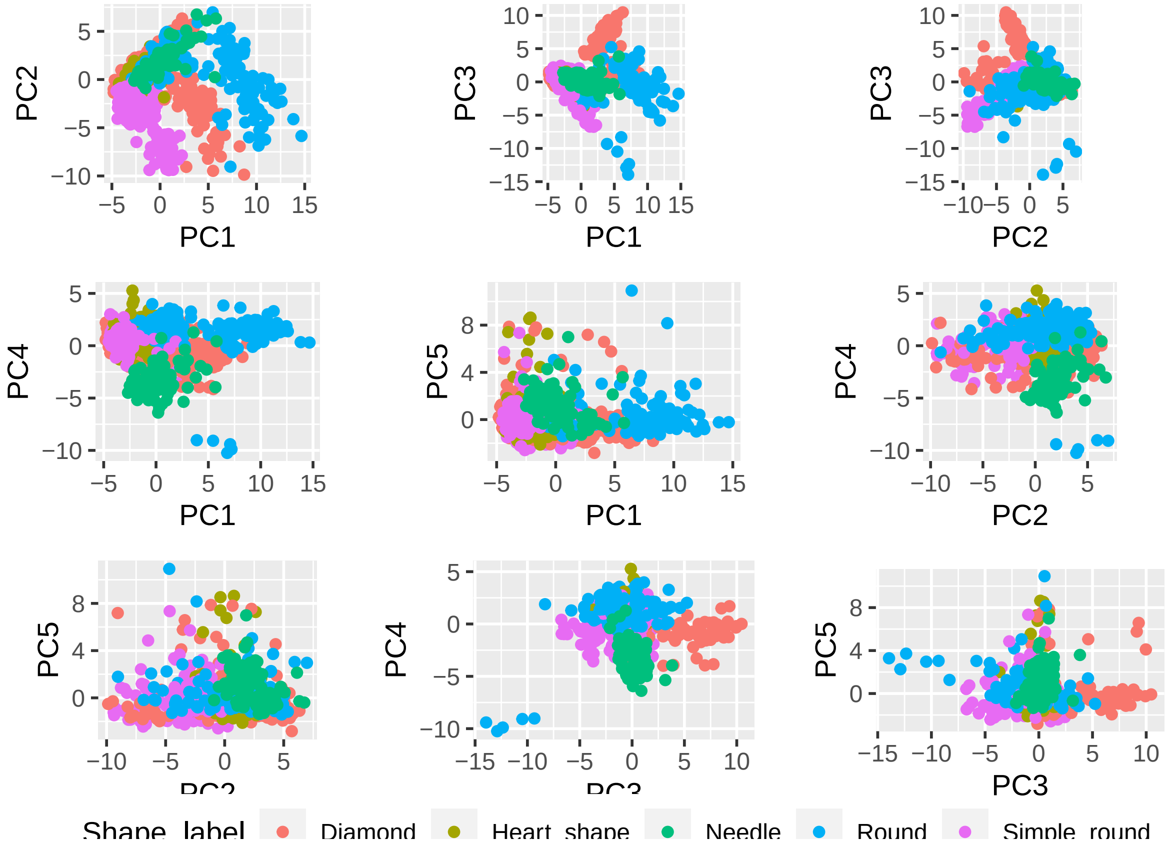


Figure 38: Example feature extraction of Flavia dataset

Feature name	Software	Package

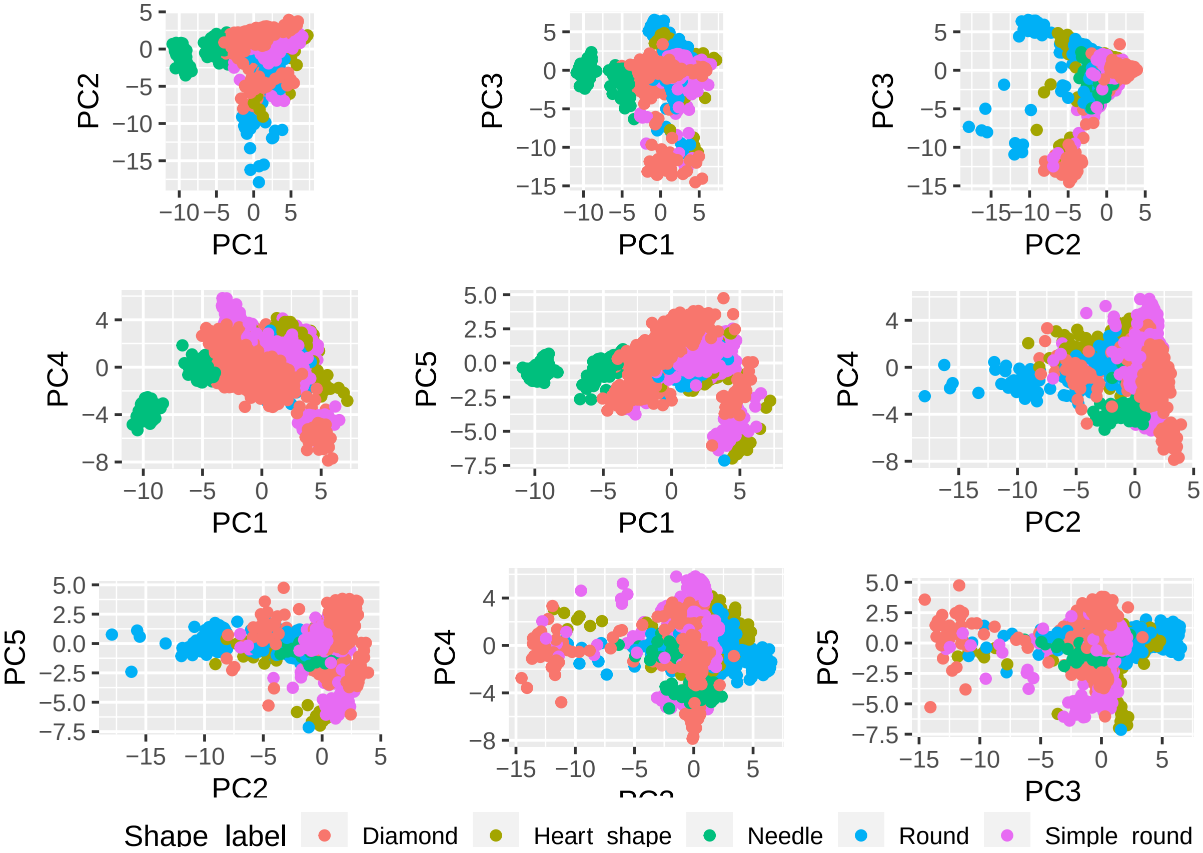
Table 6: Software in feature extraction

8.1.1 PCA projection



8.2 Flavia Dataset

8.2.1 PCA projection



9 Discussion and Conclusions

Reference

Aakif, Aimen, and Muhammad Khan. 2015. "Automatic Classification of Plants Based on Their Leaves." *BIOSYSTEMS ENGINEERING* 139 (November): 66–75. https://doi.org/10.1016/j.biosystemseng.2015.08.003.

Anantrasirichai, Nantheera, Sion L. Hannuna, and Nishan Canagarajah. 2017. "Automatic Leaf Extraction from Outdoor Images." *CORR* abs/1709.06437. http://arxiv.org/abs/1709.06437.

Azlah, Muhammad, Lee Suan Chua, Fakhrul Rahmad, Farah Abdullah, and Sharifah Alwi. 2019. "Review on Techniques for Plant Leaf Classification and Recognition." *COMPUTERS* 8 (October): 77. https://doi.org/10.3390/computers8040077.

Boland, MV. 2000. "Quantitative Description and Automated Classification of Cellular Protein Localization Patterns in Fluorescence Microscope Images of Mammalian Cells."

Caglayan, Ali, Oguzhan Guclu, and Ahmet Can. 2013. "A Plant Recognition Approach Using Shape and Color Features in Leaf Images." In, 161–70. https://doi.org/10.1007/978-3-642-41184-7_17.

Dang, T. N., A. Anand, and L. Wilkinson. 2013. "TimeSeer: Scagnostics for High-Dimensional Time Series." *IEEE Transactions on Visualization and Computer Graphics* 19 (3): 470–83. https://doi.org/10.1109/TVCG.2012.128.

Dang, Tommy, and Leland Wilkinson. 2014. "ScagExplorer: Exploring Scatterplots by Their Scagnostics." In *IEEE Pacific Visualization Symposium*, 73–80. https://doi.org/10.1109/PacificVis.2014.42.

Gonzalez, Rafael C., and Richard E. Woods. 2006. *Digital Image Processing (3rd Edition)*. USA: Prentice-Hall, Inc.

Goyal, N., Kapil, and N. Kumar. 2018. "Plant Species Identification Using Leaf Image Retrieval: A Study." In *2018 International Conference on Computing, Power and Communication Technologies (Gucon)*, 405–11.

Herdiyeni, Yeni, and Ni Wahyuni. 2012. "Mobile Application for Indonesian Medicinal Plants Identification Using Fuzzy Local Binary Pattern and Fuzzy Color Histogram." In *2012 INTERNATIONAL CONFERENCE ON ADVANCED COMPUTER SCIENCE AND INFORMATION SYSTEMS, ICACSIS 2012 - PROCEEDINGS*.

Jeon, Wang-Su, and Sang-Yong Rhee. 2017. "Plant Leaf Recognition Using a Convolution Neural Network." *THE INTERNATIONAL JOURNAL OF FUZZY LOGIC AND INTELLIGENT SYSTEMS* 17 (March): 26–34. https://doi.org/10.5391/IJFIS.2017.17.1.26.

Kalyoncu, Cem, and Onsen Toygar. 2014. "Geometric Leaf Classification." *COMPUTER VISION AND IMAGE UNDERSTANDING* 133 (November). https://doi.org/10.1016/j.cviu.2014.11.001.

Kodituwakku, Saluka, and S.Selvarajah. 2010. "Comparison of Color Features for Image Retrieval." *Indian Journal of Computer Science and Engineering* 1 (October).

Waldchen, Jana, and Patrick Mader. 2018. "Plant Species Identification Using Computer Vision Techniques: A Systematic Literature Review." *ARCHIVES OF COMPUTATIONAL METHODS IN ENGINEERING* 25 (April): 507–43. https://doi.org/10.1007/s11831-016-9206-z.

Wilkinson, Leland, A. Anand, and Robert Grossman. 2005. "Graph-Theoretic Scagnostics." In *INFOVIS*, 5:157–64. https://doi.org/10.1109/INFVIS.2005.1532142.

Wilkinson, Leland, and Graham Wills. 2008. "Scagnostics Distribution." *Journal of Computational and Graphical Statistics - J COMPUT GRAPH STAT* 17 (June): 473–91. https://doi.org/10.1198/106186008X320465.

Wu, S. G., F. S. Bao, E. Y. Xu, Y. Wang, Y. Chang, and Q. Xiang. 2007. "A Leaf Recognition Algorithm for Plant Classification Using Probabilistic Neural Network." In *2007 Ieee International Symposium on Signal Processing and Information Technology*, 11–16.