

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



---

# Image Processing for Pollen Classification

---

Marcos del Pozo-Baños, Jaime R. Ticay-Rivas, Jousé Cabrera-Falcón, Jorge Arroyo, Carlos M. Travieso-González, Luis Sánchez-Chavez, Santiago T. Pérez, Jesús B. Alonso and Melvín Ramírez-Bogantes

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/47527>

---

## 1. Introduction

Palynology - *"The study of pollen grains and other spores, especially as found in archaeological or geological deposits. Pollen extracted from such deposits may be used for radiocarbon dating and for studying past climates and environments by identifying plants then growing."* [1]

Over 20% of all the world's plants are already at the edge of becoming extinct [2]. Saving earth's biodiversity for future generations is an important global task [3] and as many methods as available must be combined to achieve this goal. This involves mapping plants distribution by collecting pollen and identifying them in a laboratory environment.

Pollen grain classification has been an expensive qualitative process, involving observation and discrimination of features by a highly qualified palynologist. It is still the most accurate and effective method. But it certainly limits research progress, taking considerable amounts of time and resources [4].

Automatic recognition of pollen grains can overcome these problems, producing purely objective results faster. Such a tool would provide invaluable in the studies of flora. This advantages were obvious for Flenley [5] [6], who proposed the implementation of an automatic pollen grain classification system in 1968. However, the idea was intractable at that time. Mainly, because of technology restrictions. Nowadays, technology is not a barrier any more, and the discussed system is a reality thanks to computer vision.

This chapter presents the latest results obtained by the authors in the field of automatic pollen grain classification. This will be done by introducing a developed system, paying special attention to the phases of *preprocessing* (section 3.1) and *feature extraction* (section 4). Results for a 17 pollen species database obtained with the commented system will also be shown (section 6).

## 2. Related work

The begins of automatic pollen identification were based on scanning electron microscope (SEM) images. Langford applied statistical classifiers on texture parameters on 1988, reporting a 94.30% of accuracy on a six pollen class database [7]. Later, artificial neural networks (ANN) were used on the classification task, achieving a success rate of 100% with 3 classes [8].

However, SEM images are expensive and difficult to produce and the use of light microscope (LM) images were explored in 1998 [9]. Again, first attempts were not fruitful due to the low quality images provided by the technology of the time. But recent works has demonstrated that the use of LMs images is, in fact, possible.

For example, [10] reported a 100% of success with a small database containing 4 classes. Moreover, it was one of the first works using artificial neural networks for the classification phase, along with texture parameters. Again, [11] used artificial neural networks for classification. This time, brightness and shape descriptors were extracted as pollen features. A 90% of accuracy with a 3 class database was reported.

[12] and [13] presented a more complex work, combining shape and ornamentation of the grains; using simple geometric measures, and concurrence matrices applied for the measurement of texture. Again, artificial neural networks were used for classification. These works reported a 87.7% recognition rate for a 5 classes database and a 97.7% for a three class database respectively.

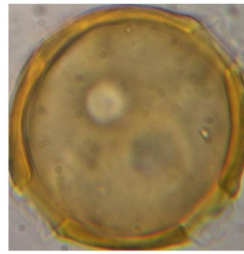
[14] describes an automatic optical recognition and classification of pollen grains system. This is able to locate pollen grains on slides, focus and photograph them before identify the species applying a trained neural network. The system achieved a 90% of recognition rate with a 3 class database.

Other works use more sophisticate capture methods, achieving 3 dimensional representations of the pollen grains. [15] presented a combination of statistical reasoning, feature learning and expertise knowledge. A feature extraction algorithm was applied alternating 2D and 3D representations. Iterative refinement of hypotheses was used during the classification process. This work reported a 77% of accurate rate in a database with 30 classes and 97% when only 4 classes were used. An other example, [16], which used a confocal laser microscope to create the 3D models, achieved a 90% recognition rate with 3 classes database.

## 3. Pollen extraction

At the actual development stage of the system, the detection of pollen grain is highly but not fully automatic. This should not be of any surprise, as the task of pollen location inside sample images is itself a different problem, which is as much complicated as the problem studied in this chapter.

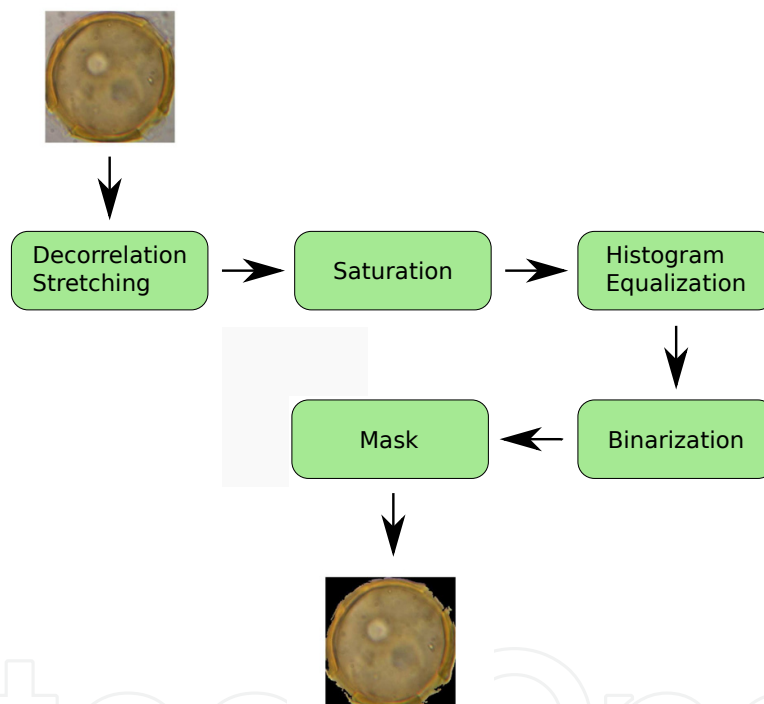
Thus, users should first select and area with a pollen grain inside. Preferably, an area, as small as possible, where an isolated pollen grain is located. This user selected region of interest (ROI) is then automatically preprocessed to detect the contour of the grain (see figure 1).



**Figure 1.** An example of a pollen grain manually selected by the user.

### 3.1. Preprocessing

This section introduces the automatic preprocessing algorithm used for pollen extraction and preparation. It is important to remind that this process is applied to the image area manually selected by the user, like that showed on figure 1. The preprocessing steps are (see figure 2):



**Figure 2.** Automatic preprocessing steps for pollen extraction.

1. *Decorrelation stretching*: This process aims to reduce the autocorrelation of the information contained in the image [17]. This is done as a three steps process:
  - (a) The original bands are transformed to their principal components.
  - (b) The principal components are then stretched separately.
  - (c) The resulting data is transformed back to the original space applying the inverse of the principal component transformation.

The results is a linear transformation of the spectral bands, resulting in uncorrelated variables with unit variance, and enhancing displays. The result can be seen in figure 3.

2. *Saturation*: The saturation channel of the image represents the amount of colour used at each pixel, i.e. the lower the saturation is the greyer the pixel is. This channel is actually extracted from the HSV image representation [18].

In particular, the saturation channel is computed as:

$$S = \begin{cases} 0, & \text{if } MAX = 0 \\ 1 - \frac{MIN}{MAX} & \text{otherwise} \end{cases} \quad (1)$$

The result of computing the saturation channel of the docorrelation stretched image is shown on figure 3. The simplification of the task of differentiating pollen and background is obvious.

3. *Histogram equalization*: Equalizing the histogram of an image aims to obtain a uniform distribution of the pixel values. This maximizes the contrast without losing structural information, i.e., conserving the entropy [19].
4. *Binarization*: The binarization of an image consist of transform each pixel's value to '0' or '1' depending on whether it has a value lower or higher (respectively) than a set threshold. This results on a simple image containing pure geometric information.
5. *Mask*: Finally, in a bid to obtain a clear mask of the pollen grain, several image processing functions are applied such as "imfill" and "bwareaopen" provided by the Image Processing Toolbox of Matlab [20].

The resulting mask can be either used for feature extraction or to remove the background of the pollen grain image. The result of applying each preprocessing step can be seen on figure 3

## 4. Feature extraction

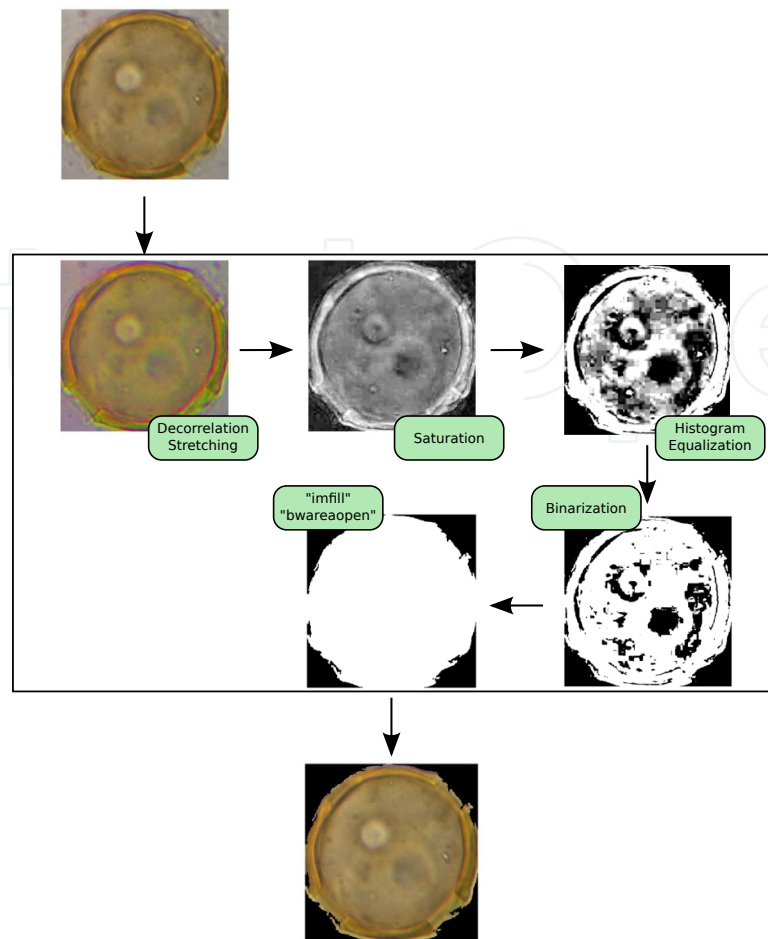
Pollen images by their own does not prove to be a high quality information for the task of automatic pollen grain classification. Although they contain the necessary information, this information is hidden and diffused around the image and behind other unimportant data. In order to extract the relevant information from raw samples, they need to be further processed by the *feature extractor*.

A total of 50 features are extracted from the pollen images. I.e. the output of the *feature extraction* block is a vector with length 50. These 50 features corresponds to 24 geometric parameters carrying information regarding size and basic shape, and 26 texture parameters with information about how pixel intensities are distributed on the image. A detailed view of each of these features will be given here.

Certainly, colour may be an attractive source of information. However, since the preparation of pollen grain samples imply the use of a stain, it is not recommended to use it. Moreover, the stain effects is not constant along time and the colour of the same sample may change.

### 4.1. Geometric parameters

Geometric parameters contain information about the size and the basic shape of the pollen grains. The 24 geometric parameters extracted in the systems presented in this chapter are:



**Figure 3.** The result of applying each preprocessing step to a pollen grain image. Note that the sequence followed is the same as in figure 2.

- *Area*: Refers to the amount of pixels with level '1' in the pollen mask.
- *BoundingBox*: Smallest rectangle enclosing the pollen. In particular, parameters width and height are used as:

$$\begin{aligned} \text{BoundingBox}(1) &= \text{width} \\ \text{BoundingBox}(2) &= \text{height} \end{aligned} \quad (2)$$

- *Centroide*: Refers to the mass centre of the pollen grain. Coordinates  $(x, y)$ .
- *MajorAxisLength*: Length of the major axis of the ellipse with the same second order normalized central moment of the object.
- *MinorAxisLength*: Length of the minor axis of the ellipse with the same second order normalized central moment of the object.
- *ConvexArea*: Area of the smallest convex shape enclosing the object.
- *EquivDiameter*: Diameter of the circle with the same area as the object.

$$EquivDiameter = \sqrt{\frac{4 \times Area}{\pi}} \quad (3)$$

- *Solidity*: Portion of the area of the convex region contained in the pollen.

$$Solidity = \frac{Area}{ConvexArea} \quad (4)$$

- *Perimeter*: Length of the perimeter of the mask image.
- *Extent*: Portion of the area of the bounding box contained in the pollen.

$$Extent = \frac{Area}{Area_{BoundingBox}} \quad (5)$$

- *Eccentricity*: Relation between the distance of the focus of the ellipse and the length of the principal axis.
- *WeightedCentroid*: This is a centroid computing weighted by the pixel values of the grey-scale image.
- *Shape*: Measures how circular is the pollen. Its values are in the range [0,1], where 1 corresponds to a perfect circle.

$$Shape = \frac{4 \times \pi \times Area}{Perimeter^2} \quad (6)$$

- *Thickness*: This is the number of times that the mask has to be eroded with a 3x3 square filter, until it disappears, e.i. the image gets black.
- *Box*: These are the coordinates of an inner rectangle area computed from the *BoundingBox* parameters as:

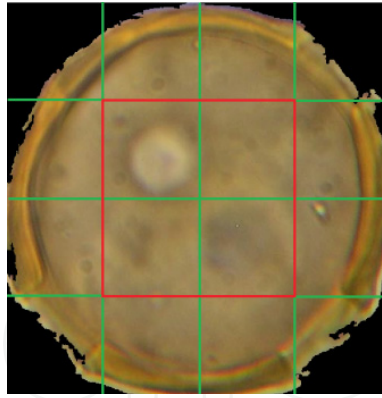
$$\begin{aligned} Box(1) &= \frac{BoundingBox(1)}{4} \\ Box(2) &= \frac{BoundingBox(2)}{4} \\ Box(3) &= \frac{BoundingBox(1)}{2} \\ Box(4) &= \frac{BoundingBox(2)}{2} \end{aligned} \quad (7)$$

- *Hight*: Length of the largest line enclosed in the pollen.
- *Width*: Length of the largest line enclosed in the pollen and perpendicular to *Hight*.

## 4.2. Texture parameters

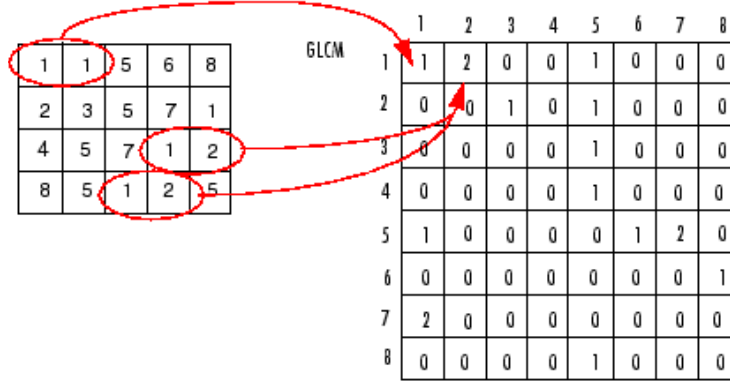
Texture parameters provide information regarding how pixels are distributed on the image, such as contour changes or objects inside the pollen grain.





**Figure 4.** Example of the inner rectangle area computing from the *BoudingBox*.

The first 4 of the 26 texture parameters introduced in this section are computed using the grey level co-occurrence matrix (GLCM). This matrix gives information about the frequency of pixel value pairs combinations. In particular, the value of  $GLCM(i,j)$  is the number of times that a pixel with value 'j' sits next and at the left of a pixel with value 'i'. Figure 5 shows an example of this.



**Figure 5.** Example of a grey level co-occurrence matrix.

- **Contrast:** Mean intensity difference between a pixel and its neighbours. This value is computed as:

$$Contrast = \sum_{i,j} |i - j|^2 p(i, j) \quad (8)$$

- **Correlation:** Measures how much correlated is a pixel with respect to its neighbours. This value is computed as:

$$Correlation = \sum_{i,j} \frac{(i - \mu_i)(j - \mu_j)p(i, j)}{\sigma_i \sigma_j} \quad (9)$$

- **Energy:** Sum of the squared elements of the GLCM. This is:

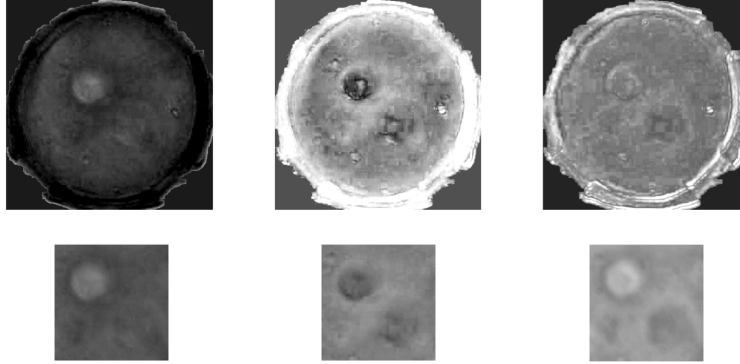
$$Energy = \sum_{i,j} p(i, j)^2 \quad (10)$$



- *Homogeneity*: Measures how close the distribution of objects of the GLCM are to the diagonal of the GLCM. This is:

$$Homogeneity = \sum_{i,j} \frac{p(i,j)}{1 + |i - j|} \quad (11)$$

- *Entropy*: This measure is applied to six different images derived from the original pollen grain image. These images are the the outer and inner bounding box (*BoundingBox* and *Box*) of the blue channel of the RGB representation, the saturation and the value channels of the HSV representation. A representation can be seen in figure 6.



**Figure 6.** Images used to compute the *entropy* measures. They correspond to channels blue, saturation and value (left-right) and outer and inner bounding box (up-down).

*Entropies* are scalar values representing a statistical measure of the randomness of the pixel values. Each value is computed as:

$$Entropy = \sum p \log_2 p, \quad (12)$$

where  $p$  is the histogram count of the corresponding image.

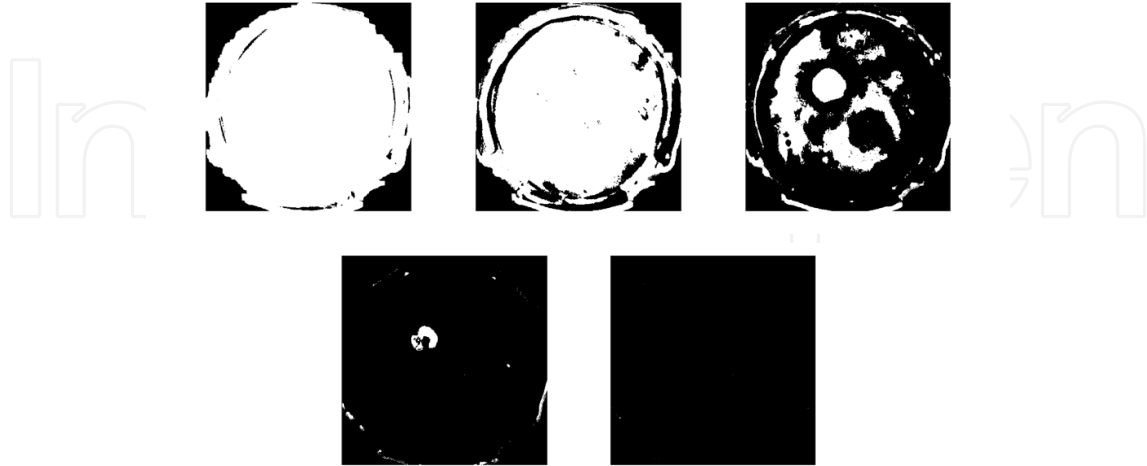
- *Fourier Descriptors*: These measures are based on the analysis of the pollen contour points, and it provides information about the pollen shape. It is worth it to mention that a major property of the *fourier descriptors* is its invariance to geometric transformations, such as rotation, scale and sift.

To compute these parameters, the complex representation of the contour  $z_i = x_i + jy_i$  is used, where  $i = 0, 1, 2, \dots, N_c - 1$  with  $N_c$  the number of points of the contour. Moreover, the contour is sampled every 2 degrees. Now, the discrete Fourier transform (DFT) of  $z$  is:

$$a(u) = \frac{1}{N_c} \sum_{i=0}^{N_c-1} z_i e^{-j2\pi u/N_b} \quad u = 0, 1, 2, \dots, N_b - 1 \quad (13)$$

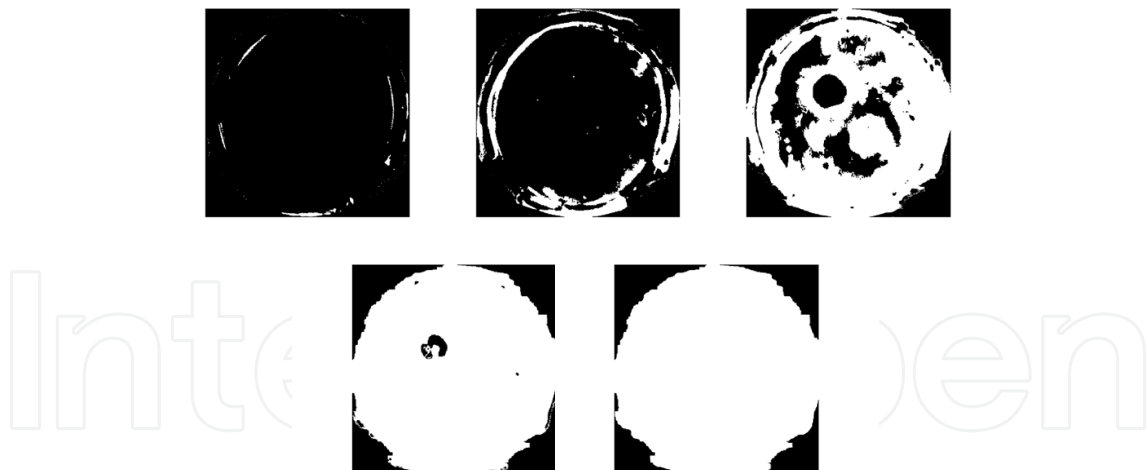
The resultant complex coefficients  $a(u)$  are transformer in a power spectrum  $|a(u)|^2$ . Finally, the discrete cosine transform (DCT) is applied to reduce the dimensionality of the vector, ending up with a vector of length 5.

- *Relative areas*: This is a 5 elements vector which values correspond to the number of active pixels (pixels with value '1') after binarizing the pollen image with different thresholds. In particular, the thresholds used are 0.3, 0.4, 0.5, 0.6 and 0.7. Figure 7 shows an example of this.



**Figure 7.** Results of applying thresholds 0.3, 0.4, 0.5, 0.6 and 0.7 respectively to a pollen image.

- *Relative objects*: In this case the number of objects (group of connected pixels with value '1' and surrounded of pixels with value '0') contained inside the pollen grain are counted, using an inverted and masked version of the binarized images computed the *relative areas*. See figure 8 for an example.



**Figure 8.** Images used to compute the *relative objects*.

## 5. Classification

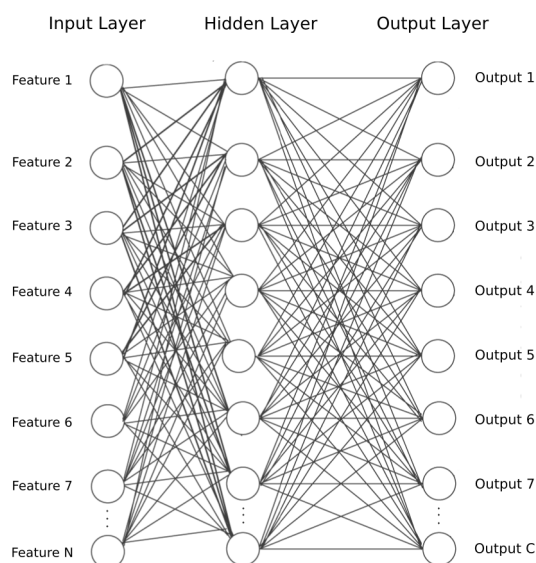
Several works such as [10], [22], [11] and [13] used artificial neural networks (ANNs) as classifiers. These algorithms works as follow:

- Parameters are computed from a set of training samples.

- The computed parameters are passed to the ANN so that it gets trained. This means that the ANN automatically adjusts its parameters to solve the problem of classify the parameters in different classes.
- After the training process, a new testing parameters vector can be passed to the ANN and it will produce an output regarding the sample class.

An ANN is a mathematical model inspired in the structure and functional aspects of the biological neural networks. It could be defined as a set of simple computational elements massively interconnected following a hierarchical organization [21].

In this case, a multilayer perceptron architecture trained by a back propagation algorithm (MLP-BP) is proposed. The principal characteristic of this algorithm is its ability to solve non-linear problems. Its architecture is composed of several layers. Each layer corresponds to a set of neurons receiving data from the previous layer and transmit data to the next layer. This layer can be divided in “input layer”, “hidden layer” and “output layer” as shown in figure 9. In this case, the number of hidden layers is set to one.



**Figure 9.** Architecture of the multilayer perceptron.

It is important note that the training process of the ANN contain an aleatory factor which determines the solution found. In other words, the training process does not avoid local minimums. To overcome this limitation, the proposed classifier implements 11 individual ANNs and sum their resulting scores to obtain a final response. The idea behind this fusion is that the set of computed solutions complement each other, i.e. some solutions correct the errors produced by others.

## 6. Experimentation methodology, results and discussion

A system were implemented in order to test the quality of the proposed approach. This system uses all the techniques introduced in previous sections (preprocessing, feature extraction and

classification). This section gives the details about the database used and the experimental procedure, along with a detailed explanation of the obtained results.

### 6.1. Database

The database used for the experimentation contains 345 images of 17 different pollen grain classes. Images has been captured with a 2 mega-pixels digital camera connected to a microscope set to apply a 40 times zoom.

More precisely, these images correspond to 17 sub-genders and species of 11 different families of tropical honey plants situated in Costa Rica (Central America). Table 1 shows the exact information about family, gender and specie.

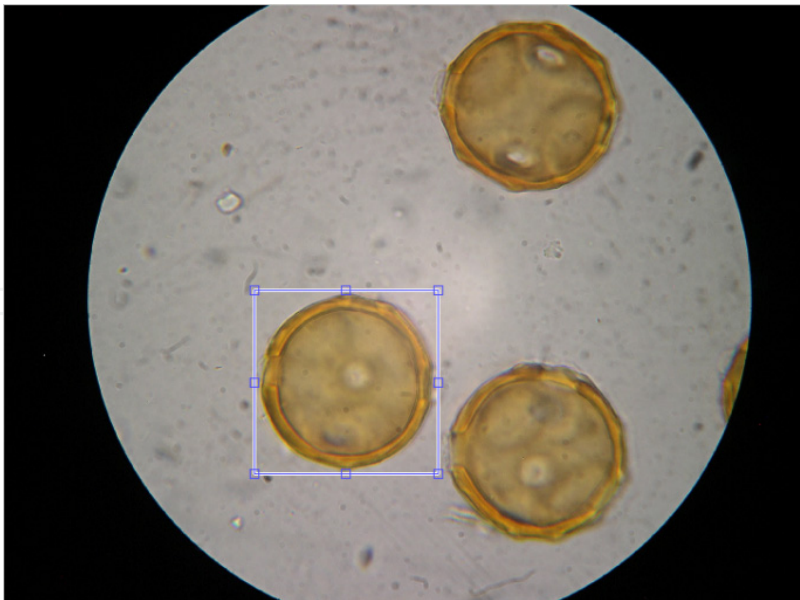
Class	Family	Gender	Specie	Samples
1	Asteraceae	Baltimora	Recta	24
2	Asteraceae	Tridats	Procumbels	47
3	Asteraceae	Critonia	Morifolia	21
4	Asteraceae	Elephantopus	Mollis	17
5	Bombacaceae	Bombacptis	Quinata	18
6	Caesalpinaceae	Cassea	Gradis	35
7	Combretaceae	Combretum	Fruticosum	25
8	Comvulvulaceae	Ipomea	Batatas	15
9	Fabaceae	Aeschynomene	Sensitiva	24
10	Fabaceae	Cassia	Fistula	36
11	Fabaceae	Miroespermyn	Frutesens	18
12	Fabaceae	Enterolobium	Cyclocarpun	18
13	Myrsinaceae	Ardisia	Revoluta	18
14	Malpighiaceae	Bunchosin	Cornifolia	36
15	Saphindaceae	Cardioesperman	Grandiflorus	20
16	Saphindaceae	Melicocca	Bijuga	26
17	Verbenaceae	Lantana	Camara	25

**Table 1.** The exact information about family, gender and specie of the 17 classes included in the DDBB used. The last column expresses the number of samples of pollen grains extracted from the database.



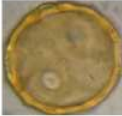
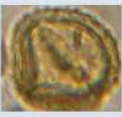



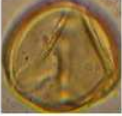

Applying the pollen grain extraction algorithm introduced in section 3, a total of 423 pollen images distributed on all species were obtained. The number of samples extracted for each sample was greater than one. This was possible thanks to images such as that shown in figure 10 where more than one pollen grain could be extracted. Figure 11 shows a sample of each pollen specie included in the DDBB.

### 6.2. Experiments

First, remember from section 5 that the design of the classifier include 30 ANNs fused at the score level. Thus, the number of hidden units on the ANNs had to be specified. To do so, a set of experiments with different configurations were executed to find the optimal value. To obtain a valid measure of the performance of the system, 30 iterations of a hold 50% out cross-validation procedure was executed. Results will be shown and discussed in sections 6.3



**Figure 10.** Database image sample. Note that more than one pollen grain can be extracted from this image.

Class	Sample	Nº	Class	Sample	Nº	Class	Sample	Nº
1		24	7		25	13		18
2		47	8		15	14		36
3		21	9		24	15		20
4		17	10		36	16		26
5		18	11		18	17		25
6		35	12		18			

**Figure 11.** Samples of the 17 different pollen grain species.

and 6.4 respectively. For now, it is enough to note that the optimal value were found with a 30 neurons hidden layer.

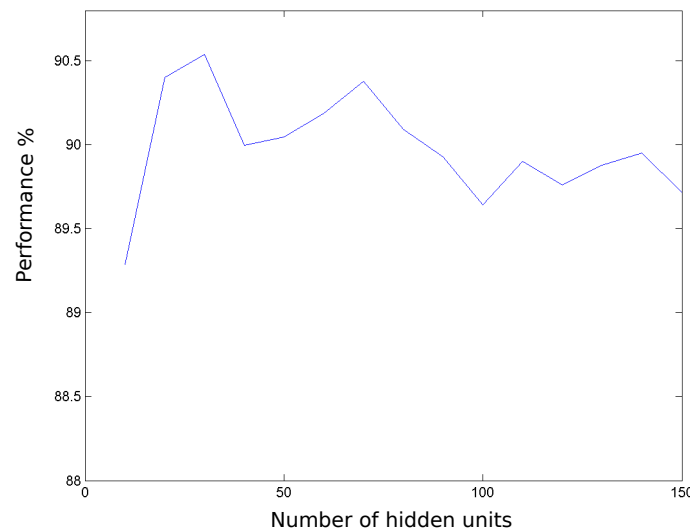
Thus, using this optimal configuration of the ANNs, further experiments were executed to evaluate the performance of the designed system. In this case, 30 iterations of a K-folds cross-validation procedure were applied with values of 'K' equal 3, 5, 7 and 10.

Note that the set of all experimental procedures (hold-50%-out and 3, 5, 7 and 10 folds) are based on divisions of the database in disjoint training and test sets. Moreover, this experiments can be seen as using different proportions of the database of training, i.e. using a different number of samples for training. In particular, the proportions of samples used for training are 1/2, 2/3, 4/5, 6/7 and 9/10 respectively.

### 6.3. Results

It is important to note that every experiment was repeated 30 times in order to obtain a valid measure of the system's performance. Therefore, results are given in terms of mean percentage and standard deviation (mean % and std).

The first experiment tested different configurations of the ANNs. Figure 12 shows the progress of the success rate when the number of units in the hidden layer increased from 10 to 150. A highest rate of 90.54% of success rate were obtained with 30 units (see table 2).



**Figure 12.** Performance progress for different number of units in the hidden layer of the ANN.

A second group of experiments aimed to measure the system's performance with different number of samples for training. Table 3 shows the results obtained for 3, 5, 7 and 10 folds. Note that the success rate increased with the number of training samples (from 90.54% to 92.81%), while the std decreased (from 1.29 to 0.74).

### 6.4. Discussion

It can be argued that the number of hidden units of the ANNs could be further optimized executing a finer search around the point found. However, based on the similar accuracy measures obtained between 10 and 80 units and stds higher than the range of accuracy



Neurons in the hidden unit	Mean % $\pm$ std
10	89.29% $\pm$ 2.11
20	90.40% $\pm$ 1.69
30	<b>90.54% <math>\pm</math> 1.29</b>
40	90.00% $\pm$ 1.66
50	90.05% $\pm$ 1.78
60	90.19% $\pm$ 1.33
70	90.38% $\pm$ 1.52
80	90.09% $\pm$ 1.37
90	89.92% $\pm$ 1.42
100	89.64% $\pm$ 1.49
110	89.90% $\pm$ 1.34
120	89.76% $\pm$ 1.55
130	89.87% $\pm$ 1.37
140	89.95% $\pm$ 1.42
150	89.72% $\pm$ 1.64

**Table 2.** Performance progress for different number of units in the hidden layer of the ANN.

Experiment	Mean % $\pm$ std
Hold-50%-Out	90.54% $\pm$ 1.29
3 <i>k</i> -folds	91.40% $\pm$ 1.05
5 <i>k</i> -folds	92.38% $\pm$ 0.75
7 <i>k</i> -folds	92.43% $\pm$ 0.82
10 <i>k</i> -folds	92.81% $\pm$ 0.74

**Table 3.** Results for 30 iterations of different experiments.

percentages, paying the cost of running a finer search for a minimal increment of performance was not worth it. Therefore, 30 units were chosen as the optimal point.

On the other hand, the results obtained for the second round of experiments show an increasing in both system’s performance and stability. This seems to indicate that the performance of the system may increase with a bigger training database.

### 7. Conclusions

This chapter has introduced the problem of automatic pollen grain classification, which is vital for biologists and flora researches among others. As pointed out in section 3, the task of automatically detecting the pollen grains from samples is a complex problem itself and fall beyond the scope of this chapter. Thus, a semi-automatic algorithm for pollen extraction was explored instead,

The chapter mainly focused its attention in giving a fair amount of both geometric and texture parameters. Moreover, the extraction of this parameters relied on the good work performed by the preprocessing block during the pollen’s perimeter definition.

Finally, these parameters were tested implementing a completed system. In particular, the system used the semi-automatic pollen detection and preprocessing algorithms introduced, along with the mentioned feature extraction techniques and a classifier based on the fusion of



11 ANNs at the score level. The system was tested executing a number of experiments using different hold-out and k-folds cross-validation procedures. The results showed success rates between 90.54% and 92.81%, pointing out the quality of the presented parameters for pollen grain classification. Moreover, these results improve those achieved by other authors such as [10], [22], [11] and [13], even though the number of classified species was significantly larger.

## Acknowledgements

This work has been supported by Spanish Government, in particular by “Agencia Española de Cooperación Internacional para el Desarrollo” under funds from D/027406/09 for 2010, D/033858/10 for 2011, and A1/039531/11 for 2012. To M.Sc. Luis A. Sánchez Chaves, Tropical Bee Research Center (CINAT) at Universidad Nacional de Costa Rica, for provide the database images and palynology knowledge.

## Author details

del Pozo-Baños Marcos, Ticay-Rivas Jaime R., Cabrera-Falcón Jousé, Travieso Carlos M., Pérez Santiago T. and Alonso Jesús B.

*Signal and Communication Department Institute for Technological Development and Innovation in Communications University of Las Palmas de Gran Canaria IdeTIC-ULPGC. Las Palmas de Gran Canaria, Spain*

Arroyo Jorge, Sánchez-Chavez Luis and Ramírez-Bogantes Melvin

*Escuela de Matemáticas, Universidad Nacional, Costa Rica*

## 8. References

- [1] Definition of “palynology” by the Oxford Dictionaries site (<http://oxforddictionaries.com/definition/palynology>). Visited last time on April 2011.
- [2] Plants under pressure: a global assessment. The first report of the IUCN Sampled Red List Index for Plants. Royal Botanic Gardens, Kew, UK (2010)
- [3] Sytnik KM (2010) Preservation of biological diversity: Top-priority tasks of society and state. *Ukrainian Journal of Physical Optics* 11(suppl. 1), S2-S10
- [4] Stillman EC, Flenley JR (1996) The Needs and Prospects for Automation in Palynology. *Quaternary Science Reviews* 15, 1-5.
- [5] Flenley JR (1968) The problem of pollen recognition. In: *Problems in Picture Interpretation*, (ed. M. B. Clowes and J. P. Penny), pp. 141-145. CSIRO, Canberra.
- [6] Flenley JR (1990) Some prospects for palynology in the South- West Pacific Region. *Massey University Faculty of Social Sciences Occasional Papers*, No. 1.
- [7] Langford M, Taylor GE, Flenley JR (1990) Computerized identification of pollen grains by texture analysis. *Review of Palaeobotany and Palynology*, Volume 64, Issues 1-4, 23 October 1990, pp.197-203.
- [8] Treloar WJ (1992) Digital image processing techniques and their application to the automation of palynology. Ph.D. Thes., University of Hull, Hull UK.

- [9] Treloar WJ and Flenley JR (1996) An investigation into the potential of light microscopy for the automatic identification of pollen grains by the analysis of their surface texture. Poster Presentation, 9th International Palynological Congress, Houston TX.
- [10] Li P and Flenley J (1999) Pollen texture identification using neural networks. *Grana* 38(1), pp.59-64.
- [11] Rodriguez-Damian M, Cernadas E, Formella A, Sa-Otero R (2004) Pollen classification using brightness-based and shape-based descriptors. *Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004*, August 23-26, vol.2, pp.23-26.
- [12] Zhang Y, Fountain DW, Hodgson RM, Flenley JR and Gunetileke S (2004) Towards automation of palynology 3: pollen pattern recognition using Gabor transforms and digital moments. *Journals of quaternary science*, vol.19, 2004, pp.763-768.
- [13] Rodriguez-Damian M, Cernadas E, Formella A, Fernandez-Delgado M and De Sa-Otero P (2005) Automatic detection and classification of grains of pollen based on shape and texture. *IEEE Trans. on Systems, Man, and Cybernetics, Part C: Applications and Reviews* 36(4), 531-542.
- [14] Allen GP, Hodgson RM, Marsland SR and Flenley JR (2008) Machine vision for automated optical recognition and classification of pollen grains or other singulated microscopic objects. *Mechatronics and Machine Vision in Practice*, 15th International Conference on , 2008, pp.221-226.
- [15] Boucher A and Thonnat M (2002) Object recognition from 3D blurred images, 16th International Conference on Pattern Recognition, vol.1, pp.800-803.
- [16] Ronneberger, Burkhardt O, Schultz H, General E (2002) Purpose object recognition in 3D volume data sets using gray-scale invariants-classification of airborne pollen-grains recorded with a confocal laser scanning microscope. *Proceedings on Pattern Recognition*, 2002. 16th International Conference, vol.2, pp.290- 295.
- [17] Campbell NA (1996) The decorrelation stretch transformation. *INT. J. Remote Sensing*, vol.17, no.10, pp.1939-1949.
- [18] Agoston MK (2005) *Computer graphics and geometric modeling: implementation and algorithms*. Springer Verlag.
- [19] Baxes GA (1994) *Digital Image Processing: Principles and Applications*. John Wiley & Sons.
- [20] Matlab documentation at Mathworks. <http://www.mathworks.com>. Visited last time on April 2011.
- [21] Hopfield JJ (1982) Neural networks and physical systems with emergent collective computational abilities. *Proc. NatL Acad. Sci. USA Biophysics*, vol.79, pp.2554-2558.
- [22] France I, Duller A, Duller G and Lamb H (2000) A new approach to automated pollen analysis. *Quaternary Science Reviews* 18, pp.536-537.