Automatic detection and classification of grains of pollen based on shape and texture

Article *in* IEEE Transactions on Systems Man and Cybernetics Part C (Applications and Reviews) · August 2006

DOI: 10.1109/TSMCC.2005.855426 · Source: IEEE Xplore

Automatic Detection and Classification of Grains of Pollen Based on Shape and Texture

María Rodríguez-Damián, Eva Cernadas, Arno Formella, Manuel Fernández-Delgado, and Pilar De Sá-Otero

***Abstract*—Palynological data are used in a wide range of applications. Some studies describe the benefits of the development of a computer system to pollinic analysis. The system should involve the detection of the pollen grains on a slice, and their classification. This paper presents a system that realizes both tasks. The latter is based on the combination of shape and texture analysis. In relation to shape parameters, different ways to understand the contours are presented. The resulting system is evaluated for the discrimination of species of the *Urticaceae* family which are quite similar. The performance achieved is 89% of correct pollen grain classification.**

***Index Terms*—Pollen classification, pollen grain, segmentation, shape analysis.**

I. Introduction

PALYNOLOGICAL DATA are used in a wide range of applications such as paleoclimatic reconstruction, allergenic process, and food technology. Fossil pollen spectra are used for paleoenvironmental reconstruction in the Quaternary and in the geological past. The count of pollen in the air allows us to suggest safety treatments to patients who are allergic to pollen. Pollinic analysis of honeybees is used to determine their geographical origins.

The principal tasks in almost all of these applications are *classification* and *counting*, which are laborious and time consuming due to the fact that they are done manually by highly skilled experts. A high level of training is needed to obtain accurate identification results. The morphology of the grains is fundamental to identify the type of pollen. The key parameters are shape, polarity, symmetry, aperture, size, exine stratification, and sculpturing type of the grain. The study of a preparation usually requires the classification of a great number of pollen grains. In quaternary studies, the number is in the range of 300–500 pollen grains, but in honeybee applications the number increase to 750–1500. The studies maintain a trained person occupied for 2–10 h on average. Another issue is that the pollen identification problem is subjective, due to the identification by using "personal equations" [1].

The application of computer vision techniques to pollen analysis is relatively innovative. In 1996, Stillman and Fleney identified the needs in palynology and summarized the advantages that would arise if an automatic system could be developed [1].

The most ambitious attempt is described in [2], where two important tasks are identified: finding the locations of the pollen grains in an image taken from a slice, and classifying the pollen grains into taxonomic categories. They employ neural networks for both steps. Others studies focus on texture analysis of digitized images of the exine surfaces of the pollen grain on scanning electron microscope (SEM) images [3], [4]. This technique was successful, but seems difficult to fully automate, because it requires the preparation of a whole sample for SEM analysis.

In routine analysis, most pollen grains are identified only to the genus level, and rarely beyond to the family level. A distinction down to finer levels (even to species) would be possible in a number of cases with the more sophisticated procedures such as SEM or biometrical analysis. These techniques require laborious measurements of individual grains, and they are not well suited for systematic pollen counts on a simple slice. Nevertheless, the analysis would be far more informative for the palynologist if these distinctions were possible.

Hence, the benefits of the development of a system for automated pollen analysis are clear. The system should be able to operate without requiring specialized preparation procedures beyond those currently in use for pollen count. Further, it should involve the detection of the position of the pollen grains on a slice and their classification into taxonomic categories. The system should consist of a standard optical microscope connected to a computer, i.e., a digital imaging system, and a computer to undertake the analysis of the objects that are found on the slice. This paper only focuses on the analysis.

All automatic approaches reporting on pollen classification involve pollinic types that human operators are able to distinguish in routine pollinic analysis. However, we are interested in discriminating species of the same family *Urticaceae*, namely, *Parietaria Judaica*, *Urtica Urens*, and *Urtica Membranacea* which is not achieved in routine pollinic analysis.

Many studies about the *Urticaceae* family have been described in the palynologist literature. Their main conclusion is: the morphologies of the pollen are quite similar; the pollen grains are small and of almost equal size (13–17 μm in *Parietaria Judaica*, 13–19 μm in *Urtica Urens*). Both genera exhibit similar roundness and do not have excessive ornamentation of exine, which is always microechinate. The distribution of small echinae in *Urtica Membranacea* is more regular. There are some

Manuscript received January 20, 2002; revised November 18, 2002. This work was supported in part by Vigo University Project *Automatic classification and count of pollen from digital optical microscope images* and in part by Spanish CICyT under Project TIC2003-09400-C04-03.

M. Rodríguez-Damián, E. Cernadas, and A. Formella are with the Department of Computer Science, ESEI, Universidade de Vigo, Campus Universitario As Lagoas, 32004 Ourense, Spain.

M. Fernández-Delgado is with the Intelligent System Group, Department of Electronics and Computing, Universidade de Santiago de Compostela, 15782 A Coruña, Spain.

P. De Sá-Otero is with the Department of Vegetal Biology and Soil Science, Facultad de Ciencias, Universidade de Vigo, Campus Universitario As Lagoas, 32004 Ourense, Spain.

Digital Object Identifier 10.1109/TSMCC.2005.855426

apertures of type pore (3 to 4 porate in *Parietaria Judaica*, and 3 to 4 zonoporate or pantoporate in *Urtica Urens*) [5].

Experts are able to distinguish the species of the family *Urticaceae* using SEM, but they are unable to do it with sufficient accuracy using an optical microscope. Even with biometrical studies, Dopazo *et al*. [6] were not able to discriminate the species *Urtica Dioica*, *Urtica Urens*, and *Parietaria Judaica* of the *Urticaceae* family.

The importance of the family is described in Section II. Afterward, we present an automatic algorithm to distinguish the species of the family *Urticaceae* combining shape and texture analysis.

II. Medical Relevance of the Family *Urticaceae*

The *Urticaceae* family is a plant type which grows especially in soils rich in nitrogen, such as overgrowing weeds in cultivations or abandoned lands near inhabited places. In the Iberian peninsula, seven species of the family *Urticaceae* grow in a natural way. We concentrate on the more distributed genera, namely, *Urtica* (*Urtica Urens L.*, *Urtica Membranacea Poir. in Lam.*) and *Parietaria* (*Parietaria Judaica L.*).

The species have periods of extensive and overlapping flowering throughout most parts of the year, during which the pollen count in the air is of special importance. Their anatomical characteristics facilitate the propagation of the pollen in the air. During the major blossom months, the pollen grains are transported by the wind in copious amounts. When coming into contact with the respiratory tract in humans and in particular on the mucous surface, pollen grains may produce important alterations in the organism, such as *allergic rhinitis* and *asthma type I*. Although both genera, *Urtica* and *Parietaria*, are allergenic [7], [8], the pollen of the genus *Parietaria* is one of the most common and the one with greater allergenic potential [9]–[11].

In clinical practice, it is possible to distinguish the sensitivity to *Parietaria* or to *Urtica*, but an antigenic cross reaction between the two genera has not been detected [12], [13]. Since the allergenic potentials are different, the personal sensitivity is specific to each specie. Thus, clinical practice shows special interest in the differentiation of the following species: *Parietaria Judaica*, *Urtica Urens* and *Urtica Membranacea*. In addition, it would be useful for clinical practice to count the relative percentage of pollen of each genus in the air. If the pollen count could be systematically and periodically repeated, it would enable recommendations that patients take the appropriate safety measures.

III. Proposed Method

Our attempt is to classify and count the grains of pollen in a slice that represents the visual information reported by an optical microscope. The method consists of two steps: detection of pollen grains in the slice, and their classification. The detection step includes two phases: finding the location of the grains and extracting their silhouette. The classification step also includes the phase of computing the feature vector that represents each pollen grain.

As previously mentioned, palynologists have indicated that the major criteria for distinguishing the pollen into taxonomic categories are their morphology (shape) and their sculpture (texture of the inner grain). Our approach is based on combining both the morphological and the texture parameters of the grain.

Although there exist databases of pollen images, some available via the world wide web, there is none available that includes species of the *Urticaceae* family. So, we have captured our own images, which will be described in Section VIII.

Sections IV to VII explain the technical details of every step of the system. The results of the experiments are presented in Section VIII. In Section IX, we discuss possible future directions of our investigations and our main conclusions are described in Section X.

IV. Detection of Grains of Pollen in a Slice

Since the pollen grains of the *Urticaceae* family are nearly circles, the detection of their locations can be seen as a circle detection problem. The Hough transform (HT) [14], [15] is a well-known method to detect circles or circular arcs in images. It is basically a template matching scheme, which maps every edge pixel in the image onto a conic surface. The circle in the edge image space with center (x_o, y_o) and radius r_o is defined by the expression: $(x - x_o)^2 + (y - y_o)^2 = r_o^2$, yielding the conic surface in the three-dimensional (x_o, y_o, r_o) parameter space. Generally speaking, the application of the HT to detect circles in a gray level image involves the generation of a binary edge map and the selection of a suitable range of radii and thresholds. The performance of the HT approach strongly depends on these three critical facts: the input binary edge map, the range of radii to search in, and the threshold values to detect local maxima in the parameter space. A binary edge map from the original image can be generated by a variety of methods as described, for instance, in [16].

The binary edge map should only contain the true outlines of the grains of pollen. This is frequently not possible to achieve, because the slices show complex scenes which usually present brightness inhomogeneities and debris. The selection of a threshold for the binarization is a difficult but important task. Inaccuracy in the detection of the true boundaries of the objects will be propagated into the following steps and might significantly degrade the final system performance. In an effort to achieve the desired binary edge map, we proposed an empirical combination of filters applied to the gray level image [17].

After downsampling the original image of 2048×3072 pixels, the algorithm consists of the following steps. The image is filtered by a median filter [18] to attenuate the spurious noise, the mask size is 3×3 pixels. Then, a gradient operator (e.g., the Sobel operator) is applied to obtain an edge strength image, which is binarized by thresholding it. The value of the threshold is again a critical decision. We choose it from the content of edge strength image using the Otsu method [19], [20]. As no edge detection is perfect, there will be inevitably spurious pixels and other abnormalities in the detected sections. Also, the edges are normally broken. We use morphological operators such as erosion (with a matrix of 3×3 pixels) to further filter spurious pixels [21]. The result of this operation is subtracted

```
FOR every slice in the data set DO
  1. Down sample original image four times.
  2. Filter image 1 by a 3x3 median filter.
  3. Apply Sobel operator to image 2.
  4. Calculate threshold with Otsu algorithm.
  5. Threshold image 4.
  6. Binary erode image 5 by 3x3 square operator.
  7. Subtract binary image 6 from image 5.
  8. Thinning image 7.
  9. Calculate Hough transform on image 8.
 10. WHILE peaks of the Hough space greater than THRESHOLD DO
       get (x,y) coordinates of location of pollen grain
       up sample (x,y) coordinates to the original image
       extract the grain of pollen.
     ENDWHILE
ENDFOR
```

Fig. 1. Algorithm for the detection process of the location of pollen grains.

from the previous binary image. Thinner borders are computed with a classical thinning algorithm [22].

Once the binary edge map is available, the HT is calculated. Two parameters are necessary to determine when a peak in the Hough space is really a pollen grain. The first parameter represents the possible values for the radio of the pollen grain. The range is estimated, bearing in mind the known size of pollen grains. The second parameter represents a threshold value to indicate which points of the parameter space represent true circles; different values of thresholds will result in different compromises of true positive and false positive errors. Fig. 1 shows an original image, its binary edge map, after having applied the HT. The whole detection process of the location of pollen grains is summarized in Fig. 2.

V. Extraction of the Pollen Silhouette

Accurate extraction of the silhouettes of the pollen grains is the next important task in the shape discrimination problem. We implemented three approaches, one based on the gradient image and two based on active contours. They operate as follows.

A. Edge-Contour Approach

This approach is based on the gradient image. The images which contain isolated grains of pollen are binarized following the process detailed in Section IV. Afterward, closed regions are searched for, the largest region is chosen (which discriminates internal borders and debris), the region is filled, and morphological operators are used to get just the boundary. The boundary of this region is the outline of the pollen grain [23].

B. Snake-Contour Approach

This approach is based on active contours, commonly referred to as deformable models or snakes [24]. Starting from an initial estimation of the object boundary, the snakes can dynamically conform to the true boundary according to the action of the internal resistance (elasticity or rigidity) of the snakes and their external image attraction. Let $\mathbf{u}$ be the discrete deformable contour consisting of n points in the image plane, i.e., $\mathbf{u} = \{u_i = (x_i, y_i), i = 0, 1, \ldots, n-1\}$. For a closed curve, $u_{-1} = u_{n-1}$ and $u_n = u_0$. The energy function of the snake

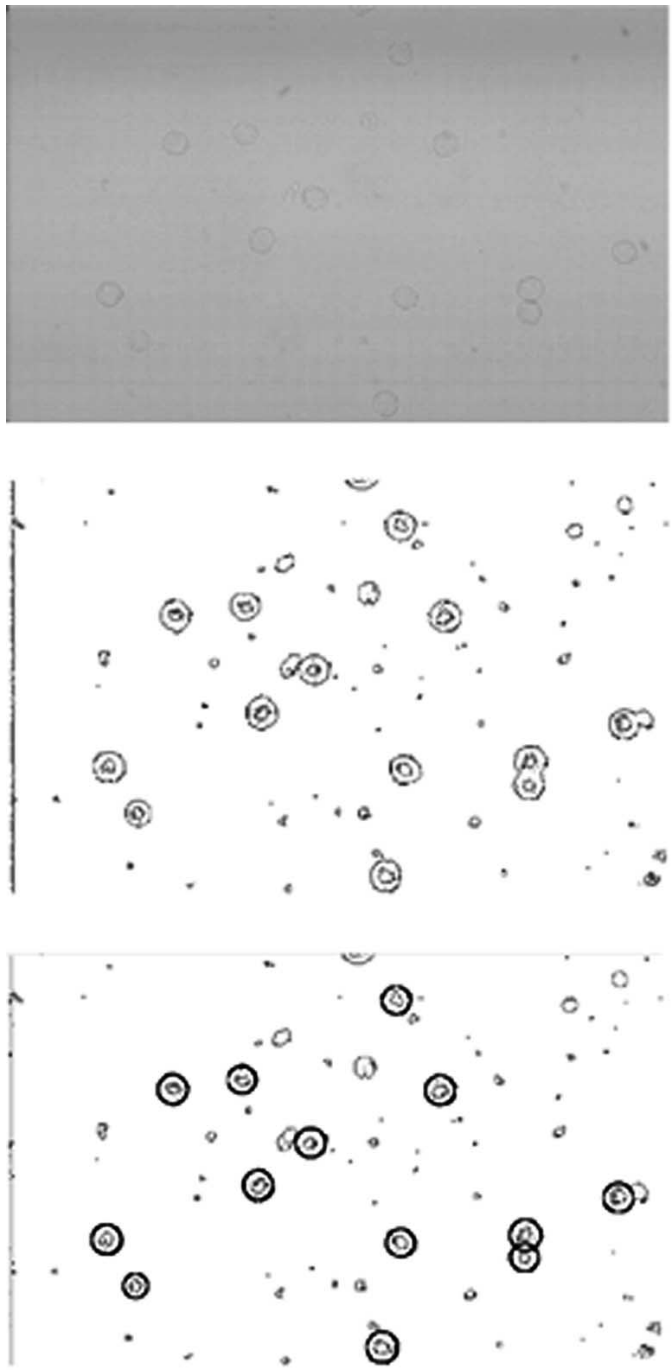

Fig. 2. Original image of a slice (top), its binary edge map (middle), and the detected pollens by HT overlapped on the previous binary image (bottom).

is defined as

$$E_{\text{snake}}(\mathbf{u}) = \sum_{i=0}^{n-1} [E_{\text{int}}(u_i) + E_{\text{image}}(u_i)]. \tag{1}$$

The terms E_{int} and E_{image} represent the internal resistance and external image attraction of the snake. In an iterative process, the curve is modified by minimizing its energy function $E_{\text{snake}}(\mathbf{u})$. The final position corresponds to the minimum

E^*_{snake} of the $E_{\text{snake}}(\mathbf{u})$. We implemented the minimization process with a greedy algorithm [25].

The internal resistance term $E_{\text{int}}(u_i)$ imposes regularity on the curve by bending and stretching. It is defined as

$$E_{\text{int}}(u_i) = \alpha E_{\text{cont}}(u_i) + \beta E_{\text{curv}}(u_i) \tag{2}$$

where the factors α and β control the mechanical properties (i.e., elasticity and rigidity) of the model, and the terms $E_{\text{cont}}(u_i)$ and $E_{\text{curv}}(u_i)$ represent the continuity and local curvature of the snake. The external image attraction term $E_{\text{image}}(u_i)$ serves as a potential field which attracts the snakes toward salient features in the image such as lines or edges, and is defined as $E_{\text{image}}(u_i) = \gamma P_{\text{feature}}(u_i)$ where the factor γ is a weight for the image attraction and the term $P_{\text{feature}}(u_i)$ corresponds to target image features and is commonly set to the image gradient strength.

Some tradeoffs of standard snakes are the selection of the external image energy, the location of the initial snake, and the weights of each energy term. In our application, we calculated the initial boundaries of the pollen grains from the radius and the center of each pollen grain provided by the HT, selecting a set of equidistant points from the circle. We assume constant and equal values for the weights ($\alpha = \beta = \gamma = 0.5$). The image energy is the normalized gradient magnitude after having applied the Sobel operator.

C. Convex-Hull Approach

This approach is also based on the contour approach. An important class of shape representation methods uses the convex-hull as bounding region that encloses the shape of interest. The convex-hull is the smallest convex set that contains the shape. We construct the convex-hull with the QuickHull algorithm [26] from the points of contour.

Fig. 3 shows an example of the boundaries of pollen grains extracted for each species using the three methods: edge-contour, snake-contour, and convex-hull contour.

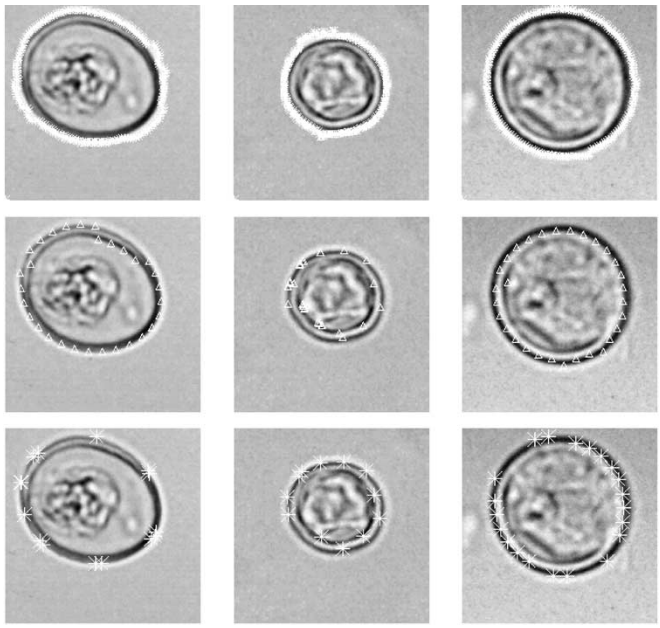

Fig. 3. Grains of pollen of the specie *Urtica* (right column), *Membranaceae* (center column), and *Parietaria* (left column). The first row shows the contour obtained with the edge contour, the second row with the snake contour, and the third row with the convex-hull contour approach.

VI. Computing Shape and Texture Features

One of the fundamental challenges in pattern recognition is choosing an appropriate set of features for the problem, a crucial step for the overall success of the recognition system [27]. The process of selecting effective features for the classification relies on previous experience, intuition, and the peculiarities and variability of the considered objects. Our aim in this paper is to evaluate the recognition performance using combinations of various attributes derived from the silhouette and the texture of the object. The first category may be associated to the pollen morphology, and the second one to the ornamentation or the surface of the pollen grain. In the rest of this section, we describe the object representation and the different feature sets computed for a pollen grain.

A. Representation and Description of Objects

A number of shape description and classification schemes have been developed over the years [28]–[31]. Humans perceive the shape as a property or characteristic of a set of points that define a visual object. The property is determined by geometrical relations between the involved points that are invariant to translations, rotations, and reflections. The main representation methods can be broadly divided into contour-based and region-based approaches. Formally, a discrete two-dimensional binary shape (object or region) can be defined by some finite set S of connected points. Let S be a subset of $\mathbf{Z} \times \mathbf{Z}$; a generic point $(x_i, y_i) \in \mathbf{Z} \times \mathbf{Z}$ can be an *interior*, an *exterior*, or a *boundary* point of S. A binary shape can also be seen as a binary image of $M \times N$ pixels, where $f(x, y) = 1$ if $(x, y) \in S$ and 0 otherwise. In a textured shape, the pixels represent the gray levels or intensity values (for instance, $f(x, y) \in [0, 255]$).

There are different methods to represent the contours of a binary shape S. Let B be the set of boundary points of S. The contour can be represented in the xy plane by a connected ordered set of points $z_i = (x_i, y_i), z_i \in B, B \subset S$ or in the complex coordinate system (in which case $z_i = x_i + jy_i$). Let N_b be the total number of boundary points (or pixels) in B, and let (x_i, y_i) be a pixel on the boundary. Shapes can also be represented by a set of points sampled from the contour of the object. They need not—and typically will not—correspond to key points such as maxima of curvature or inflection points. Operationally, the selection of contour points could be random with roughly uniform spacing.

Shape description can be made using the following schemes: contour approximation [30], discrete binary shape [28], or structural representation [31]. Contour approximation selects a number of points from the outline of an object and approximates with curve segments that pass through those points. Usually, the points have some specific properties, such as minimum curvature, inflections between convex and concave parts of a

contour, etc. In structural shape representation, a shape is described in terms of simpler shape components, or shape primitives and the relationships among the shape primitives. However, a substantial body of work assumes that shape is a characteristic of a binary image region. Shape analysis uses either the boundary of an object or its interior to compute shape descriptors. These approaches make use of the spatial arrangements of pixels and are called silhouette-based methods [32]–[34]. Spatial silhouettes contain the internal and the boundary pixels of the object.

All the methods mentioned previously ignore the internal information of the object. The intensity dependencies among interior points of the object may also provide relevant information. This idea was also exploited in [35] to recognize occluded objects in a scene. Broadly speaking, the descriptors can be categorized into boundary-based, region-based, and texture-based features, which are indicators of the outline, the spatial arrangement, and the texture of the object. In the following sections, we describe the feature sets that we have computed.

B. Common Geometrical Features (CGF)

Geometrical information is defined in an informal way as a natural or visual measure of the shape. Let $z_c = (x_c, y_c)$ be the centroid of a binary shape S with contour $B, B \subset S$. The geometrical shape vector encloses the following 18 measures [28].

1) Area (A): The area of an object S is defined as the number of internal points of S.
2) Ratio of area and perimeter (RAP): This parameter is the ratio between the area and the perimeter $(\mathrm{RAP} = A/P)$. The perimeter (P) of an object S can be defined as the arc length of its contour. We estimate the arc length as the sum of the distances between the points of the boundary of object, i.e., $P = \sum_{i=0}^{N_b} |z(i+1) - z(i)|$ where $z(N_b) = z(0)$.
3) Diameter (D): D is the largest distance between any two points on the boundary of the object.
4) Compactness (C): The compactness is calculated as $C = P^2/(4\pi A)$.
5) Roundness (R): It is the inverse of the compactness $R = 1/C$.
6) Rates of changes $(R_{Mm}, R_{M\mu}$, and $R_{m\mu})$: The measures of $D_\mu, D_{\max}$, and $D_{\min}$, respectively, are defined as the average, the largest, and the smallest distance between the centroid and the boundary of the object. The rates of changes of these measures are given as $R_{Mm} = D_{\max}/D_{\min}, R_{M\mu} = D_{\max}/D_\mu$, and $R_{m\mu} = D_{\min}/D_\mu$.
7) Holes (H): The holes are defined as
$$H = \sum_{i=0}^{N_b - 1} D_{\max} - \|z_c - z_i\|.$$
8) Thickness (T): The thickness T is the number of times that the binary object can be eroded by a 3×3 squared filter mask until it vanishes.
9) Elongatedness (El):
$$El = A/(2T^2).$$
10) Centroid size (Cs):
$$Cs = \sqrt{\frac{1}{N_b} \sum_{i=0}^{N_b - 1} [(x_i - x_c)^2 + (y_i - y_c)^2]}.$$
11) Euclidean norm (En): Defined as the centroid size multiplied by the factor $1/\sqrt{2}$.
12) Mean size (Ms):
$$Ms = \frac{1}{N_b} \sum_{i=0}^{N_b - 1} \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2}.$$
13) Eccentricity (Ec): The eccentricity estimates the ratio of the major axis and the minor axis of the best fit ellipse over the object. The axes are determined by the second-order central moments of the binary shape S.
14) Bending energy (BE): The bending energy is given by the curvature normalized with the boundary length P, i.e.,
$$\mathrm{BE} = \frac{1}{P} \sum_{i=1}^{N_b} C^2(i).$$
15) (RC): related to the roundness and circularity $\mathrm{RC} = (P - \sqrt{P^2 - 4\pi A})/(P + \sqrt{P^2 - 4\pi A})$.

C. Statistical Moments

Moments are widely used for shape classification [28], [33], [36], [37]. They can be derived from a binary, a boundary, and a textured shape. The first two types of moments give information about the mass distribution of the object, and the last type is associated with the spatial distribution of the gray levels. Obviously, the boundary-based moments are computationally less expensive, since only boundary points need to be considered in the calculations.

Let $S \subset \mathbf{Z} \times \mathbf{Z}$ be a textured shape. The definition of the order moments m_{pq} of a textured shape generally refers to all pixel locations $(x, y) \in S$ and pixel values $f(x, y)$

$$m_{pq} = \frac{1}{R} \sum_{(x,y) \in S} x^p y^q f(x, y) \tag{3}$$

where R is a normalization constant. The moments of a binary shape are a special case in which $f(x, y) = 1$ for all $(x, y) \in S$ and $R = 1$. In a textured shape, R is the number of pixels in the region S.

For binary shapes, the lowest order moment m_{00} simply represents the sum of pixels of the region, and gives a measure of the area. It is not used very often to discriminate shapes, because a given object may comprise a smaller or larger portion of an image depending on the scale of the picture, the distance of the object from the observer, and the perspective. However, in our problem, every pollen grain in the slice is taken at the same scale and the area could be an important feature to discriminate the pollen. The first-order moments in the x and y direction normalized by the area yield the centroid (x_c, y_c). They determine the location of the pollen, which we are not interested in. For textured shapes, m_{00} represents the mean gray level.

The central moments μ_{pq} represent descriptors of a region which are normalized with respect to their locations in order to obtain translation invariance. They are defined in terms of the location of the centroid

$$\mu_{pq} = \sum_{(x,y)\in S} (x - x_c)^p (y - y_c)^q f(x, y). \tag{4}$$

Usually, the central moments are normalized with respect to the 0th moment to yield the normalized central moment

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}} \quad \gamma = \frac{(p+q)}{2} + 1. \tag{5}$$

From the central normalized moments, Hu [38] derived seven orthogonal invariant moments that are invariant to a general linear transformation.

Although in theory, an infinite number of moments are required to portray an image precisely, only a few moment terms are normally required for most recognition tasks. However, which and how many moments are required for solving the shape discrimination problem is often a critical decision for complex or very similar shapes.

D. Fourier Descriptors

Fourier descriptors (FDs) are a promising technique for shape classification. They are based on the analysis of the boundary points of the shape [28]. The most frequently used shape signatures for FD's calculation are: central distance, complex coordinates (position function), curvature, and cumulative angular function. We use the complex representation of each boundary point. The contour is an ordered set of points $z_i = x_i + jy_i, i = 0, 1, \ldots, N_b - 1$ in the complex plane. The discrete Fourier transform (DFT) of the signature z_i is

$$a(u) = \frac{1}{N_b} \sum_{i=0}^{N_b-1} z_i e^{-j2\pi ui/N_b}, \qquad u = 0, 1, 2, \ldots, N_b - 1. \tag{6}$$

The resulting complex coefficients $a(u)$ yield Fourier descriptors in form of a power spectrum $|a(u)|^2$. High-frequency components of $a(u)$ account for the fine details of the boundary, and lower-terms describe the global shape of it. The requirement for the FDs to be invariant under translation imposes the condition $|a(0)| = 0$. Rotating the contour in the spatial domain and shifting the starting point in the contour does not affect the magnitudes of the FDs. The dependence on scale can be eliminated by dividing each coefficient through $|a(1)|$. However, the area could be an important feature in our shape classification problem; hence, we do not perform the last normalization.

There are three main methods to choose points along the boundary of the shape: 1) equal point sampling; 2) equal angle sampling; and 3) equal arc length sampling. We chose K points and unwrap the contour counterclockwise using an equal angle sample $\theta = 2\pi/K$.

By varying the number of sample points, the accuracy of the shape representation can be adjusted. The larger the number of points, the more details of the shape are represented. Consequently, the matching result will be more accurate. In contrast, a smaller number of sample points reduces the accuracy of the matching results, but improves the computational efficiency.

E. Texture Features

On addition to the statistical moments computed on textured shapes, we tested the performance of other descriptors of the gray level region. Some statistical features provided good performance in many texture classification problems. In most of the applications, the experiments are carried out only to characterize rectangular shapes. A recent study [39] compares and evaluates different texture classification methods with respect to their performance in identifying textures from small and irregular samples.

We compared four different texture classification methods in our context. The first texture vector encloses only first-order statistics (FOS), which provide information about the distribution of the gray levels that fell inside the shape. The vector contains the following six measures [40]: mean gray level, variance, third and four statistical moments, energy, and entropy. The other texture vectors compute second-order statistics. We compute the following vectors:

1) Haralick's Coefficients (HC): The gray level cooccurrence matrix of the image (GLCM), is based on the estimation of the second-order joint conditional probability density functions, $P(m, n, d, \alpha)$ with $\alpha = 0°, 45°, 90°, 135°$. Each $P(m, n, d, \alpha)$ is the probability of going from gray level m to gray level n given that the intersample spacing is d and the direction is α. If an image has N_g gray levels, then a GLCM can be written as a $N_g \times N_g$ matrix. Each matrix is computed from a digital image by counting the number of times each pair of gray levels occurs at separation d and in direction α. We assume $d = 1$. The matrices are combined by averaging the GLCM for each angle in order to obtain a certain degree of rotation invariance. To deal with irregular areas, only those pairs that fell inside the irregular shape were taken into account, and this number of pairs is also used to normalize the cooccurrence matrix. The two dimensional vector could be used directly as a texture feature vector. However, it is common to use the following derived features from the matrix as defined by Haralick *et al.* [41], [42]: ENE (energy), ENT (entropy), Cor (correlation), IDM (inverse difference moment), INE (inertia), CS (cluster shade), and CP (cluster prominence).

2) Gray Level Run Length Statistics (GLRLS): A set of consecutive pixels in the image having the same gray level value is called gray level run [43], [44]. The length of the run is the number of pixels in the run. So, a run length matrix results, from which the following features are derived: SRE (short run emphasis), LRE (long run emphasis), GLNU (gray level nonuniformity), RLN (run length nonuniformity), and RP (run percentage). Length matrix was also normalized by the number of runs in the shape.

3) Neighboring Gray Level Dependence Statistics (NGLDS): This method considers the relationship between an element and all its neighboring elements at one time. It is based on the calculation of a gray level spatial dependence matrix (NGLDM) of the image [44]. The usual numerical measures derived from NGLDM are: SNE (small number emphasis), LNE (large number emphasis), NNU (number nonuniformity), SM (second moment), and ENT (entropy).

We also consider a texture vector TF (texture features) which is the union of the previous ones.

VII. Classification of Pollen Grains Into Taxonomic Categories

The next issue is how to assign each query case to a preestablished class (in our case, Parietaria Judaica, *Urtica Membranacea*, and *Urtica Urens* species). The classification task has been developed using three standard classifiers. The simplest is the minimum distance classifier (MDC) based on the Mahalanobis distance among patterns and prototypes and combined with several feature selection methods. Since pattern classification is one of the important fields of application for neural networks, we have used the multilayer perceptron (MLP) [47]. Finally, support vector machine (SVM) was also used, because it is the state-of-the-art classifier and it provides the best performance for a wide range of classification problems.

A. Minimum Distance Classifier (MDC) With Feature Selection

The minimum distance classifier is one of the simplest in the literature [45]. It works as follows. Let M be the number of classes, and let N be the number of pollen grains. Let $\mathbf{x}_i = [x_{1i}, \ldots, x_{ni}]$ be the feature vector of n elements that represents grain i. We use the Mahalanobis distance D_j to measure the similarity between a query case and the class prototypes denoted as

$$D_j^2 = (\mathbf{x}_i - \mathbf{m}_j)^\top \Sigma^{-1} (\mathbf{x}_i - \mathbf{m}_j), \qquad j = 1, \ldots, M \tag{7}$$

where Σ is the covariance matrix of the training set, and $\mathbf{m}_j$ is the mean class prototype of class j. The mean class prototype is calculated by taking the mean vector of each class in the training set. We assume the same covariance matrix for all classes.

Some of the features may have meaningless classification capabilities, or do not improve overall system performance due to the correlation among them. In order to improve the classification ratio and the efficiency of the system, an important issue is how to select the optimal features. An exhaustive search of k features out of n available yields a rather high number of possible combinations. So, in practice, other suboptimal searching algorithms are suggested [40]. They fall into three categories: global, scalar, and vector feature selection.

Scalar feature selection is based on measuring the discrimination capabilities of individual features. We first compute a discrimination criterion $C(p)$ for each feature $p = 1, 2, \ldots, n$. Then, the features are ranked in descending order of their values $C(p)$. The features corresponding to the k highest values of $C(p)$ are then selected to form the feature vector. We adopted the Fisher discriminant ratio (FDR) defined as

$$\sum_i \sum_{j \neq i} (\mu_{ip} - \mu_{jp})^2 \,/\, (\sigma_{ip}^2 + \sigma_{jp}^2)$$

of feature p as the criterion $C(p)$. The classification of an observation into a concrete class is simply a matter of calculating the Mahalanobis distance of the observation to each mean class prototype in the training set and assigning the pattern to the class with the smallest distance.

The scalar approach has the advantage of computationally simplicity. However, such approaches do not take into account existing correlations between features. In contrast, feature vector selection approaches measure the capabilities of a feature vector (or subsets of the set of available features). We used the floating search method (FSM) [46] which works as follows.

Given a set of n features, the idea is to search the best subset of k of them for $k = 1, 2, \ldots, l \leq n$ so that a cost criterion C is optimized (for us, C is the sensibility of the system using Mahalanobis distance). Let $X_k = \{x_1, x_2, \ldots, x_k\}$ be the set of the best combination of k features and Y_{n-k} the set of the remaining $n - k$ features. We also keep the lower dimension best subsets $X_2, X_3, \ldots, X_{k-1}$. The algorithm can be summarized in the following steps.

1) *Step I—Inclusion:* Form the best subset of $k+1$ elements, $X_{k+1} = X_k, x_{k+1}$. The feature x_{k+1} is chosen from Y_{n-k}. It is the one that, combined with X_k, results in the best value for C.
2) *Step II—Test and exclusion:* Return to the previously selected lower dimension subsets to check whether the inclusion of this new element improves the criterion C. If it does, the new element replaces one of the previously selected features and the backward search is performed until $C(X_{k+1} - x_r) \leq C(X_k), x_r \in X_{k+1}$. If it does not, go to Step I.

The algorithm is initialized by running the inclusion step twice to form X_2, and ends when k features have been selected. The main drawback of normal FSM is that it can be trapped in cycles.

Global feature selection or transformation can be performed by a principal component analysis (PCA) [45]. The principal components of a set of observation vectors $\mathbf{x}_i$ are the characteristic vectors of the covariance matrix Σ_i constructed from the data set. The characteristic values describe the variances associated with the principal components. The dimensionality of the data set can be reduced by ignoring the principal components with low (or zero) characteristic values. Observation vectors can be approximated from the PCA model using $\mathbf{x_i} \approx P\mathbf{b_i} + \mathbf{m}$, where $\mathbf{x}_i$ is the ith observation vector, $\mathbf{m}$ is the mean observation over a class population, P is the matrix of the most significant principal components, and $\mathbf{b_i}$ is a vector of lower dimensionality of $\mathbf{x_i}$. The weights of the principal components $\mathbf{b_i}$ for an observation can be obtained by $\mathbf{b_i} = P^{-1}(\mathbf{x_i} - \mathbf{m})$. The PCA is applied to produce a population of parametric vectors $\mathbf{b_i}$ of reduced dimension.

As our number of microscopy preparations is still limited, the training set is constructed using $N - 1$ pollen grains and the test is carried out using the excluded one, i.e., we apply the leave-one-grain-out approach. If this pollen grain is correctly classified, a hit is counted. The process is repeated N times, each time excluding a different grain of the dataset.

B. Multilayer Perceptron

Artificial neural networks, and specifically MLPs [47] that are trained using the backpropagation algorithm, are very powerful nonlinear classifiers which often provide better performance

than other classical approaches. MLP is composed by several layers (one input layer, one or several hidden layers, and one output layer) of processing units (neurons). The output of each neuron depends both on its input and weight vectors $\mathbf{w}$. This output is calculated by applying a combination function on the input and weight vectors (usually the dot product) and a nonlinear activation function. Each neuron acts as a linear classifier on its input space, and its decision hyperplane is determined by its weight vector. The outputs of each layer of neurons are the inputs to the following layer. MLP uses several hidden neurons in the same layer, and/or several hidden layers, to build piecewise linear decision surfaces with arbitrary complexity to discriminate among classes. Each neuron of the output layer is associated to a different class: the input pattern is classified by the MLP in the class associated to the neuron with highest output. Before the start of the classification process, the offline training stage calculates the connection weights in such a way that they minimize the total training error (sum of the differences between the obtained and the desired outputs for all the training patterns). Weight calculation is done automatically by using some of the variants of the error backpropagation algorithm, which usually reaches good performance, although it is prone to fall in local minima of the error function and to have low generalization ability (overtraining problems).

C. Support Vector Machine

SVMs [48] have become in the last years very popular tools for example-based learning, both for classification and regression tasks. The good performance of SVMs is due to its strong mathematical foundations, based on the statistical learning theory [49]. The main objective of SVM is to control the generalization ability of its decision function by minimizing the structural risk, which is the sum of the empirical risk (training error) and the confidence interval (principle of structural risk minimization). The confidence interval is a measure of the classifier complexity: higher complexity gives more expresive power and less training error, but also less generalization ability. This interval is increasing with the Vapnik–Chervonenkis dimension of the classifier. Vapnik [50] showed that the upper bound of this dimension for a linear classifier—an SVM is a linear classifier in the hidden space—is decreasing with its margin (distance among the classifier hyperplane and its nearest training patterns, called support vectors) and that minimization of this margin is the key to getting a lower confidence interval and better generalization. Following this principle, overtraining is minimized and generalization ability is maximized in SVM, as opposed to MLP, which just minimizes the empirical risk and may suffer from overtraining problems. Also, SVMs use global methods for solving this optimization problem instead of gradient descent techniques, as opposed to error backpropagation, which only guarantees finding local minima. SVM maps the input space to a highdimensional "hidden" space using a nonlinear application implicitly defined by a kernel function, and it acts as a linear classifier on this hidden space. This mapping raises the probability of linear separability for the training set, as stated by Cover's theorem [51], and it avoids the well-known "curse of dimensionality" problem, because the bounds stated by the statistical learning theory for the SVM do not depend on the dimension of the input space. SVM finds the training patterns with more information about the problem (support vectors), which are the nearest to the SVM hyperplane in the hidden space, and the most difficult to classify. The SVM decision function is based on these vectors. Specifically, the decision function for class k (usually it is necessary to have one SVM for each class, although there are other proposals for multiclass SVM [52]) is given by

$$y_k(\mathbf{x}) = \operatorname{sgn}\left[\sum_{i \in SV} \alpha_i d_i K(\mathbf{x}_i, \mathbf{x}) + b\right] \tag{8}$$

where sgn is the signum function, SV is the set of support vectors $\mathbf{x}_i, d_i$ is the desired output of the kth SVM ($d_i = \pm 1$) for support vector $\mathbf{x}_i$, and K is the kernel. Parameters α_i are the Lagrange multipliers, one for each support vector, $\alpha_i = 0$ for training patterns $\mathbf{x}_i \notin SV$. Parameters α_i and b are calculated in such a way that the sum of the training error (empirical risk) and the hyperplane margin (related to the confidence interval) is minimized. There are several iterative techniques to solve this optimization problem, including sequential minimum optimization (SMO) [53], one of the most efficient and popular ones. SVMs have been applied to many problems (see [54, ch. 8]), and several implementations are available.

VIII. Results

Before addressing the results, we will describe our current database of images.

A. Validation Set

Microscopical preparations of the previously mentioned species of the *Urticaceae* family were made from herbarium leaf and natural material. Plants were picked from various villages of Galicia (northwestern Iberian Peninsula). An optical microscope Nikon SKT connected to a photograph camera Nikon 35S was used to take the analogue photographs. They are digitalized by a general purpose scanner. The final spatial resolution at which the images have been captured is 8196 dots per millimeter, yielding images of 2048×3072 pixels. Nine images of *Parietaria Judaica*, Six images of *Urtica Membranacea*, and Ten images of *Urtica Urens* have been used to evaluate the performance of the system. The numbers of pollen grains per image ranges from Two to 16 pollen grains. The total number of pollen grains of each species are: 98 *Parietaria Judaica*, 100 *Urtica Membranacea*, and 93 *Urtica Urens*.

B. Results and Comparison

We have implemented the pollen detection with the method described in Section IV. An example is shown in Fig. 2. Afterward, the outline of each pollen is extracted as described in Section V. We applied the three approaches to compute the closed outline of each pollen, namely, edge-contour, snake-contour, and convex-hull-contour. An example of these contours for each specie is shown in Fig. 3. Then, shape and texture features as

described in Section VI are calculated, which are finally used to classify each grain (Section VII).

Before applying the pollen detection process using the HT, some problem specific parameters must be fixed. The range for the radius to search is determined from the true size of the pollen grains under study, which are typically between 13 and 19 μm. Taking in mind the spatial resolution at which the images are captured, the radius ranges from 11 to 25 pixels. Fixing the value for the threshold to obtain the local maximum in the Hough space is a more critical decision. In a previous work [17], the detection step was statistically optimized using free response operating characteristic curves (FROC) [55]. These curves plot the true positive rate (percentage of grains correctly detected) as a function of the number of false positives per image. The highest performance was 82% correct detection with almost no false positives per image. The best threshold value was found to be 0.45.

The MLP implementation used was provided by the Torch3 machine learning library [56]. Trials were done with several configurations (1–3 hidden layers and 10–20 hidden neurons for each layer) and various activation functions for hidden and output neurons. The best results were obtained using two hidden layers with 20 neurons in each one and hyperbolic tangent activation functions $\mathrm{out} = \tanh(\sum_j w_j x_j)$ for the hidden neurons, and log-sigmoid $\mathrm{out} = \log(1/(1 + \exp(-\sum_j w_j x_j)))$ for the output neurons (x_j is the j-th input). Hyperbolic tangent activation functions are the most used because they are antisymmetric $(\mathrm{out}(\mathbf{x}) = -\mathrm{out}(-\mathbf{x}))$, which allows faster convergence for the backpropagation algorithm [47, p. 179]. The learning rate was 0.01 and the number of training epochs was 1000 for all the simulations.

For the SVM, we used SVMTorch [57], an implementation well suited for large scale problems which is based on SVM-Light [58]. We tried polynomial, sigmoid, and Gaussian kernels, but the best results were obtained with Gaussian kernels. Parameters to tune were C (regularization parameter in the optimization problem: we used $C = 100$, which is the default SVMTorch value) and σ (deviation of the Gaussian kernel: the best results were for σ in the range 1–4).

The percentage of correct classification with geometrical features of the binary shape (common geometrical feature (CGF) vector) for each recognition method and every classifier are summarized in Table I. The minimum distance classifier (MDC) is combined with three feature selector: principal component analysis (PCA), Fisher discriminat ratio (FDR), and floating search method (FSM). The highest percentage of success (73%) is achieved with the MDC classifier using the feature selector FSM and is independent of the method used to extract the pollen grain.

Before computing statistical moments, an important issue to resolve is to determine the number moments necessary for the discrimination process. The lower order moments give information about the coarse shape of the object, while higher order moments encode information of finer details. As the pollen grains are rather similar, we calculate the moments up to the fifth level. We also studied the influence of different normalization methods for their potential of classification. Hence, the vectors are: central moments (CM) with 36 elements, normalized central moments (NCM) with 33 elements, Hu vector with seven elements, and the union of all moments (AM vector). The moments are computed on S, where S is a binary region, its boundary, and a textured shape. For textured shapes, invariance to rotation in normalized central moments is achieved by taking the average of η_{pq} and $\eta_{qp}, q \neq p$. The normalization reduced the number of elements from 33 to 21. Table II summarizes the results. The FSM feature selector shows superior performance compared to FDR and PCA for every feature vector. Then, for clarity purpose, there are only shown in the tables the results obtained with the MDC when it is combined with the feature selector FSM. The highest percentage of correct classification is 80% and it was achieved calculating the central moments on the texture shape of the convex-hull of the grain using the MDC. Approaches based on active contours to obtain the silhouette of pollen grains seem to be slightly superior to the edge approach. The fact that central moments show higher discrimination capabilities than the other feature vectors is consistent with the believe that the size of the object is an important feature in our discrimination problem.

For texture features, the percentage of correct classification are summarized in Table III for every feature vector (rows) and each classifier and pollen extraction approach (columns).

TABLE I
PERCENTAGE OF CORRECT POLLEN GRAIN CLASSIFICATION FOR DIFFERENT POLLEN EXTRACTION APPROACHES (ROWS) AND CLASSIFIERS, MDC, MLP, AND SVM, USING GEOMETRICAL FEATURES OF A BINARY REGION

		MDC			MLP	SVM
		Feature selector				
Region recognition	Features	PCA	FDR	FSM		
Edge contour	18	–	68	73	34	34
Snake contour	18	–	69	73	62	69
Convex–hull contour	18	66	70	73	59	67

TABLE II
PERCENTAGE OF CORRECT POLLEN GRAIN CLASSIFICATION FOR DIFFERENT SETS OF MOMENTS (ROWS) AND DIFFERENT CLASSIFIERS AND POLLEN EXTRACTION APPROACHES (COLUMNS) USING MOMENTS FEATURES

		Edge–contour			Snake–contour			Convex hull		
Binary shape										
Moments	N	MDC	MLP	SVM	MDC	MLP	SVM	MDC	MLP	SVM
CM	36	69	64	69	75	67	69	73	63	71
NCM	33	48	34	58	48	37	47	46	34	49
Hu	7	41	–	44	44	37	48	45	41	46
AM	76	70	62	69	77	64	72	72	66	70
Textured shape										
CM	36	69	61	66	77	69	72	**80**	72	75
NCM	21	74	72	71	71	71	76	77	73	75
Hu	7	70	65	73	74	72	71	75	73	73
AM	61	75	68	70	77	71	76	79	70	76
Boundary										
CM	36	67	61	63	63	67	70	74	67	73
NCM	33	65	53	65	71	66	71	72	64	69
Hu	7	62	56	68	71	63	68	72	67	71
AM	76	67	61	65	77	67	70	77	65	73

TABLE III
PERCENTAGE OF CORRECT POLLEN GRAIN CLASSIFICATION FOR DIFFERENT SETS OF TEXTURE FEATURES (ROWS) AND DIFFERENT CLASSIFIERS AND POLLEN EXTRACTION APPROACHES (COLUMNS)

		Edge–contour			Snake–contour			Convex hull		
Features	N	MDC	MLP	SVM	MDC	MLP	SVM	MDC	MLP	SVM
FOS	6	71	67	81	79	70	81	81	73	82
HC	7	69	57	77	70	56	76	76	57	76
GLRLS	5	73	67	76	76	75	79	79	74	77
NGLDS	5	63	63	73	72	62	81	73	67	78
TF	23	80	69	84	86	73	85	**88**	70	86

TABLE IV
PERCENTAGE OF CORRECT POLLEN GRAIN CLASSIFICATION FOR DIFFERENT NUMBER OF BOUNDARY POINTS (ROWS) AND DIFFERENT CLASSIFIERS AND POLLEN EXTRACTION APPROACHES (COLUMNS) USING FOURIER DESCRIPTORS

	Edge–contour			Snake–contour			Convex hull		
N of points	MDC	MLP	SVM	MDC	MLP	SVM	MDC	MLP	SVM
32	69	64	66	69	68	70	78	69	74
64	70	65	66	76	74	72	76	68	75
128	70	65	66	76	71	74	**80**	69	76
256	69	62	66	78	72	71	78	67	74

The highest performance is 88%. It is achieved using the union of all texture features (TF vector), the MDC classifier with feature selector FSM, and the pollen extracted with the convex-hull approach.

The maximum number of samples for the Fourier analysis is determined by the spatial resolution of the images. The results are summarized in Table IV for different numbers of points on the pollen boundary, different classifiers, and pollen extraction approaches. The values for Fourier descriptors lie almost all in the range of 70% to 80% for the MDC classifier, independently of the pollen extraction approach. The highest rate of pollen grain classification (80%) is achieved taking 128 points of the contour generated by the convex hull.

IX. DISCUSSION

The features have been divided into the following categories: boundary features, texture features, and binary shape features. Boundary features represent the silhouette of the object and they group Fourier descriptors and boundary moments. Texture features are associated with the intensity level or gray level and include textured shape moments and texture features. At last, binary shape features include binary region and geometrical features representing general information about the shape of the object. The group with the highest results contains texture features (reaching 88%), followed by boundary features with Fourier descriptors (reaching 80%).

In general, the central moments achieve higher results than normalized central moments or Hu moments, which implies that the area is an important parameter in pollen discrimination. Nevertheless, the climatic and growing conditions of the plant can affect the size of pollen grain, which must be taken into account once applying the system to larger data sets and more species. So an important question is whether Fourier descriptors (FD) are able to achieve high rates including normalization according to the area. The tests we performed using 16 descriptors with the snake-contour and with the convex-hull contour showed a decrease of approximately eight points (decreasing up to 70%).

The performance of the Fourier descriptors does not depend much on the number of points taken from the object border. Different samples were probed from 32 points to 256 points, doubling each time the vector and the performance is only slightly different, reaching the hightest rates for 64 or 128 points. They are further evaluated with respect to the number of descriptors (FD). The rate of classification keeps constant for feature vectors containing the 12 of the FDs encoding the lowest frequencies, independently of the classifier used.

Three boundary representations were evaluated: edge-contour, snake-contour, and convex-hull-contour. The last two are superior and show very similar behavior with respect to their classification rates. The idea of the convex hull was to eliminate those points that are strongly attracted to internal features; on the other hand, the convex hull smoothes fine details of the boundary. The worst behavior exhibits edge-contour approach for all classifiers.

For the classification stage, three different classifiers were tested: MDC with feature selectors (PCA, FDR, and FSM), MLP and SVM. In general, for all pollen silhouette extraction approaches and feature vectors computed, the MDC classifier combined with the feature selector FSM is slightly superior than the SVM classifier for almost all cases tested, and the performance of MLP classifier is clearly lower than the others. The good result for the MDC suggests that feature selection may helps to increase the performance of the other sophisticated classifiers (SVM or MLP).

Although the classification rate for different types of information about pollen, such as its boundary, its binary shape, or its texture, is encouraging, system performance may be improved by combining the information. For MDC with feature selector FSM, we tested two strategies: merging individual outputs of simple classifiers, and merging selected features to yield a new feature set [59]. The output of the combined classifier was done using class majority voting between individual ones. Let P the number of simple classifiers combined. The majority voting will assign to a query case the class level vote by at least $(P+1)/2$ individual classifiers. We took the best one for each type of information, i.e., FD, vector of all texture features, and central moments of the binary shape. The result obtained was 89% of correct pollen classification, hardly an improvement compared to using texture features only. For the other strategy, we constructed a vector including the features selected by the FSM algorithm for the best individual classifiers. The result obtained was 86% of correct pollen classification. The largest values of correctness has been reached for the *Urtica Membranacea* with 98%, but the value fell down to 82% for the *Urtica Urens* and to 76% for the *Parietaria Judaica*. On average, the combination of different types of information about the pollen only improves system performance slightly.

In the current implementation, which is not optimized, the treatment of an image take less than 4 min on a general-purpose PC.

X. Conclusion

An automatic system was developed to discriminate species of the *Urticaceae* family (*Parietaria Judaica*, *Urtica Membranacea*, and *Urtica Urens*). The system consists of the following steps: detect the position of pollens based on the application of HT, extract their sillouete applying snakes, calculate shape and texture features from the pollen, and classify it into taxonomic categories. The system achieves 89% correct pollen classification, which is much higher than palynologists can distinguish in routine analysis. The system analyzes both the border and the texture of the pollen, imitating the common way used by palinologists in the routine identification of pollen. Both aspects achieve, separately a similar rate of classification; by merging them, the rate increases.

It is important to remark that some images of the pollen are not taken under optimal conditions. The depth of focus can be a major problem when dealing with three-dimensional objects with an optical microscope. Working with images of different focus planes may improve the performance of the system. Also, a wider test of the system will be needed before applying this method in systematic pollinic counting.

References

[1] E. Stillman and J. Fenley, "The needs and prospects for automation in palynology," *Quart. Sci. Rev.*, no. 15, pp. 1–7, 1996.

[2] I. France, A. Duller, G. Duller, and H. Lamb, "A new approach to automated pollen analysis," *Quart. Sci. Rev.*, vol. 18, pp. 537–536, 2000.

[3] M. Langford, G. Taylor, and J. Flenley, "Computerized identification of pollen grains by texture analysis," *Rev. Palaeobot. Palynol.*, vol. 64, pp. 197–203, 1990.

[4] P. Li and J. Flenley, "Pollen texture identification using neural networks," *Grana*, pp. 59–64, 1999.

[5] M. P. S. Otero, M. Suárez-Cervera, and V. R. Gracia, *Atlas de Polen de Galicia I*, vol. 358. Ed., Deputación de Ourense, 1996.

[6] A. Dopazo, J. Mendez, and C. Seijo, "Aplicación de los estudios morfológicos y biométricos a la caracterización de tipos polínicos," *Primera Reunión de la Red de Investigación Aerobiológica de Galicia*, Actas, Ed., M. J. Aira, V. Jato, and I. Iglesias, Santiago, 2000.

[7] G. D'Amato and F. Spieksma, "European allergic pollen types," *Aerobiologia*, vol. 8, pp. 447–450, 1992.

[8] A. C. Negrini, R. Ariano, G. Delbono, A. Ebbli, A. Quaglia, and D. Arobba, "Incidence of sensitization to the pollens of urticaceae (parietaria), poaceae and oleaceae (olea europea) and pollen rain in liguria (Italy)," *Aerobiologia*, vol. 8, pp. 355–358, 1992.

[9] J. Belmonte and J. Roure, "Characteristics of the aeropollen dynamics at several localities in spain," *Grana*, no. 30, pp. 364–372, 1991.

[10] R. Ayuso, J. Carreira, and F. Polo, "Quantification of the major allergen of several Parietaria pollens by an anti-Par 1 monoclonal antibody-based ELISA. Analysis of crossreac among purified Par J 1, Par o 1 y Par m 1 allergens," *Clinical Experimental Allergy*, vol. 25, no. 10, pp. 888–993, 1985.

[11] I. Liarte and Y. Seras, "Sensibilización al polen de parietaria. causa insospechada de palinosis en nuestro medio," *Allergol. Et Immunologpathol*, vol. 5, no. 21, pp. 202–203, 1993.

[12] A. Corbi, C. Corte, J. Bousquet, A. Basomba, A. C. J. Garcia-Selles, G. D'Amato, and J. Carreira, "Allergenic cross-reactivity pollens of urticaceae," *Int. Arch. Allergy Appl.*, vol. 77, pp. 377–383, 1985.

[13] A. Corbi, Z. Pelaez, E. Errigo, and J. Carreira, "Cross-reactivity between parietaria judaica and parietaria officinalis," *Ann. Allergy*, no. 54, pp. 142–147, 1985.

[14] S.-C. Pei and J.-H. Horng, "Circular arc detection based on Hough transform," *Pattern Recognit. Lett.*, vol. 16, pp. 615–625, 1995.

[15] E. Davies, *Machine Vision: Theory Algorithms, Practicalities*, 2nd ed., New York, San Diego, CA: Academic, 1997.

[16] X. Muñoz, J. Freixenet, X. Cufí, and J. Martí, "Strategies for image segmentation combining region and boundary information," *Pattern Recognit. Lett.*, vol. 24, pp. 375–392, 2003.

[17] M. Rodríguez-Damián, E. Cernadas, A. Formella, and A. González, "Automatic identification and classification of pollen of the Urticaceae family," *Acivs 2003*, 2003.

[18] M. Petrou, *Image Processing*. New York: Wiley, 2000.

[19] G. Ritter and J. Wilson in *HandBook of Computer Vision in Image Algebra*, Boca Raton, FL: CRC, 1996.

[20] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man Cybern.*, vol. 9, no. 1, pp. 62–66, Jan. 1979.

[21] J. Serra, *Image Analysis and Mathematical Morphology*. London, U.K.: Academic, 1982.

[22] P. Theo, *Algorithms for Graphics and Image Processing*. Rockville, MD: Computer Science, 1982.

[23] P. De-Sá, A. P. González, M. Rodríguez-Damián, and E. Cernadas, "Computer-aided identification of allergenic species of the, urticaceae pollen," *Grana p.* (in press), 2004.

[24] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Proc. 1st Int. Conf. Computer Vision*, vol. 2, pp. 259–268, London, U.K., 1987.

[25] D. Williams and M. Shah, "A fast algorithm for active contour and curvature estimation," *Comput. Vis. Graph. Image Proccess.*, vol. 55, no. 1, pp. 14–26, 1992.

[26] J. O'Rouke, *Computational Geometry in C*. Cambridge, U.K.: Cambridge Univ. Press, 1993.

[27] A. N. Jain, P. Duin, and M. Jao, "Statistical pattern recognit. A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 1, pp. 5–37, 2000.

[28] L. F. Costa and R. M. Cesar, *Shape-Analysis and Classification Theory and Practice*. Boca Raton, FL: CRC, 2001.

[29] L. Wang, T. Tan, W. Hu, and H. Ning, "Automatic gait recognition based on statistical shape analysis," *IEEE Trans. Image Process.*, vol. 12, no. 9, pp. 1120–1131, Sep. 2003.

[30] L. Cinque, S. Levialdi, and A. Malizia, "Shape description using cubic polinomial Bezier curves," *Pattern Recognit. Lett.*, vol. 19, pp. 821–828, 1998.

[31] J. Xu, "Efficient morphological shape representation with overlapping disk components," *IEEE Trans. Image Process.*, vol. 10, no. 9, pp. 1346–1356, Sep. 2001.

[32] H. Drolon, F. Druaux, and A. Faure, "Particles shape analysis and classification using the wavelet transform," *Pattern Recognit. Lett.*, vol. 21, pp. 473–482, 2000.

[33] Y. Zhu, L. C. de Silva, and C. C. Ko, "Using moment invariants and HMM in facial expression recognition," *Pattern Recognit. Lett.*, vol. 23, pp. 83–91, 2002.

[34] M. Carlin, "Measuring the performance of shape similarity retrieval methods," *Comput. Vision Image Understanding*, vol. 84, pp. 44–61, 2001.

[35] P. L. Worthington and E. R. Hancock, "Object recognition using shape-from-shading," *IEEE Trans. Pattern Anal. Mach. Intell. 2001*, vol. 23, no. 12, pp. 1250–1267, Dec. 1999.

[36] L. Shen, M. Rangayyan, and L. Desautels, "Application of shape analysis to mammographic calcifications," *IEEE Trans. Med. Imag.*, vol. 13, pp. 263–274, Jun. 1994.

[37] L. Gupta and M. Srinath, "Contour sequence moments for the classification of closed planar shapes," *Pattern Recognit.*, vol. 20, 1987.

[38] M. Hu, "Visual pattern recognition by moments invariant," *IRE Trans. Inf. Theory*, vol. IT-8, no. 2, pp. 179–187, Feb. 1962.

[39] P. García-Sevilla and M. Petrou, "Analysis of irregularly shaped texture regions," *Comput. Vision Image Understanding*, vol. 84, pp. 62–76, 2001.

[40] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*. New York: Academic, 1999.

[41] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Trans. Man Cybern.*, vol. 3, no. 6, pp. 610–621, Nov. 1973.

[42] R. M. Haralick and L. Shapiro, *Computer and Robot Vision*. Reading, MA: Addison-Wesley, 1993.

[43] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis, and Machine Vision*, 2nd ed., International Thomson: Pacific Grove, CA, 1999.

[44] L. H. Siew, R. M. Hodgson, and E. J. Wood, "Texture measures for carpet wear assessment," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 10, no. 1, pp. 92–104, Jan. 1988.

[45] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. New York: Wiley, 2001.

[46] P. Pudil, J. Novovicova, and J. Kittler, "Floating search methods in feature selection," *Pattern Recognit. Lett.*, vol. 15, pp. 1119–1125, 1994.

[47] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed., Englewood Cliffs, NJ: Prentice-Hall, 1999.
[48] C. Cortes and V. Vapnik, "Support vector networks," *Mach. Learning*, vol. 20, pp. 273–297, 1995.
[49] V. Vapnik, *The Nature of Statistical Learning Theory.* Berlin, Germany: Springer, 1995.
[50] ——, *Statistical Learning Theory*. New York: Wiley, 1998.
[51] T. M. Cover, "Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition," *IEEE Trans. Electron. Comput.*, vol. EC-14, pp. 326–334, Jun. 1965.
[52] C. Angulo, X. Parra, and A. Catalá, "K-SVCR: A support vector machine for multi-class classification," *Neurocomputing*, vol. 55, pp. 55–77, 2003.
[53] J. Platt, "Fast training of support vector machines using sequential minimal optimisation," in *Advances in Kernel Methods. Support Vector Learning*, B. Schölkopf, C. J. C. Burges, and A. J. Smola, Eds. Cambridge, MA: MIT Press, 1999, pp. 185–208.
[54] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge, U.K.: Cambridge Univ. Press, 2000.
[55] N. MacMillan and C. Creelman, *Detection Theory: A Users Guide*. Cambridge, U.K.: Cambridge Univ. Press, 1991.
[56] R. Collobert, S. Bengio, and J. Marieťhoz, "Torch: A modular machine learning software library, Tech. Rep. IDIAP-RR 02-46 02-46, 2002," http://www.idiap.ch/˜bengio/ publications/pdf/rr02-46.pdfhttp://www.torch.ch.
[57] R. Collobert and S. Bengio, "SVMTorch: Support vector machines for large-scale regression problems," *J. Mach. Learning Research*, vol. 1, pp. 143–160, 1999.
[58] T. Joachims, "Making large-scale support vector machine learning practical," in *Advances in Kernel Methods*, B. Schölkopf, C. J. C. Burges, and A. J. Smola, Eds. Cambridge, MA: MIT Press, 1999, pp. 169–184.
[59] L. I. Kuncheva, "A theoretical study on six classifier fusion strategies," *IEEE Trans. Pattern Anal. Mach.*, vol. 24, no. 2, pp. 281–286, Feb. 2002.

María Rodríguez-Damián was born in Ourense, Spain, in 1972. She received the B.S. degree in computer science from the University La Salle, Mexico City, Mexico, in 1996.

She has been an Assistant Professor of computer science at the Faculty of Computer Science at the University of Vigo, Spain, since 1998 and is a Ph.D. candidate at the university. Her research focuses on image processing and pattern recognition (mainly in shape analysis).

Eva Cernadas was born in A Coruña, Spain, in 1969. She received the B.S. and Ph.D. degrees in physics from the University of Santiago de Compostela, A Coruña, in 1992 and 1997, respectively.

She is a Lecturer of computer science on the Faculty of Computer Science at the University of Vigo, Ourense, Spain. Her research focuses on image processing and pattern recognition (mainly applied in the food technology and medical domains).

Dr. Cernadas is member of the Spanish society AERFAI.

Arno Formella was born in Saarlouis, Germany, in 1963. He received the B.S. and Ph.D. degrees in computer science from the University of the Saarland, Saarbrücken, Germany, in 1989 and 1993, respectively.

From 1993 to 1999 he was Lecturer and Researcher at the University of the Saarland, at the International Computer Science Institute in Berkeley, and at the University of Applied Science of the Saarland. Since 2000, he has been a Visiting Professor of computer science at the University of Vigo, Ourense, Spain. His research interests included and include computer architecture, computer graphics, applications of computational geometry, and pattern recognition.

Manuel Fernández-Delgado was born in A Coruña, Spain, in 1971. He received the B.S. degree in physics and the Ph.D. degree in computer science from the University of Santiago de Compostela, A Coruña, in 1994 and 1999, respectively.

He is currently an Assistant Professor of computer science at this university. His research interests include pattern recognition, neural computation, and intelligent monitoring of physiological signals.

Pilar De Sá-Otero was born in Ourense, Spain, in 1951. She received the B.S. and Ph.D. degrees (with honors) in biological sciences from the University of Santiago de Compostela, A Coruña, Spain, in 1973 and 1985, respectively.

She taught at the University of Santiago de Compostela between 1974 and 1990. Now she is Professor of botany of the Department of Vegetal Biology and Land Science, University of Vigo, Ourense. Her research focuses on Palinology Applications, in which she has supervised many Ph.D. students, written book chapters, four books, and over 70 scientific papers. She also developed many research projects.

She is a member of the Spanish societies Asociación de Palinólogos de Lengua Española, Real Sociedad Española de Historia Natural, and Asociación de Palinólogos de Lengua Francesa, Société Botanique de France, Asociación Francesa de Estudios de Quaternario.