

University kasdi merbah ouargla
Faculty of New Information Technologies and Communication
Department of informatique and Information Technologies



**Memory
academique master**

Sector: Computer science

Field: New Information Technologies and Communication

Specialty: Industry informatics

presented by: MIDOUN ASMA

DJENDOUCI NESRINE

Theme:

**classification of plant leaves using morphological features and Hu
moments**

publicly graduate in:

.././.....

Name of juries:

FARAHDEBBAGHsupervisor

M.BEKKARI FOUAD.....president of jury

M.KORICHI MERIEM.....examining board

OUSSAMA AIDI, KHELDI BILEL..... co-supervisor

College year:2015-2016

Acknowledgment

First and foremost, we have to thank our parent's for their love and support throughout our lives. Thank you both because you have supported us always. Our sisters, our brothers, and every person help us in this work.

We would like to express our gratitude to our co-supervisors, OUSSAMA AIDI, and KHELDI BILÉL for their support, patience, and encouragement throughout our graduate studies. It is not often that one finds co-supervisors that always find the time for listening to the little problems and roadblocks that unavoidably crop up in the course of performing research. Their technical was essential to the completion of this memory and has taught us lot of lessons about academic research in general.

I would like to sincerely thank my supervisor FARAHDEBBAGH, and the president of jury Mr. BEKKARI FOUAD and examining board KORICHI MERIEM for accepting judging us.

*Also our gratitude to all those teachers of the department of Computer Science, University KASDI MARBAH OUARGLA.
Also my colleagues promoting 2015-2016 master computer science*

Dedicate

For our father and our mother

Our brothers and sisters

Friends very close friends: AMEL

All our friends.

ABDALLAH and LYESS

All our family

All...

Abstract: Plants are an important in our ecosystems. Their identification and classification has always been a matter of interest for the botanists as well as for humans. With huge number of plant species, only a tiny part of the plants is known. The leaves of the plant are so important because they carry a lot of information about the plant species. The aim of this work is to automatically classify leaves into different classes through their images. The advancement in image processing and machine learning has made this a quick and easy process. Specifically, we describe leaf images using morphological features together with Hu moments. Then, we use the KNN and Naive Bayes classifiers to first learn the visual characteristics of different leaf classes and then to decide for test leaves the classes they belong to. In order to truly investigate the performance of our system, we carried out many experiments on the well-known Flavia dataset. Results reported at the end of this work show promising results.

Key words:

classification, morphological, shape, Hu moments, K-nn.

Resumé: Les plantes sont des éléments importants dans nos écosystèmes. Leur identification et leur classification a toujours été une question d'intérêt pour les botanistes, ainsi que pour les humains. Avec grand nombre d'espèces de plantes, peu genre des plantes sont connue. Les feuilles des plantes sont importantes, car ils portent beaucoup d'informations sur les espèces végétales.

Le but de ce travail est de classer automatiquement les feuilles on assotions chaque feuille à une certaine classe.

L'avancement dans le traitement d'image et la machine d'apprentissage d'image a donner une rapidité a ce processus.

Plus précisément, nous décrivons les images des feuilles à l'aide des caractéristiques morphologiques et des moments Hu. Ensuite, nous utilisons le classificateur KNN et classificateur Naïve Bayes.

D'abord nous calculons les caractéristiques morphologique et moment Hu des différentes classes des feuilles, puis nous laisserons notre système de tester auxquelles classent ils appartiennent. Afin d'étudier vraiment la performance de notre système, nous avons réalisé de nombreuses expériences sur l'ensemble de données Flavia bien connu. Les résultats présentés à la fin de ce travail.

Mots clés :

classification, morphologique, forme, Hu moments, K-nn

Summary

General introduction	1
----------------------------	---

Chapter I: Images features

I.1. Introduction	3
I.2. Image descriptors	3
I.2.1. color descriptor.....	3
a) Dominant Color Descriptor-DCD.....	4
b) Color Layout Descriptor- CLD.	4
c) Group-of-pictures-GoP.....	4
d) Edge Histogram descriptor	4
e) Color histogram	5
I.2.2. texture descriptors	7
a) Frequency	7
b) Statistical	9
c) Geometrical	10
I.2.3. Shape descriptors.....	11
3.1. Classification of shape representation	12
3.1.1. region-based.....	12
A-Geometrical features.....	13
B-Hu-moment	15
C-Zernike moments	16
D- Legendre moments	17
3.1.2.Contour based	17
A-Fourier Descriptor	17
I.3.Conclusions	18

Chapter II: Classification methods

II.1.Introduction:.....	19
II.2.Difference between supervised and unsupervised algorithms:	19
II.3. Algorithms	19
II.3.1.K-nearest-neighbor (kNN)	19
II.3.2.NaiveBayes	22
II.3.3.SVM.....	24
II.3.4.LDA	25
II.4.Conclusion	28

Chapter3: Results and discussion:

III.1. Introduction	29
III.2. Experiment Condition.....	29
III.2.1- hard ware	29
III.2.2-software	29
III.3.System presentation	30
III.4.Experiments and results	35
III.5. Result Analysis	36
III.6. Conclusion	37
General Conclusion	38
Reference	39

Figures list

Figure I.1: CLD method	4
Figure I.2: GOP method	4
Figure I. 4: color histogram for a Tigre image	6
Figure I. 5: color histogram for a group of flower image	6
Figure I. 6: color histogram for green leaf image	6
Figure I.7: color histogram for green leaf image	6
Figure I.8: set of different kind of texture	7
Figure I.9: image responses for Gabor filter kernels	8
Figure I.10: real image and Fourier transform image	9
Figure I.11: Description of texture with local binary patterns.....	10
Figure I.12: Classification of shape representation and description techniques	12
Figure I.13: Relationship between physiological length and physiological width	13
Figure II.1: the different value of neighbor affects the classification.....	20
Figure II.2: the nearest-neighbor classification method.	21
Figure II.3: different possibility of a hyperplane and best hyperplane position	24
Figure II.4: difference between class separation in low dimension and high dimension	25
Figure II.5: maximizing the component axes for class separation.....	26
FigureIII.1: Matlab environment	30
Figure III.2: interface of our application	31
Figure III.3: classification application window.	31
Figure III. 4: loading image	32

Figure III.5 : result of calculate morphological and geometrical features	33
Figure III. 6 : the result of classification.....	33
Figure III.7 : leaves images similarity	34
Figure III. 8 : About application.....	34
Figure III.9: a graph of naive Bayes accuracy with invariants features and morphological features and both of them.	36
Figure III.10: a graph of K-nn accuracy with invariants moment and morphological features and both of them.	36

Table list

Table 1 : results tabulated for flavia dataset tested on our classifiers.....	35
---	----

General introduction

Thousands species of plants exist on earth and they share a very close relationship to human beings. This relation is a long and continuous. We need plants for basic human purposes, and many things in life depend on them. Also world health organization estimates that 80% of people in Asia and Africa rely on herbal medicines due to the fact that they are safe to human health and affordable.

Plants play a very important role in our life because without plants, there could be no life on Earth. They are the primary producers that sustain all other life forms. This is so because plants are the only organisms that can make their own food. Although we now live in a highly industrialized society, we have not lost this dependence on plants. We need to be aware of the part that plants play in our lives and must ensure that we care for these plants to continue this long relationship. It is vital that we remind ourselves of our reliance on green plants and our duty to protect the plant species.

In the nature there are numerous kinds of plants: helpful, damaging, eatable, poisoning, most of them are anonymous. For the domain specialists, manually identifying the type of a given plant is known to be tedious, slow and time-consuming. In addition, and because of the existing of a large variety of plants, it is very difficult to for even a specialist to identify the kind of some plants.

Technologies are now everywhere and its development is increasing particularly in multimedia field, many information has been digitized. Digital information has become a heart of many activity zones like: industry, medical, juridical, geographic, etc. This development due to low costs of equipment witch facilitated the exchange of digitized media information. Images are one of the digitized information, digitizing more and more digital images have been generated. As a result there is an urgent need to create a tool that effectively classifies plant images as fast as possible the fastest the easiest way. The aim is to define, for a given plant, the kind it belongs to, based on the plant leaf.

Computer vision is defined as the science which is concerned with artificial system that extracts information from images. It consists in three parts: features extraction, pattern classification based on these features, and pattern recognition. Machine learning is a sub-field of computer vision, which aims to use the knowledge acquired after a training stage in order to give judgments for unseen cases.

In this work, we develop a system that classifies leaves into different classes based on machine learning techniques. Our system is composed of two main modules: image features and classification algorithms. Image features are used to describe the leaves in a manner that permit to distinguish between two different classes. Whereas, classification algorithms are used to learn the visual characteristics of each class and then to classify new leaf images without any human assistant.

In chapter one, we will introduce a set of images features that can be extracted from an image with different approaches. In chapter two we will review some classification methods with their advantages and limits.

In the last chapter we have mentioned all the steps of our application realization beginning with introducing our work environment ending with experiments and analyses.

CHAPITRE I

Image features

I.1. Introduction:

Image features are special characteristics in image content; in the first place they were widely used in image retrieval, then the same features also used in image classification.

As we know, visual features of the images provide a description of their content. Image classification has been a topic of intensive research in recent years. It is the process of classifying images based on automatically extracted features.

What is an image?

In information technology, an image is a binary representation of visual information such as drawings, graphs, logos, etc. It can be saved electronically on any storage device. Digital images are made of primitive units called pixels. Typically, pixels are organized in an ordered array of two or three dimensions.

The number of columns and rows of an array represent, respectively, the width and the height of the corresponding image. Each pixel defines a coordinate (x, y). An image could be, either, described with tags that are provided by an expert or with meaningful information extracted from the content of the image.

This information is called visual features, which are used to classify or retrieve images in computer vision algorithms. Image features could be broadly, categorized into three main categories which are color, texture and shape features.

I.2. Image descriptors:

I.2.1. color descriptor:

In general the color is one of the most distinguishing visual features and it is rich with information. There are several color features descriptors such as:

- a) **Dominant Color Descriptor-DCD:** describes the representative color distributions in an image or a region of interest through an effective, compact and intuitive format. It contains two main components:

- 1- Representative colors.
 - 2- Their percentages in the region or image.
- b) **Color Layout Descriptor- CLD**: represents the spatial distribution of colors in an image.

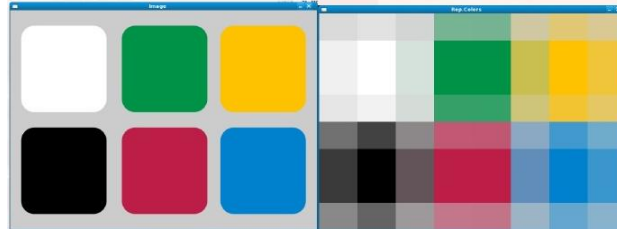


Figure I.1: CLD method

- c) **Group-of-pictures-GoP**: are individual frames of pictures that are grouped together and played back so that the viewer registers the video's spatial motion.

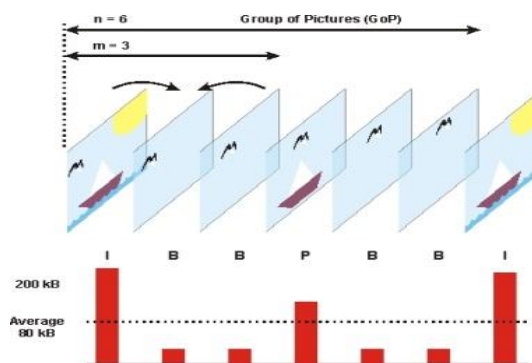


Figure I.2: GOP method

- d) **Edge Histogram descriptor**: describes edge distribution with a histogram based on local edge distribution in an image are amongst the most commonly used features.

- e) **Color histogram**:

Stability, simplicity on calculation, beside it is insensitive to rotation, translation, noise and scale changes make the color histogram a discriminate-full feature. Color histogram represents quantifiably the distribution of colors in the image. There are mainly three kinds of color histograms. Jain proposed three separated one-dimensional histogram to reduce the number of dimensions but he did not take in account the correlation between pixels. So Swain, who proposed the three dimensional histogram to cover the correlation, but the problem was its static table which make it suitable only for trademark image database. Finally, MEHTRE and others refer to table-based color histogram .The representation of color histogram of an image through a vector.

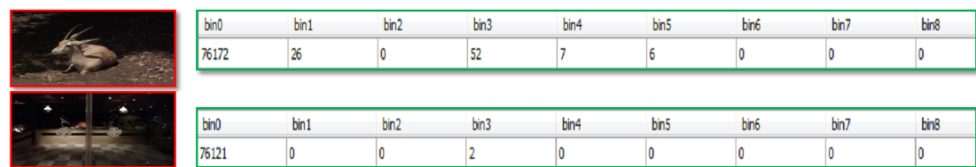


Figure I.3: the representation of an image through a vector

This distribution on the vector table through this histogram equation:

$$9R+3G+B.$$

Although color histogram is an effective feature; it does not take into consideration the spatial distribution of colors in images, which is a serious weakness. In our work when we used the color histogram, the difference between two different images with same or similar color the difference between them is inconsiderable. The figure bellow shows that color histogram in images that have different colors is considerable rather than a different images with same or similar color.

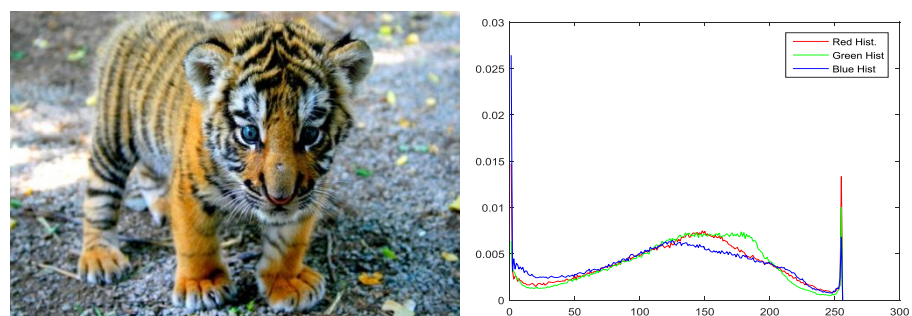


Figure I. 4: color histogram for a Tigre image

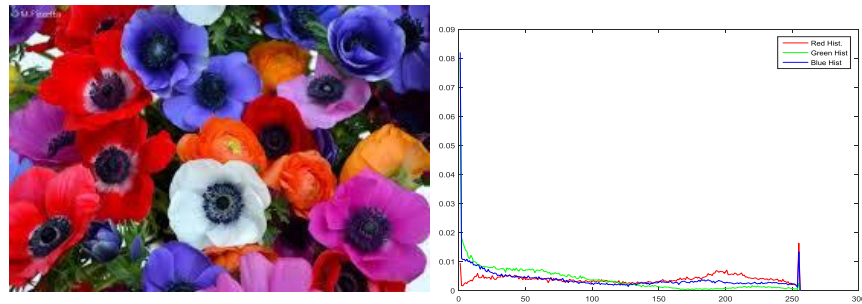


Figure I. 5: color histogram for a group of flower image

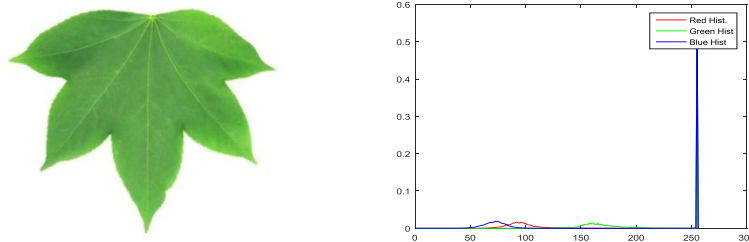


Figure I. 6: color histogram for green leaf image

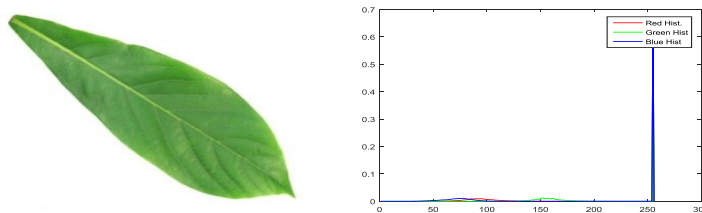


Figure I.7: color histogram for green leaf image

I.2.2. texture descriptors:

There is no formula definition of texture. Many researchers have been trying to define textures from a certain perspective. For example, Haralick considers a texture as an "organized area phenomenon" which can be decomposed into "primitives" having specific spatial distributions [RMH,79]. Generally, we can say

that texture refers to the visual patterns that have properties of homogeneity and do not result from the presence of only a single color or intensity.

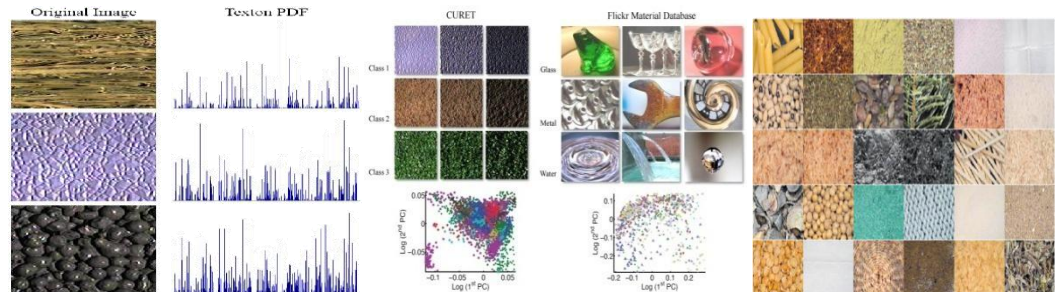


Figure I.8: set of different kind of texture

Several approaches and models have been proposed to describe texture. We mention the three most known approaches that are statistically, Frequency and Geometrically Approach.

a) Frequency:

Frequency-based methods define the image as a set of signals with amplitudes and directions. These methods consist in extracting the energy carried by signals in different frequency bands. One of many frequency based method is: Gabor filters .

An image is filtered with a set of Gabor filters with different preferred orientations and spatial frequencies, which cover appropriately the spatial frequency domain, and the features which are obtained from a feature vector field which is used further. Gabor feature vectors can be used directly as input to a classification [TMR]

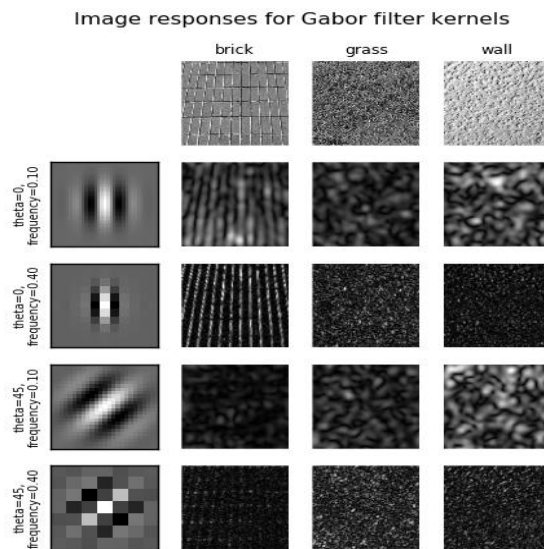


Figure I.9: image responses for Gabor filter kernels

Other widely used method is: Fourier theory states that any signal, in our case visual images, can be expressed as a sum of a series of sinusoids. In the case of imagery, these are sinusoidal variations in brightness across the image.

*The Fourier Transform is used in a wide range of applications, such as image analysis, image filtering, image reconstruction and image compression.

The image below shows a sinusoidal brightness image, and its two-dimensional Fourier transform, presented here also as a brightness image.

Every pixel of the Fourier image is a spatial frequency value, the magnitude of that value is encoded by the brightness of the pixel. In this case there is a bright pixel at the very center - this is the DC term, flanked by two bright pixels either side of the center, that encode the sinusoidal pattern.

The brighter the peaks in the Fourier image, the higher the contrast in the brightness image.

Since there is only one Fourier component in this simple image, all other values in the Fourier image are zero, depicted as black.



Figure I.10: real image and Fourier transform image

b) Statistical:

Statistical methods consist in calculating the appearance frequency different gray-levels in the image. Then they derivate some statistical moments (such as correlation, energy, etc.) from these appearance frequency histograms.

*Gray level co-occurrence matrix (GLCM): Is a statistical method of examining texture that considers the spatial relationship of pixels, also known as the gray-level spatial dependence matrix.

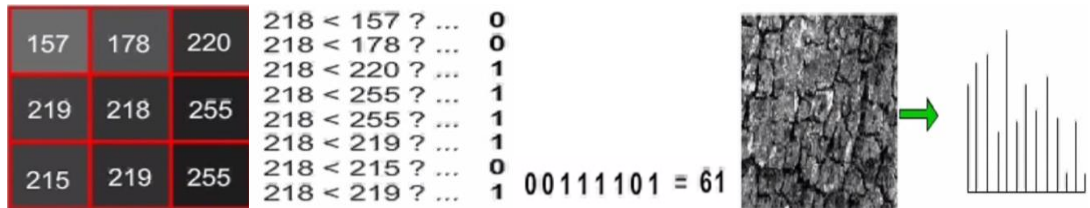
The GLCM functions characterize the texture of an image by calculating how often pairs of pixel with specific values and in a specified spatial relationship occur in an image, creating a GLCM, and then extracting statistical measures from this matrix.

*Local Binary Pattern (LBP): is a simple yet very efficient texture operator which labels the pixels of an image by thresholding the neighborhood of each pixel and considers the result as a binary number.

It has a discriminative power and computational simplicity which makes it possible to analyze images in challenging real-time settings. It can be seen as a unifying approach to the traditionally divergent statistical model of texture analysis. Perhaps the most important property of the LBP operator in real-world applications is its robustness to monotonic gray-scale changes caused, for example, by illumination variations.



1-LBP: texture represented 2-Convert image to Grey level 3-find for each image its LBP



4-LBP for pixel [1, 1]

5-Compare grey level value of pixel [1, 1] with values of its neighbors

6-LBP for pixel [1, 1]

7-Storing count of each LBP for this texture

Figure I.11: Description of texture with local binary patterns.

c) Geometrical:

Geometrical-based methods used for structural analyze of a texture in order to identify of the motif (primitive) that constitute a texture and the rule it follows in its distribution.

Illustrate example of textures that have a primitives and a distribution rules.

One of the famous geometrical based methods is edge detection ,Edge detection is a fundamental tool in feature extraction, It is the name for a set of mathematical methods which aim at identifying points in a digital image at which the image brightness changes sharply or, more formally, has discontinuities. The points at which image brightness changes sharply are typically organized into a set of curved line segments termed edges.

I.2.3. Shape descriptors:

Shape is an important visual feature and it is one of the basic features used to describe image content [Den, Guo, 02]. If we are talking about shape of objects such as a football or an egg, we are implicitly referring to the boundary of the object [CST]. So it refers to the general appearance of an object's outline. Other aspect of shape is the region occupied by the object in image. The information shape needed to be known from an image is done by extracting its shape features.

Extracting a shape feature in accordance with human perception is not an easy task. Due to the fact that human vision and perception are an extraordinary complicated system, it will be best if machine vision has superexcellent performance with small complexity.

so why do we need shape feature extraction?

Shape feature extraction and representation plays an important role in the following categories of applications [MIY, KIK, JOR, 08]:

- * Shape retrieval: searching for all shapes in a typically large database of shapes that are similar to a query shape. Usually all shapes within a given distance from the query are determined or the first few shapes that have the smallest distance.

- * Shape recognition and classification: determining whether a given shape matches a model sufficiently or which of representative class is the most similar.

- * Shape alignment and registration: transforming or translating one shape so that it best matches another shape, in whole or in part.

- * Shape approximation and simplification: constructing a shape of fewer elements (points, segments, triangles, etc.), that is still similar to the original. And in our work because leaf recognition based on color even we have taken it in consideration although it has the same green or similar one, it will have a declining performance due to leaf color change in seasons or color change due to environmental factors.

*recognition based on texture because of its descriptors it has many inconveniences such as in:GLCM:

- 1- They require a lot of computation (many matrices to be computed).
2. Features are not invariant to rotation or scale changes in the texture and LBP which is based on the assumption that the local differences of the central pixel and its neighbors are independent of the central pixel itself, & this independence is not warranted in practice. [Vis, Jay, 13]

We present in the following the method used for recognizing a given shape in an image.

3.1. Classification of shape representation:

Shape representation can be generally classified into two classes of methods: contour-based methods and region-based methods. The classification is based on whether shape features are extracted from the contour only or are extracted from the whole shape region. The figure below shows Classification of shape representation and description techniques.

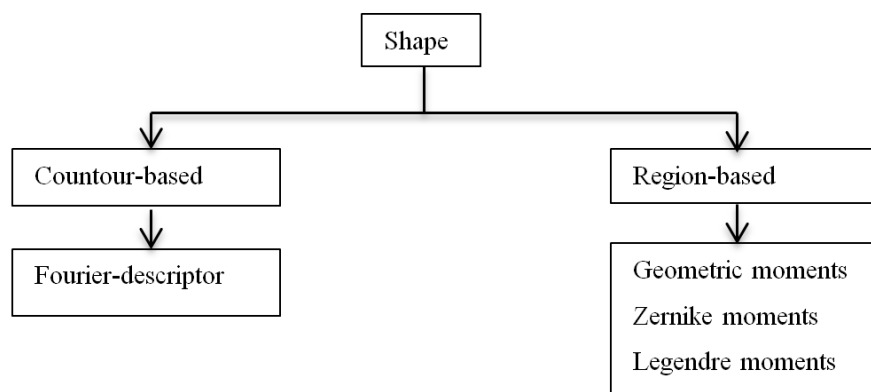


Figure I.12: Classification of shape representation and description techniques

3.1.1. Region based:

The commonly basic extracted features are five [KSI,IGU,SGU, 10], and there are 10 features derived from the previous one.

A) Geometrical Features

a) *Diameter*: The diameter is defined as the longest distance between any two points on the margin of the leaf. It is denoted as D.

b) *Physiological Length*: The distance between the two terminals of the main veins of the leaf is defined as the physiological length. It is denoted as L_p .

c) *Physiological Width*: The maximum length of a line, which is orthogonal to the main vein, is defined as the physiological width. It is denoted as W_p . The relationship between Physiological Length and Physiological Width is shown in Figure 13.

d) *Leaf Area*: Denoted as A, It is the number of pixels forming the leaf. It can be estimated by calculating the number of 1's in the binary image.

e) *Leaf Perimeter*: Denoted as P, leaf perimeter is calculated by counting the number of pixels consisting leaf margin.

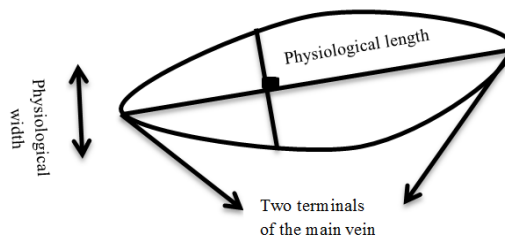


Figure I.13: Relationship between physiological length and physiological width

f) *Aspect Ratio*: Aspect ratio is defined as the ratio of physiological length L_p to physiological width W_p . It is also called as Eccentricity or Slimness.

$$\text{Aspect Ratio} = L_p / W_p$$

g) Form Factor: This feature is used to describe the difference between a leaf and a circle. It is the ratio of the area of the leaf A and the square of the perimeter of the leaf margin P . It is also called as Roundness, Compactness or Circularity.

$$\text{Form Factor} = A/P^2$$

h) Smooth Factor: The effect of noise to image area is used to illustrate the smoothness of the leaf image.

Smooth Factor: Area of the image obtained by 5x5 rectangular averaging filter/Area of the image obtained by 2x2 rectangular averaging filters

i) Narrow Factor: Narrow factor is the ratio of the diameter of the leaf D and physiological length L_p .

$$\text{Narrow Factor} = D/L_p$$

j) Rectangularity: Rectangularity is the similarity between a leaf and a rectangle. It is the ratio of the product of physiological length L_p and physiological width W_p and the leaf area A .

$$\text{Rectangularity} = (L_p * W_p)/A$$

k) Perimeter Ratio of Diameter: It is the ratio of perimeter to the diameter denoting the ratio of leaf perimeter P and leaf diameter D .

$$\text{Perimeter Ratio of Diameter} = P/D$$

l) Perimeter Ratio of Physiological length & width: This feature is the ratio of the leaf perimeter P and the sum of physiological length L_p and the physiological width W_p .

$$\text{Perimeter Ratio of Physiological length width} = P/(L_p + W_p)$$

m) Solidity: solidity is the ration of leaf area A and the area of convex hull A_c .

$$\text{solidity} = A/A_c$$

n) Convexity: convexity is the ratio of area of the convex perimeter P_c and the perimeter of the leaf P .

$$\text{convexity} = P_c / P$$

o) Irregularity:

Irregularity is the ratio of the radius of max circle R_{\max} enclosing the region and the radius of min circle R_{\min} that can be contained in the region. This feature is important when the shape of the leaf is irregular.

$$\text{Irregularity: } R_{\max} / R_{\min}$$

B) Hu moments:

Invariant moments (IM): are also called geometric moment invariants.

Geometric moments, are the simplest of the moment functions

[MIY,KIK,JOR,08]. Geometric moment function

m_{pq} of order $(p + q)$ is given as:

$$m_{pq} = \sum_x \sum_y x^p y^q f(x, y) \quad p, q = 0, 1, 2 \dots$$

this allows computing the center of mass of the image as following:

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad p, q = 0, 1, 2 \dots$$

where $\bar{x} = m_{10} / m_{00}$ and $\bar{y} = m_{01} / m_{00}$

A set of 7 invariant moments (IM) are given by [MKH,62]:

$$\varphi_1 = \eta_{20} + \eta_{02}$$

$$\varphi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$

$$\varphi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$$

$$\varphi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$

$$\varphi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12}) [(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + 3(\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$

$$\varphi_6 = (\eta_{20} - \eta_{02}) [(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}^2 (\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$$

$$\varphi_7 = 3(\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12}) [(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{12} - \eta_{03})(\eta_{21} + \eta_{03}) [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$

where $\eta_{pq} = \mu_{pq} / \mu_{00}^\gamma$, with $\gamma = \frac{p+q}{2} + 1$, $\forall p+q \geq 2$

IM are computationally simple. Moreover, they are invariant to rotation, scaling and translation. However, they have several inconvenient [MEC,YAA,05], we mention:

- * noise sensitivity: higher-order moments are very sensitive to noise.
- * Information redundancy.
- * Large variation in the dynamic range of values: since the basis involves powers of p and q , the moments computed have large variation in the dynamic range of values for different orders. This may cause numerical instability when the image size is large.

C) Zernike moments:

Moment descriptors have been studied for image recognition and computer vision since the 1960s [TCC].Teague first introduced the use of Zernike moments to overcome the shortcomings of information redundancy present in the popular geometric moments [LJS;MRT].Zernike moments are a class of orthogonal moments and have been shown effective in terms of image representation. Zernike moments are rotation invariant and can be easily constructed to an arbitrary order. Although higher order moments carry more fine details of an image, they are also more susceptible to noise.

The Zernike polynomials are a set of complex, orthogonal polynomials defined over the interior of a unit circle $x^2 + y^2 = 1$ [KHA,HYH]

$$V_{nm}(x, y) = V_{nm}(\rho, \theta) = R_{nm}(\rho) e^{jm\theta} \quad (1)$$

$$R_{nm}(\rho) = \sum_0^2 (-1)^s \frac{(n-s)!}{s! \left(\frac{n+|m|}{2} - s\right)! \left(\frac{n-|m|}{2} - s\right)!} \rho^{n-2s} \quad (2)$$

where n is a non-negative integer, m is an integer such that $n-|m|$ is even

and $|m| \leq n$, $\rho = \sqrt{x^2 + y^2}$, and $\theta = \tan^{-1} \frac{y}{x}$.

Projecting the image function onto the basis set, the Zernike moment of order n with repetition m is:

$$A_{nm} = \frac{n+1}{\pi} \sum_x \sum_y f(x, y) V_{nm}(x, y), \quad x^2 + y^2 \leq 1 \quad (3)$$

D) Legendre moments:

Legendre moments are moments with Legendre polynomials as kernel function, were first introduced by Teague [TMR]. Legendre moments belong to the class of orthogonal moments, and they were used in several pattern recognition applications [CWC,PRA,RMU,04]. They can be used to minimize the maximum of redundancy measure in a set of moment functions [CHR,88]. By convention, the translation and scale invariant functions of Legendre moments are achieved by using a combination of the corresponding invariants of geometric moments. The two dimensional Legendre moments of order $(p + q)$, with image intensity function $f(x, y)$, are defined as [RMU,KRR,98]:

$$L_{pq} = \frac{(2p+1)(2q+1)}{4} \iint_{-1}^1 P_p(x)P_q(y)f(x,y)dxdy, x,y \in [-1,1] \quad (4)$$

where Legendre polynomial, $P_p(x)$, of order p is given by:

$$P_p(x) = \frac{1}{2^p} \sum_{k=0}^{n/2} (-1)^k \frac{(2p-2k)!}{k!(p-k)!(p-2k)!} x^{p-2k}, x \in [-1,1] \quad (5)$$

For a digital image of size $N \times N$, Eq. (1) can be approximated by:

$$L_{pq} = \frac{(2p+1)(2q+1)}{(N-1)^2} \sum_{i=1}^N \sum_{j=1}^N P_p(x_i)P_q(y_j)f(x_i,y_j) \quad (6)$$

with $x_i = (2i-N-1)/(N-1)$, $y_j = (2j-N-1)/(N-1)$

The Legendre polynomial obeys the following recursive relation

$$P_{p+1}(x) = \frac{2p+1}{p+1} x P_p(x) - \frac{p}{p+1} P_{p-1}(x), p \geq 1 \quad (7)$$

with $p_0(x) = 1, P_1(x) = x$

Zernike moments and Legendre moment are classified under orthogonal moments which are defined in an orthogonal base, which avoids the redundancy of information carried by each of the moments.

3.1.2. Contour based**a) Fourier Descriptor:**

Fourier Descriptors FDs are the most popular shape recognition and image processing [Bed]. They are widely used due to their simplicity and high performance in recognition field and classification [D. ZH, G. LU, 05]. In addition, they allow us to describe shape object with different details levels. Fourier descriptors are calculated from object contour. They represent object contour in a signal of 1 dimension, then decompose it in Fourier parts.

I.3. Conclusions:

we have detail in this chapter key point of image classification based shape features, through extraction and several approaches (Hu-moment...). So we have indicated that classification begins with features extracts. We have mentioned also that they are several features (color, texture, shape). The right choice of one of them depends on image kind which is in our work plant leaf.

CHAPTRE II

Classification methods

II.1. Introduction:

The intent of classification process is to categorize an image into one of several classes. It is used two main classification methods are: Supervised and Unsupervised learning, this table below.

II.2. Difference between supervised and unsupervised algorithms:

supervised	Unsupervised
<ul style="list-style-type: none"> *Training data includes both the input and desired results. *For some examples the correct results (targets) are known and are given in input to the model during the learning process. *the construction of a proper training, validation and test set (Bok) is crucial. *These methods are usually fast and accurate. *Have to be able to generalize: give the correct results when new data are given in input without knowing a priori target. 	<ul style="list-style-type: none"> *The model is not provided with the correct results during the training. *Can be used to cluster the input data in classes on the basis of their statistical properties only. *Cluster significance and labeling. *the labeling can be carried out even if the labels are only available for a small number of objects representative of the desired classes.

II.3. Algorithms:

we will review in the following 4 common algorithms:

II.3.1. K-nearest-neighbor (kNN):

3.1.1. algorithm description:

KNN is supervised algorithm, and one of the top 10 data mining algorithms; one of the simplest classifiers [XIW,VIK,07] which memorizes the entire data and performs classification only if the attributes of the test object matches one of the training examples exactly.

The idea behind the k-Nearest Neighbor algorithm is to build a classification method using no assumptions about the form of the function, $y = f(x_1, x_2, \dots, x_p)$ that relates the dependent (or response) variable y , to the independent (or predictor) variables x_1, x_2, \dots, x_p . It is a non-parametric because it doesn't make any assumption about data distribution (little or no prior knowledge about the distribution of data) and lazy learning algorithm because it doesn't use the training data points to any generalization.

There are three key elements of this approach:

1- a set of labeled objects, e.g. a set of named classes

2-a distance between objects. It can be measured with different distance functions:

$$\text{Euclidean } \sqrt{\sum_{i=1}^k (x_i - y_i)^2} \quad (1)$$

$$\text{Manhattan } \sum_{i=1}^k |x_i - y_i| \quad (2)$$

$$\text{Minkowski } (\sum_{i=1}^k (|x_i - y_i|)^q)^{1/q} \quad (3)$$

it should be noted that all three distance measures are only valid for continuous variables. But rather that in the instance of categorical variables the hamming distance must be used.

$$D_h = \sum_{i=1}^k |x_i - y_i| \quad (4)$$

$$x = y \rightarrow D=0 \quad (5)$$

$$x \neq y \rightarrow D=1 \quad (6)$$

3-The value of k , the number of nearest neighbors.

Choice of k effects on the result of classification the figure bellow shows it.

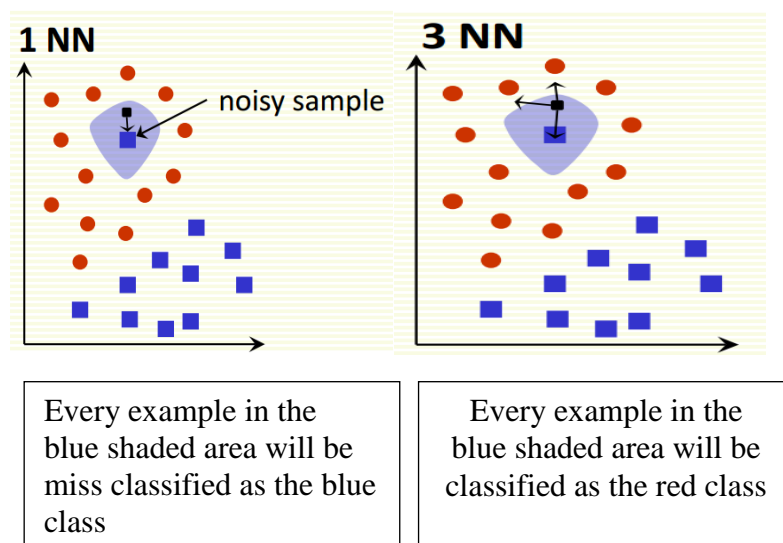


Figure II.1: the different value of neighbor affects the classification

So the choice of k is critical phase and a smaller k leads to high variance which mean less stable, and larger k leads to higher bias (less precise).

And proper choice of k depends on the data:

- Adaptive methods, heuristics
- Cross-validation

To classify an unlabeled object, the distance between this object and the labeled objects is calculated, its k -nearest neighbors are identified, and the class labels of these nearest neighbors are then used to determine the class label of the object.

Figure II.2: shows the nearest-neighbor classification method.

Given a training set D and a test object $z = (\mathbf{x}', y')$, the algorithm computes the distance (or similarity) between z and all the training objects $(\mathbf{x}, y) \in D$ to determine its nearest-neighbor list, D_z . (\mathbf{x} is the data of a training object, while y is its class.

Likewise, \mathbf{x}' is the data of the test object and y' is its class.)

Once the nearest-neighbor list is obtained, the test object is classified based on the voting of majority class of its nearest neighbors:

$$\text{Majority Voting: } y' = \operatorname{argmax}_{(x_i, y_i) \in D_z} I(v = y_i) \quad (7)$$

Where v is a class label, y_i is the class label for the i^{th} nearest neighbors, and $I(\cdot)$ is an Indicator function that returns the value 1 if its argument is true and 0 otherwise.

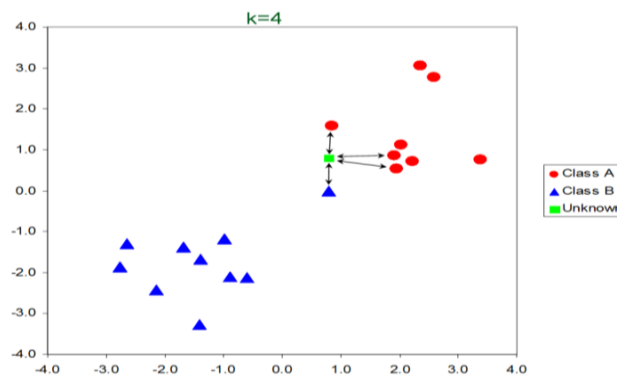


Figure II.2: the nearest-neighbor classification method.

3.1.2. Algorithm**Input:**

D , the set of k training objects, and test object $z = (x', y')$

Process:

Compute $d(x', x)$, the distance between z and every object, $(x, y) \in D$.

Select D_z the set of k closest training objects to z .

Output:

$$y' = \operatorname{argmax}_{(x_i, y_i) \in D_z} i(v = y_i)$$

3.1.3. Applications of K-NN:

- 1- Nearest neighbor based content retrieval.
- 2- Gene expression.
- 3- Protein- interaction and 3D structure prediction.

3.1.4. Drawbacks of K-NN:

- 1- Lack of generalization due to minimal or absence of training phase.
- 2- Keeps all the training data is needed during the test phase which take a long time if the set of training data is big.
- 3- Complexity in searching the nearest neighbor for each simple.
- 4- Selecting scaling features is common problem for all methods but affects k-nn even more.
- 5- Scaling of different dimensions.
- 6- The choice of k , weather it is too small, then the result can be sensitive to noise points, or is too large, then the neighborhood may include too many points from other classes.

II.3.2. Naive Bayes:**3.2.1. Algorithm description:**

One of several supervised learning algorithms naïve Bayes. Naive Bayes is a simple probabilistic classifier based on applying Bayes theorem (Bayes' rules) with strong independence (naive) assumptions between the features.

3.2.2. Explanation of Bayes' rule:

$$\text{Bayes' Rule: } p(C_k|x) = \frac{p(C_k) \times p(x|C_k)}{p(x)} \quad (8)$$

$p(C_k|x)$: probability of instance x being in class C_k .

$p(x|C_k)$: probability of generating instance x given class C_k .

$p(C_k)$: Probability of occurrence of class C_k .

$p(x)$: Probability of instance x occurring.

The basic idea of Bayes' rule is that the outcome of an event (C_k) can be predicted based on some evidences (x) that can be observed. From Bayes' rule, we have:

(1) A *priori* probability of C_k or $P(C_k)$: This is the probability of an event *before* the evidence is observed.

(2) A *posterior* probability of C_k or $P(C_k|x)$: This is the probability of an event *after* the evidence is observed.

It can be written as:

$$\text{posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}}$$

3.2.3. Gaussian naive Bayes:

When dealing with continuous data, a typical assumption is that the continuous values associated with each class are distributed according to a Gaussian distribution.

The probability distribution of v given a class c : $p(x = v|c)$ is computed as following:

$$p(x = v|c) = \frac{1}{\sqrt{2\pi\sigma_c^2}} e^{-\frac{(v-\mu_c)^2}{2\sigma_c^2}} \quad (9)$$

where μ_c is the mean of the values in \mathcal{X} associated with class c , and σ_c^2 is the variance of the values in x associated with class c .

a) Advantages:

- it only requires a small amount of training data to estimate the parameters necessary for classification.
- Fast to train and fast to classify
- Not sensitive to irrelevant features
- Handles real and discrete data

– Handles streaming data well

b) Disadvantages:

– Assumes independence of features

II.3.3.SVM:

3.3.1. Algorithm description:

SVMs are a set of supervised learning methods used solving binary classification problems, regression and outliers detection. This algorithm is related to statistical learning theory [VVA, 00], it was first introduced in 1992 [BEB,92], SVM becomes popular because of its success in handwritten digit recognition [MAL, 11].

Its aim is to find an optimal separating hyper plane (OSH) between the two data sets. SVM finds the OSH by maximizing the margin between the classes.

The figure bellow shows that:

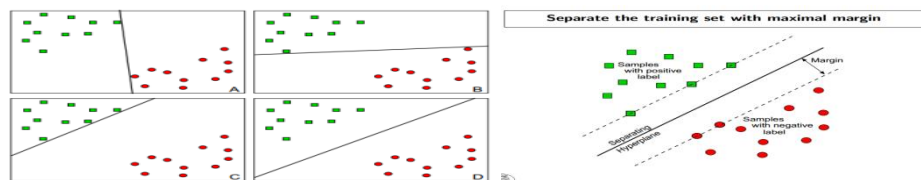


Figure I.3: different possibility of a hyperplane and best hyperplane position

The main concepts of SVM are to first transform input data into a higher dimensional space by means of a kernel function and then construct an OSH between the two classes in the transformed space. Those data vectors nearest to the constructed line in the transformed space are called the support vectors [BEG,RKP,MOW,05]. The SVM estimates a function for classifying data into two classes [VVA.00]. Using a nonlinear transformation that depends on a regularization parameter [BEG,RKP,MOW,05], the input vectors are placed into a high-dimensional feature space, where a linear separation is employed.

Some common kernels include:

$$\text{Polynomial (homogeneous): } k(\vec{x}_i, \vec{x}_j) = (\vec{x}_i, \vec{x}_j)^d \quad (10)$$

$$\text{Gaussian radial basis function: } k(\vec{x}_i, \vec{x}_j) = \exp(-\gamma \|\vec{x}_i - \vec{x}_j\|^2), \text{ for } \gamma > 0.$$

$$\text{Sometimes parameterized using } \gamma = 1/2\delta^2. \quad (11)$$

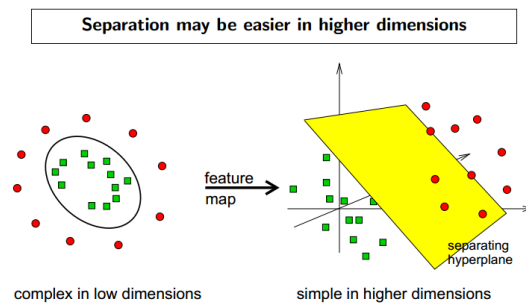


Figure II.4: difference between class separation in low dimension and high dimension

3.3.2. The advantages of support vector machines are:

- Effective in high dimensional spaces.
- Still effective in cases where number of dimensions is greater than the number of samples.
- Uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.
- Versatile: different Kernel functions can be specified for the decision function. Common kernels are provided, but it is also possible to specify custom kernels.

3.3.3. The disadvantages of support vector machines include:

- If the number of features is much greater than the number of samples, the method is likely to give poor performances.
- SVMs do not directly provide probability estimates, these are calculated using an expensive five-fold cross-validation (see Scores and probabilities, below).

II.3.4.LDA:

3.4.1.LDA description:

Linear Discriminant Analysis: is most commonly [KFU,90][SAX,95] used as dimensionality reduction technique in the pre-processing step for pattern-classification and machine learning applications[MEC,YAA ,05]. LDA is “supervised” linear transformation techniques and computes linear discriminants that will represent the axes that maximize the separation between multiple classes .The goal is to project a dataset onto a lower-dimensional space with good class-separability in order to avoid over fitting, and also to reduce computational costs. The original linear discriminant was described for a 2-class problem and it was

then later generalized as “multi-class Linear Discriminant Analysis” or “Multiple Discriminant Analysis” [CRR,48].

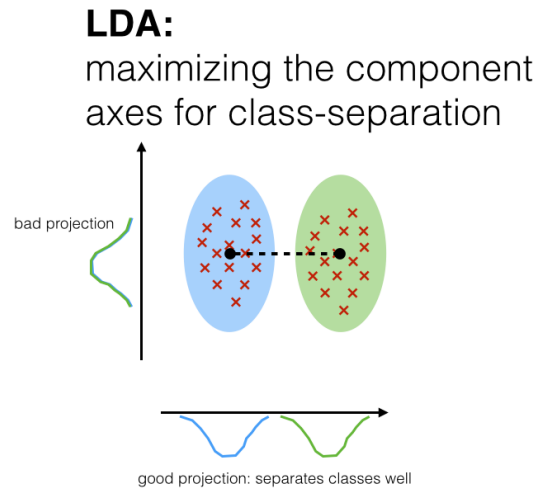


Figure II.5: maximizing the component axes for class separation

3.4.2. Mathematical formulation of LDA:

LDA can be derived from simple probabilistic models which model the class distribution of the data $P(X|y=k)$ for each class k . predictions can be obtained by using Bayes' rule:

$$P(y=k|X) = \frac{P(X|y=k)P(y=k)}{P(X)} = \frac{P(X|y=k)P(y=k)}{\sum_l P(X|y=l)P(y=l)} \quad (12)$$

more specifically in LDA, $P(X|y=k)$ is modeled as a multivariate Gaussian distribution :

$$P(X|y=k) = \frac{1}{(2\pi)^n |\Sigma_k|^{1/2}} \exp\left(-\frac{1}{2}(x - \mu_k)^t \Sigma_k^{-1}(x - \mu_k)\right) \quad (13)$$

3.4.3 .LDA 5 approaches steps:

Listed below are the 5 general steps for performing a linear discriminant analysis:

- 1- Compute the d-dimensional mean vectors for the different classes from the dataset.
- 2- Compute the scatter matrices (in-between-class and within-class scatter matrix).

2-1 within-class:

The **within-class scatter** matrix S_w is computed by the following equation:

$$S_w = \sum_{i=1}^c S_i \quad (14)$$

$$\text{where } S_i = \sum_{x \in D_i} (x - m_i)(x - m_i)^T \quad (15)$$

or Alternatively, we could also compute the class-covariance matrices by adding the scaling factor $\frac{1}{N_i - 1}$ to the within-class scatter matrix, so that equation becomes:

$$\Sigma_i = \frac{1}{N_i - 1} \sum_{x \in D_i} (x - m_i)(x - m_i)^T \quad (16)$$

$$\text{and } S_w = \sum_{i=1}^c (N_i - 1) \Sigma_i$$

where N_i is the sample size of the respective class.

2-2 between-class:

the between-class matrix S_B is computed by the following :

$$S_B = \sum_{i=1}^c N_i (m_i - m)(m_i - m)^T \quad (17)$$

where m is the overall mean, and m_i and N_i are the sample mean and sizes of the respective classes.

3- Compute the eigenvectors (e_1, e_2, \dots, e_d) and corresponding eigenvalues

($\lambda_1, \lambda_2, \dots, \lambda_d$) for the scatter matrices.

4- Sort the eigenvectors by decreasing eigenvalues and choose k eigenvectors with the largest eigenvalues to form a $k \times d$ dimensional matrix W (where every column represents an eigenvector), we can call it: Selecting linear discriminants for the new feature subspace

5- Use this $k \times d$ eigenvector matrix to transform the samples onto the new subspace. This can be summarized by the mathematical

equation: $Y = X \times W$ (where X is a $n \times d$ -dimensional matrix representing the n samples, and y are the transformed $n \times k$ -dimensional samples in the new subspace).

3.4.4. LDA advantages:

Only few parameters to estimate; accurate estimates.

3.4.5. LDA limitations:

-Inflexible (linear decision boundary)

-LDA is a parametric method since it assumes unimodal Gaussian likelihoods:

If the distributions are significantly non-Gaussian, the LDA projections will not be able to preserve any complex structure of the data, which may be needed for classification.

-LDA will fail when the discriminatory information is not in the mean but rather in the variance of the data.

I.4. Conclusion:

we have reviewed in this chapter different classification methods beginning with K-NN through NAÏVE BAYES and SVM ending with LDA, and their approaches including their mathematical functions. We have mentioned also some of advantages and limits of these algorithms

CHAPTRE III

Results and discussions

III.1. Introduction:

After having presented classification Methods in the previous chapter. We will present in the following the implementation of our application. We initially start with a chosen programming tool as the programming language and the reasons behind this choice are justify in next paragraph. Then we will describe the interface of our application.

III.2. Experiment Condition:**III.2.1- hard ware:**

all tests were performed on a personal computer DELL I3 with a 3GB capacity memory, and central process units Intel (R) Core (TM) 2 Duo 2.3GHz, with windows 7 edition integral, service pack 1 32 bit system type as an installed operating system.

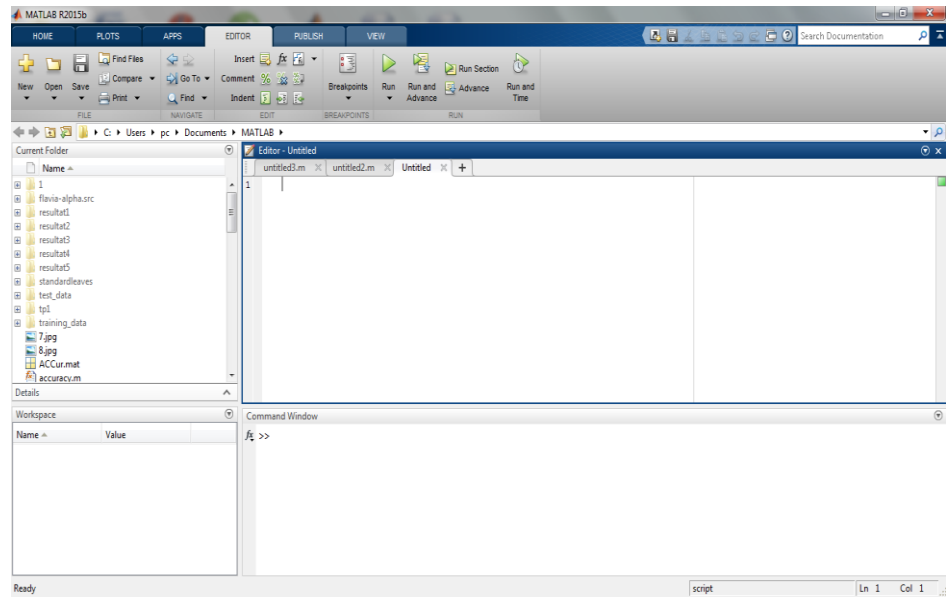
So we're not sure it would work well on other operating systems such as Linux.

III.2.2-software:

In order to realize our application, we have used the environment Matlab which bases for matrix Laboratory. MATLAB was written originally to provide easy access to matrix software developed by the LINPACK (linear system package) and EISPACK (Eigen system package) projects.

Matlab is a high-performance language for technical computing. It integrates Computation, visualization, and programming environment. Furthermore, MATLAB is a modern programming language environment: it has sophisticated data structures, contains built-in editing and debugging tools, and supports object-oriented programming. These factors make MATLAB an excellent tool for teaching and research.

MATLAB has many advantages compared to conventional computer languages (e.g.C, FORTRAN) for solving technical problems. It has powerful built-in routines that enable a very wide variety of computations. It also has easy to use graphics commands that make the visualization of results immediately available.



FigureIII.1: Matlab environment

III.3.System presentation:

We will now present the results obtained through experiments realized by the application of our approach to classification of plant leaves based on geometrical and morphological features.

The scheme of our system is as follows:

- 1) Extraction of features (morphological and geometrical features).
- 2) Classification of leaves using KNN and Naive Bayes algorithm.
- 3) Recognition of plant leaves.

We try to create a GUI that grouping as possible to the implementation details of our application in order to offer a simple and friendly-user for the user of this tool. After running the application, the main window "classification application" will showing up. This window gives the accessibility for user to use our application:



Figure III.2: interface of our application

The "ENTER" button on the interface used to start the other components and display a new window "Classification Application" as following:

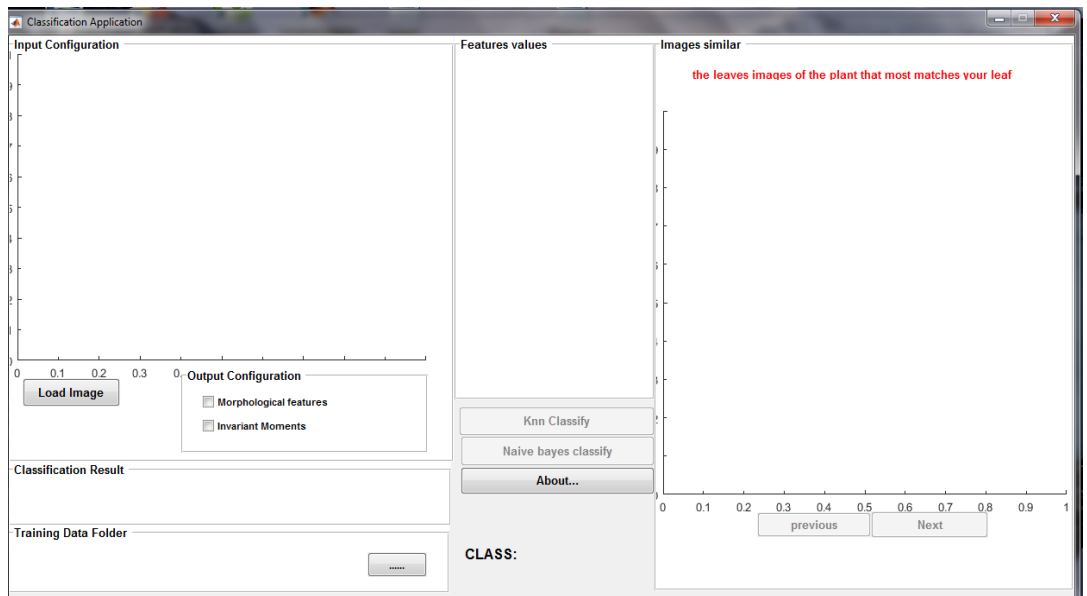


Figure III.3: classification application window.

This window contains buttons that can load image, classification with KNN and Naive Bayes, see the value of features of the image and consult training data set of our work, and others as About, next, previous, etc. ...

- 1) **Load image:** Click on button "load image" displays a new dialog box to load the image.

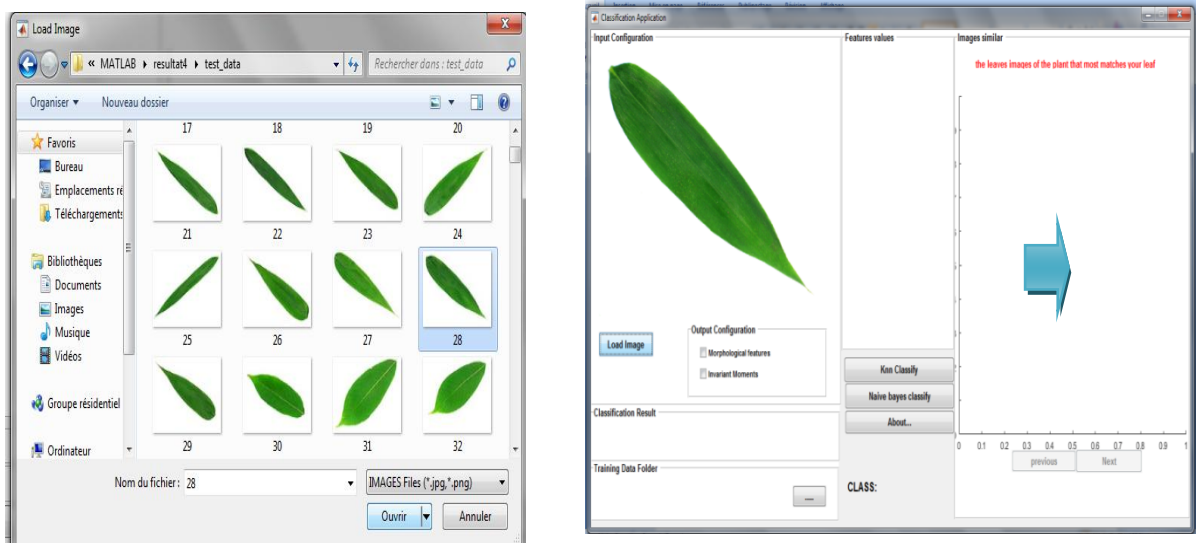


Figure III. 4: loading image

- 2) **Morphological features and Hu-moments checkboxes:** These two items offers the possibility to perform the following operations:

2.1) calculate the morphological features:

- Perimeter
- Area
- Diameter
- Major axis length
- Minor axis length

2.2) calculate the invariant moments

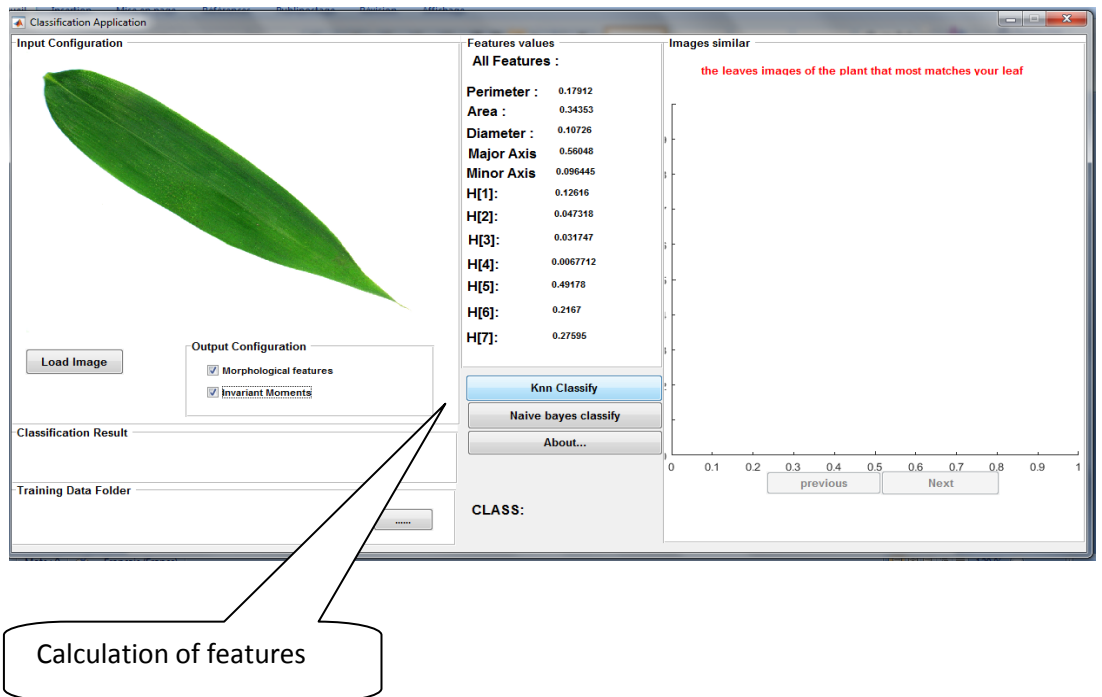


Figure III.5: result of calculate morphological and geometrical features

3) Classification KNN and Naive Bayes:

Click on these two buttons displays the result of the classification which is the name of that leaf with two algorithms KNN and naive Bayes.

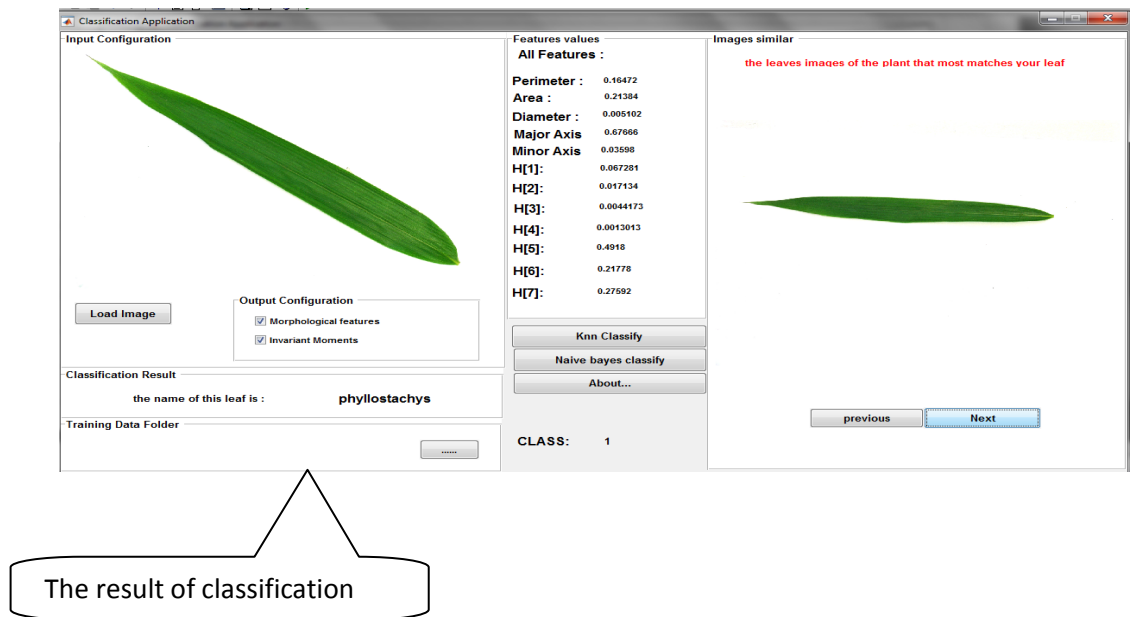


Figure III. 6 : the result of classification

4) Next and previous :

Clicking on these two buttons displays a set of images which are similar with input image

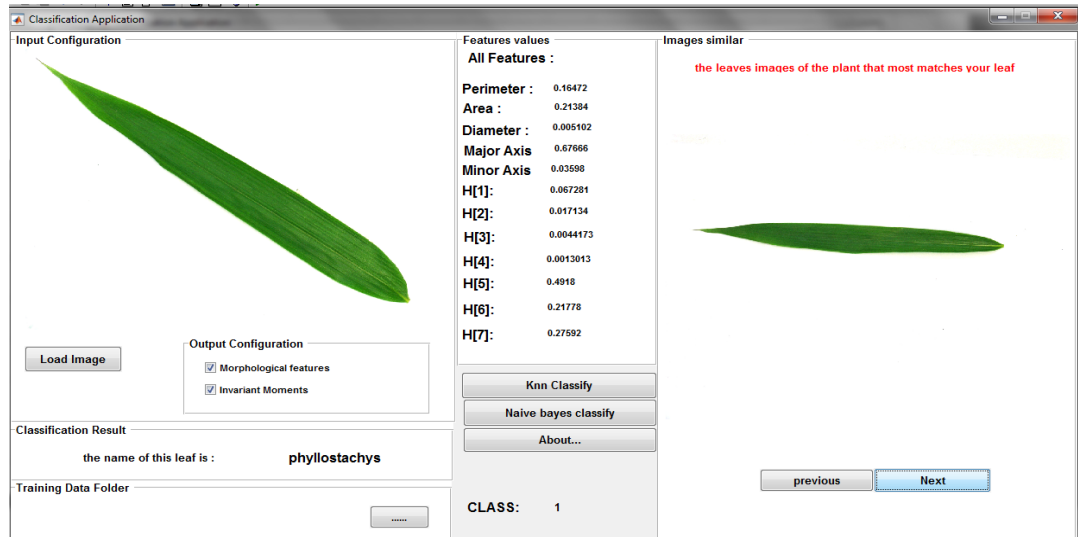


Figure III.7 : leaves images similarity

5) About :

Click in this button an interface appear which Showing information classification algorithms used and the students who developed this application.

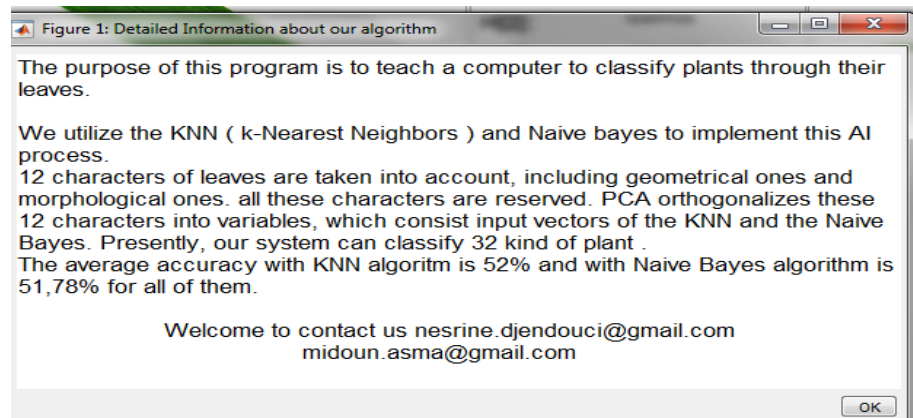


Figure III. 8: About application

III.4.Experiments and results:

The proposed technique are tested on flavia database this consist used 1888 leaf images of 32 species divide on 960 leaf image for learning system (training data) and 928 leaf image for testing system (test data) .

To evaluate the performance of the proposed leaf recognition system, a recognition model was created using a range of values with 12 features for each leaves. To each kind of plant, 29 pieces of leaves from testing sets are used to test. For classifiers were implemented, namely: K-nn and naïve Bayes .The average correct recognition rate of the two algorithms is shown in table1

	classifier	Training set Vs. Testing set	Features	accuracy
flavia	KNN	930 Vs. 928	MF*	47.22%
		930 Vs. 928	HM*	26.28%
		930 Vs. 928	MF and HM	52.48%
	Naive Bayes	930 Vs. 928	MF	45.88%
		930 Vs. 928	HM	18.57%
		930 Vs. 928	MF and HM	51.78%

HM*: Hu-moment

MF*: Morphological features

Table 1 : results tabulated for flavia dataset tested on our classifiers

III.5. Result Analysis

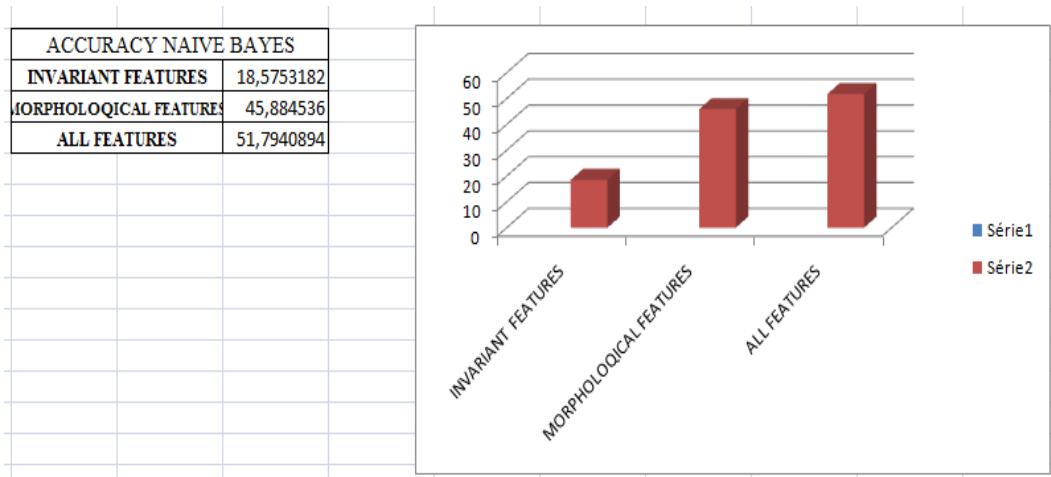


Figure III.9: a graph of naive Bayes accuracy with invariants features and morphological features and both of them

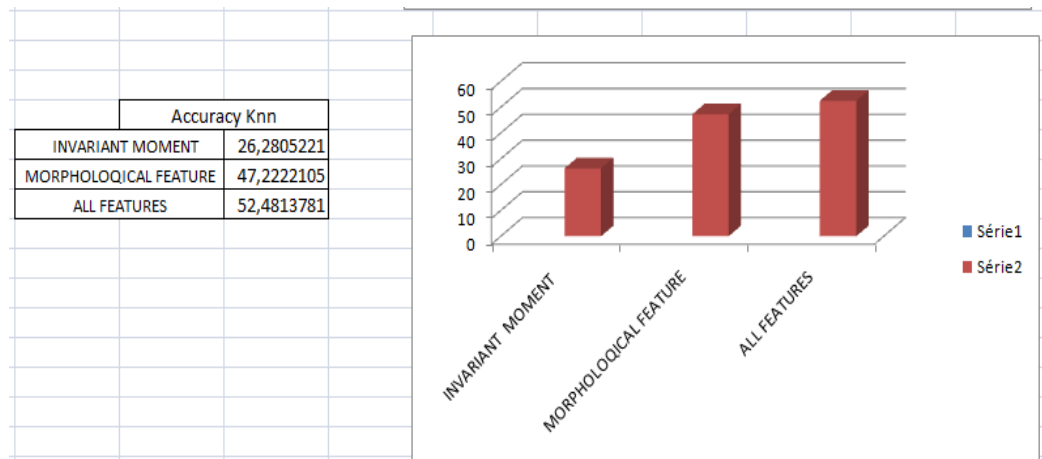


Figure III.10: a graph of K-nn accuracy with invariants moment and morphological features and both of them.

In this section we have reported the accuracies yielded by the different classifiers. So according to these results, we notice that invariant moment gives low accuracy in both algorithms, whereas morphological features give high accuracy in both algorithms comparing with the previous one. However, by combining Hu moments with morphological features, the accuracy is rising up to 51% in naïve Bayes and more than 52 % in KNN. Those values of accuracies are because results are influenced by the fact that there is a small inter-variation (strong visual

resemblance between leaves belonging to distinct classes) in Flavia dataset. So we coming up with an important result that which is in our data K-nn have superior performance than Naive Bayes (with both invariants and morphological features) because it builds a classification without any assumptions about data distribution (non-parametric). However naive Bayes has less accuracy because of it has strong feature independence assumptions.

III.5. Conclusion

We have reviewed in this chapter our application realization through a Matlab environment, and we have showing some details with screen shots about our application, then we have discussed the percentage of accuracy value represented in graphic representation.

General Conclusion

Plants identification and classification has always been a matter of interest for the botanists as well as for the laymen.

In this work, we have developed a system for leaves classification using machine learning techniques. First, we have splitted the dataset of images into two parts: the first is used for learning, whereas the second is intended for testing. The learning process is started by extracting Hu moments and five morphological features from all the images. The morphological features are area, perimeter, and major and minor axis length. The learning process is finalized by training each of the KNN and Naïve Bayes classifiers with the features previously extracted. Afterwards, a testing process is carried out to measure the performance of our system. Our system can easily be integrated in industry, as it doesn't require any human assistant. In addition, using Hu moments permit to avoid any constraints on the position of leaf during acquiring its image. That's because Hu moments are invariant to scale, rotation and position.

We have tested our system on the well-known Flavia dataset which made up of 1907 leaf images from 32 classes. Experimental results showed a promising accuracy of more than 50%.

Bibliographie

1. **B.E. Boser et al (1992)**: A Training Algorithm for Optimal Margin Classifiers. Proceedings of the Fifth Annual Workshop on Computational Learning Theory 5 144-152, Pittsburgh.
2. **Bedouhenesaida** : Recherché d'image par le contenu Traitement d'images et reconnaissance de formes, mémoire de magister,tizi-ouzzo.
3. **Begg, R.K., Palaniswami, M., Owen, B., 2005**. Support vector machines for automated gait classification. IEEE Transactions on Biomedica Engineering 53 (5), 828–838.
4. **C. Radhakrishna Rao (1948)**: The utilization of multiple measurements in problems of biological classification ournal of the Royal Statistical Society. Series B (Methodological)Vol. 10, No. 2, pp. 159-203.
5. **C.H. The, R.T. Chin (1988)**: On image analysis by the method of moments, IEEE Trans. Pattern Anal.Mach. Intell. 10 (4) 496–513.
6. **Chee-Way Chonga, P. Raveendranb and R.Mukundan (2004)** : Translation and scale invariants of Legendre moments, Pattern Recognition 37 119 – 129.
7. **CHRIS SOLOMON TOBY BRECKNO** **audition: WILEY-BLACKWELL**, fundamentals of Digital Image Processing:A pratical Approach With examples In Mtlab author.
8. **D. ZHANG, G. LU (2005)**: Study and evaluation of different Fourier methods for image retrieval, Image Vision computing, vol.23.
9. **Den, Guo, Dengsheng Zhang, Guojun Lu (2002)**: Review of shape representation and description techniques, Pattern Recognition 37.
10. **K. Fukunaga (1990)**: Introduction to Statistical Pattern Recognition, Academic Press, San Diego, California.
11. **K. Singh, I. Gupta, and S. Gupta, (2010)**: SVM-BDT PNN and Fourier Technique for Classification of Leaf Shape. International Journal of Signal Processing : Image Processing and Pattern Recognition, Vol. 3, No. 4, December,2010
12. **Khotanzad, A. and Hong, Y.H** : Invariant Image Recognition by Zernike Moments. IEEE Trans. On PAMI,12 (5). 289-497.
13. **Khotanzad, A. and Hong, Y.H**: Rotation Invariant Image Recognition using Features Selected via a Systematic Method. Pattern Recognition, 23 (10). 1089-1101.
14. **Lipscomb, J.S. A Trainable Gesture** : Recognizer. Pattern Recognition, 24 (9). 895-907.
15. **M. E. Celebi and Y. A. Aslandogan (2005)**: A comparative study of three moment-based shape descriptors, in Proc. Of the International Conference of Information Technology: Coding and Computing, pp. 788793.
16. **M.R Teague**: Image analysis via the general theory of moments, J. Opt Soc. Am. 70 (8) 920–930.

17. **M.-K. Hu (1962)**: Visual pattern recognition by moment invariants, IRE Trans. Information Theory, vol. IT-8, pp. 179187.
18. **MARTIN LAW (2011)**: A simple introduction to Support Vector Machines, Michigan State University.
19. **Mingqiang Yang, Kidiyo Kpalma, Joseph Ronsin (2008)**: A Survey of Shape Feature Extraction Techniques. Peng-Yeng Yin. Pattern Recognition, IN-TECH, pp.43-90, <hal-00446037>.
20. **R. Mukundan, K.R. Ramakrishnan (1998)**, Moment Functions in Image Analysis—Theory and Applications, World Scientific, Singapore.
21. **R. M. Haralick (1979)**: Statistical and structural approaches to texture. In *Proc. IEEE*, volume 67(5), pages 786-804.
22. **S. Axler (1995)**: Linear Algebra Done Right, Springer-Verlag New York Inc., New York, New York.
23. **Teague, M.R**: Image Analysis via the General Theory of Moments. Journal of the Optical Society of America, 70 (8). 920-930.
24. **The, C. and Chin, R.T** : On Image Analysis by the Methods of Moments. IEEE Trans. On PAMI, 10 (4).496-513.
25. **V. Vapnik (2000)**: The Nature of Statistical Learning Theory. 2nd edition, Springer,New York.(2000)
26. **Vis, Jay, (2013)**: An Overview of the Research on Texture Based Plant Leaf Classification IJCSN International Journal of Computer Science and Network, Vol 2, Issue 3.
27. **Xindong Wu · Vipin Kumar · J. Ross Quinlan · Joydeep Ghosh · Qiang Yang · Hiroshi Motoda · Geoffrey J. McLachlan · Angus Ng · Bing Liu · Philip S. Yu · Zhi-Hua Zhou · Michael Steinbach · David J. Hand · Dan Steinberg(2007)**, SURVEY PAPER, Springer-Verlag London Limited,04.