Discernment of bee pollen loads using computer vision and one-class classification techniques

Manuel Chica [a,b,*], Pascual Campoy [c]

[a] Inspiralia Tecnologías Avanzadas, Estrada 10, 28034 Madrid, Spain
[b] European Centre for Soft Computing, Gonzalo Gutiérrez Quirós, 33600 Mieres, Asturias, Spain
[c] Automatics and Robotics Center – Universidad Politécnica de Madrid, José Gutiérrez Abascal 2, 28006 Madrid, Spain

ABSTRACT

In this paper, we propose a system for authenticating local bee pollen against fraudulent samples using image processing and classification techniques. Our system is based on the colour properties of bee pollen loads and the use of one-class classifiers to reject unknown pollen samples. The latter classification techniques allow us to tackle the major difficulty of the problem, the existence of many possible fraudulent pollen types.

Also presented is a multi-classifier model with an ambiguity discovery process to fuse the output of the one-class classifiers. The method is validated by authenticating Spanish bee pollen types, the overall accuracy of the final system of being 94%. Therefore, the system is able to rapidly reject the non-local pollen samples with inexpensive hardware and without the need to send the product to the laboratory.

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

The bee-keeping sector has a notable socio-economic relevance in Europe, according to the FAO Agricultural Statistics Division. Although honey is the most important bee product, there are other well-known products that result from bee-keeping activity, such as pollen or royal jelly. Bee pollen production, for both domestic and foreign markets, is considered by many bee-keepers a means of diversification and increasing their income. Furthermore, bee pollen products are considered an important food supplement, and can be used in medical treatments, although they are not scientifically recognised.

Bee-keepers, bee-keeping associations, and laboratories are interested in detecting fraud in pollen, and require tools to standardize and authenticate bee pollen origin in order to guarantee its nutritive and health benefits. Microscopic analysis of pollen grains, which form bee pollen loads, is a precise method of identifying origin. However, this process requires the laboratory work of melissopalynology experts, and is thus time consuming and costly. There have been many attempts to automate pollen grain identification by computer algorithms but there is no inexpensive, complete, and automated process (Allen, 2006; Boucher et al., 2002; France et al., 2000; Rodríguez-Damián et al., 2006).

Experts use macroscopic identification of bee pollen loads by means of such properties as colour. This method, although it cannot guarantee complete accuracy, can provide an initial, reliable idea of bee pollen origin (Kirk, 1994). Also, some melissopalynology experts separate pollen load samples by colour as a previous step to final microscopic identification (Campos et al., 1997; de Sá Otero et al., 2002). This process is carried out manually by experts who spend more than an hour separating each sample, an indication of the difficulty and subjectivity involved. Thus, the development of a completely automated, inexpensive system which can recognize external pollen load properties, such as colour, can bring about a twofold improvement in bee pollen origin authentication: (a) recognition of local bee pollen by non-experts (bee-keeping associations, for instance) and (b) reduction of laboratory work by experts through separating the pollen loads automatically.

Developing an automatic system to recognize and separate the pollen loads by colour is a highly complex task, requiring a specific solution. Even within a single pollen type, colour variability is high due to environmental characteristics, such as humidity during the plant growth, as well as the drying process of the pollen, or the presence of impurities. In addition, classification of known local pollen loads must be made against all other world pollen types. This is an important obstacle to the designing of an automated system, since colour data cannot be collected from all existing bee pollen types. In order to overcome these main difficulties, we propose a novel bee pollen load classification system based on image processing and one-class classification. The use of computer vision and

* Corresponding author at: Inspiralia Tecnologías Avanzadas, Estrada 10, 28034 Madrid, Spain. Tel.: +34 985456545.
E-mail address: manuel.chica@softcomputing.es (M. Chica).

classification techniques is not new in the development of food quality control systems and has performed well in many situations (Mery et al., 2010; Kang and Sabarez, 2009; Du and Sun, 2004, 2006).

In our case, the well-known mean shift algorithm (Comaniciu and Meer, 2002) is used to filter and homogenize pollen load colour information. Moreover, one of principal novelties of our proposed method is the use of one-class classification (Moya et al., 1993; Ritter and Gallegos, 1997; Chandola et al., 2009; Tax, 2001), which was introduced as a classification paradigm for detecting anomalies or outliers in data distribution; that is, when there is enough data to model the positive classes but there is limited data on the negative classes. This characteristic is ideal for dealing with our problem, since local pollen types can be modelled, but not all possible fraudulent pollen types. The application field of one-class classification is enormous, from the fraud detection (Taniguchi et al., 1998; Phua et al., 2004) to image processing area (Pokrajac et al., 2007; Augusteijn and Folkert, 2002).

In addition to one-class classification, a multi-classifier algorithm was designed to aggregate one-class classifier outputs, given a unique response with a confidence measure. Unlike existing schemes, our fusion algorithm incorporates an ambiguity discovery mechanism to find pollen types with identical colour properties, in the case of which the system must behave with sufficient robustness.

The proposed method has been validated for the authentication of the most common Spanish pollen types, *Cistus ladanifer*, *Rubus*, *Echium*, and *Quercus ilex*, with respect to non-Spanish pollen types (de Sá Otero et al., 2002). In total, a dataset of around 2000 instances has been used to validate the trained system. In addition, a comparison of one-class classifier approaches has been done with four different algorithms: a Gaussian estimator, a Parzen classifier, a support vector data description (SVDD), and a k-nearest neighbours (kNN) technique. The classifiers were validated using ROC analysis and classification accuracy indicators.

This paper is structured as follows. In Section 2, the bee pollen images and the system used to acquire the images are described. Also, the proposed method, formed by image processing algorithms and one-class and multi classifiers is given in the same section. The developed experiments are presented and analyzed in Section 3. Finally, in Section 4, some concluding remarks and proposals for future work are made.

2. Materials and methods

2.1. Bee pollen problem description

Different samples of Spanish bee pollen loads were obtained from bee-keepers to build the authentication models and validate them against non-local samples. Samples belonging to the four local pollen types (*Rubus*, *Echium*, *Cistus ladanifer*, and *Quercus ilex*) and non-local samples were identified and grouped by experts. Images where these pollen loads samples appear are in Fig. 1. In this figure it can be observed how, even for experts, colour separation is difficult and subjective, and how non-local samples can be misleading (image on the right of the figure).

A computer vision system was used to take images from each of the pollen load samples, as shown in Fig. 2. It is composed of an inexpensive camera, an LED illumination system, and a tray for sample placement. This set of images is the input of the system and the pollen loads appearing in those images must be identified as local or unknown pollen types.

2.2. Outline of the proposed method

An overview of the proposed system can be observed in the diagram of Fig. 3. In this diagram, the initial acquired image of the pollen loads by computer vision hardware as explained in

Section 2.1 can be seen. This image is segmented and processed by the mean shift algorithm (see Section 2.3).

Then, the colour instances of the pollen loads are used to train a multi-classifier model based on one-class classifiers (one for each local pollen type). See Sections 2.4, 2.5, and 2.6 for the methods used in this part of the system.

Finally, the multi-classifier outputs the authentication of each colour instance, classifying them as a known local or non-local (outlier) pollen type.

2.3. Image processing

In the following sections we will describe the segmentation and selection of the colour space (Section 2.3.1) as well as the used image filtering algorithm (Section 2.3.2).

2.3.1. Segmentation algorithm and colour space

First, the well-known Otsu segmentation algorithm (Otsu, 1979) is applied to the gray-scale image to extract the pollen loads from the background. Then, a morphological opening operation is applied to the thresholded binary image (Gonzalez and Woods, 2008). This operation removes those small objects having less than 50 connected pixels in a 8-connected neighbourhood.

Pixels extracted in the latter phase are to be analysed by the remaining processing algorithms. Their colour information can be represented in several ways. The most common is the RGB space where colours are represented by their red, green, and blue components in an orthogonal Cartesian space. However, the RGB space does not lend itself to mimicking the higher level processes which allow human colour perception. Colour is better represented in terms of hue, saturation, and intensity, as HSI or HSV spaces do (Lucchese and Mitra, 2001).

However, the latter colour spaces are not perceptually uniform. The CIE $L^*u^*v^*$ and $L^*a^*b^*$ are ideal for colour recognition because of the following three properties: (a) separation of achromatic information from chromatic information, (b) uniform colour space, and (c) similarity to human visual perception (Wyszecki and Stiles, 1982). In these colour spaces, for instance, the Euclidean distance between two colour points can be easily calculated, as in Eq. (1). This property will ease the work of the classification algorithms

$$D_{12} = \sqrt{(L_2^* - L_1^*)^2 + (u_2^* - u_1^*)^2 + (v_2^* - v_1^*)^2} \qquad (1)$$

Although there are many other possible colour spaces, e.g. CIE *YUV* or CIECAM02, we have used the CIE $L^*u^*v^*$ colour space because of its good results in different colour computerised applications (Chen et al., 2004; Gökmen et al., 2007).

2.3.2. Mean shift filtering

Each extracted pollen load from the image has many different $L^*u^*v^*$ colour values, possibly as many as pixels contained in the load. This has a negative impact on pollen load colour authentication, since human experts customarily identify each pollen load as a unique colour. Therefore, there is a need for a image processing algorithm before applying the classification methods. The goal is to homogenize and smoothen the large quantity of different colour points of a pollen load in just a few unique and representative colour instances.

One of the best methods for discontinuity-preserving smoothening in image processing is the mean shift algorithm, proposed in Comaniciu and Meer (2002), and in line with the feature space analysis. The main strengths of the mean shift algorithm are: (a) it is an application independent tool, (b) it is suitable for real data analysis, (c) it does not assume any prior shape on data clusters, (d) it can handle arbitrary feature spaces, and (e) it has only one parameter, the bandwidth selection window size, to be chosen.
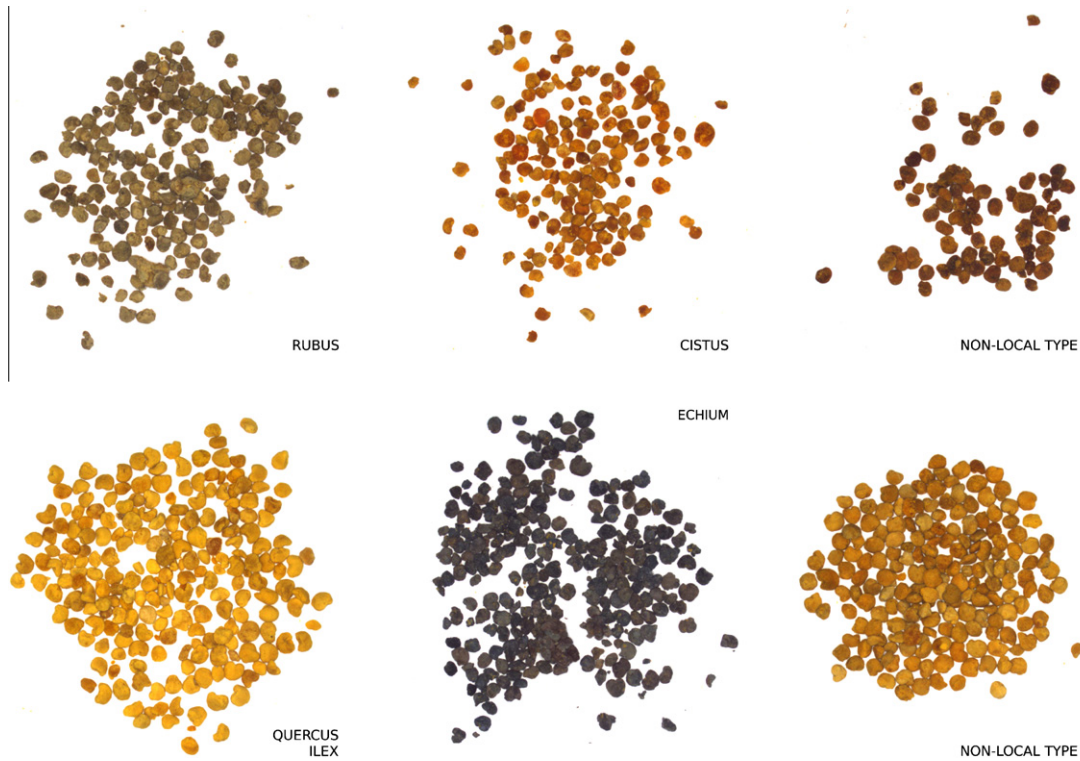
Fig. 1. Different images of pollen load samples which were taken with our vision system. Left and central images belong to known pollen types (*Rubus*, Cistus ladanifer, *Quercus ilex*, and *Echium*, respectively). Right images are non-local samples and must be rejected by the system.

Fig. 2. On the left, some separated pollen load samples are shown. On the right, the computer vision system used to get images is presented.

The mean shift procedure, originally presented by Fukunaga and Hostetler (1975), is a procedure for locating the maxima of a density function given discrete data sampled from that function. It is also an iterative method, starting with an initial estimate or points x_i and a G kernel function. This kernel function determines the weight of nearby points for the re-estimation of the mean. Denote by y_j with $j = 1, 2, \ldots$ the sequence of successive locations of the kernel G:

$$y_{j+1} = \frac{\sum\limits_{i=1}^{n} x_i G\left(\left\|\frac{x - x_i}{h}\right\|\right)^2}{\sum\limits_{i=1}^{n} G\left(\left\|\frac{x - x_i}{h}\right\|\right)^2} \tag{2}$$

is the weighted mean at y_j computed with the kernel G and y_1 is the centre of the initial position of the kernel.

Taking this theory as a base, Comaniciu and Meer (2002) extend the mean shift procedure for filtering and segmenting images. A colour image is typically represented as a two-dimensional lattice

of 3-dimensional vectors (pixels). The space of the lattice is known as the spatial domain, while the colour is the range domain. For both, the mean shift algorithm will work with the Euclidean metric defined in the previous section for the colour space. When both domains are concatenated, the dimensions of the joint spatial-range domain have to be compensated by proper normalization. Thus, the multivariate kernel is defined as the product of two radially symmetric kernels, and the Euclidean metric allows a single bandwidth parameter for each domain:

$$K_{h_s, h_r}(x) = \frac{C}{h_s^2 h_r^p} k\left(\left\|\frac{x^s}{h_s}\right\|^2\right) k\left(\left\|\frac{x^r}{h_r}\right\|^2\right), \tag{3}$$

where x^s is the spatial part, x^r is the range part of a feature vector, $k(x)$ the common profile used in both domains, h_s and h_r the kernel bandwidths, and C the normalization constant. As the normal kernel always provides satisfactory results, the user just has to provide one parameter $h = (h_s, h_r)$, which controls the size of the kernel, and then the smoothening resolution.
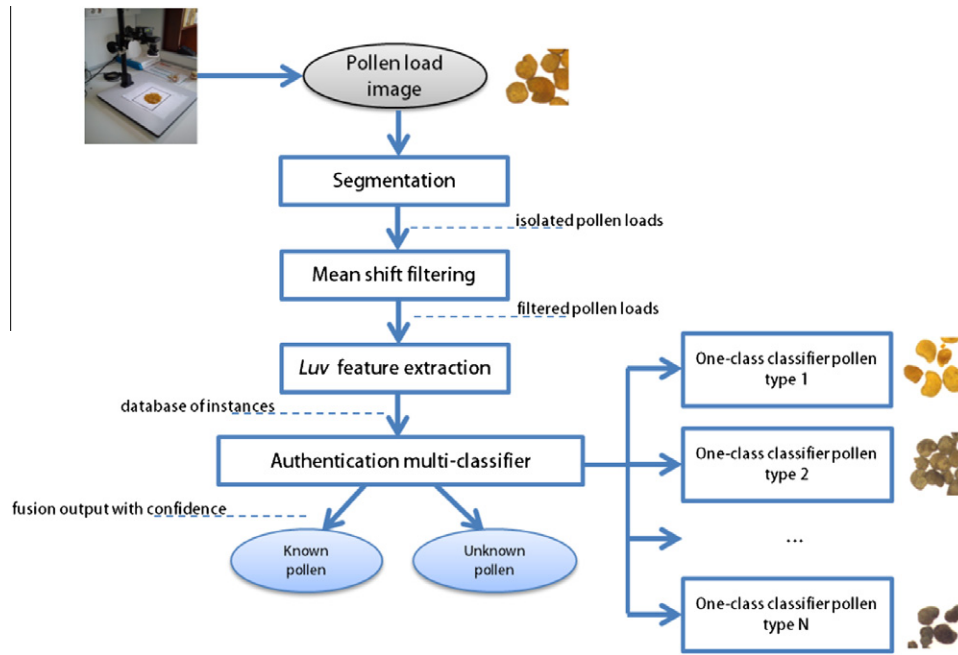
Fig. 3. The diagram of the new discernment methodology, based on computer vision and one-class classification, is shown. The processing chain provides with a final authentication output starting from the image acquisition step.

Nevertheless, replacing the pixel in the centre of the window by the average of the pixels in the window blurs the image. Discontinuity-preserving smoothening techniques, on the other hand, reduce the amount of smoothening near abrupt changes. The mean shift algorithm uses a bilateral filtering which works in the joint spatial-range domain. The data are independently weighted in the two domains and the centre pixel is computed as the weighted average of the window.

The kernel in the mean shift procedure moves toward the maximum increase in joint density gradient, while bilateral filtering uses a fixed static window.

Finally, the mean shift filtering algorithm works as follows. Let x_i and z_i, $i = 1, \ldots, n$ be the $(p + 2)$-dimensional input and filtered image pixels in the joint spatial-range domain. Being the superscripts s and r the spatial and range components of a vector, respectively, and c the point of convergence, for each pixel:

1. Initialize $j = 1$ and $y_{i,1} = x_i$.
2. Compute $y_{i,j+1}$ according to Eq. (2) until convergence, $y = y_{i,c}$.
3. Assign $z_i = (x_i^s, y_{i,c}^r)$.

The spatial bandwidth has a distinct effect on the output when compared to the range (colour) bandwidth. Only features with large spatial support are represented in the filtered image when h_s increases. On the other hand, only features with high colour contrast survive when h_r is large.

2.4. One-class classification

In pollen authentication, there is limited data to model the non-local pollen types (negative classes or outliers). Although it is possible to model the local pollen types, we cannot do the same with all possible existing fraudulent pollen types from around the world. One-class classification has been selected as an appropriate paradigm to deal with this problem.

One-class classification is different from conventional binary or multi-class classification. This distinction lies in the absence of the negative class (normally called outlier) or in the vagueness of its

definition and sampling (Tax, 2001; Chandola et al., 2009). Originally, the term was given by Moya et al. (1993) and some authors refer to this problem as outlier detection (Ritter and Gallegos, 1997), novelty detection (Bishop, 1994), or concept learning (Japkowicz et al., 1995).

This absence or limitation of the negative data makes the problem harder to solve than conventional classification problems. The goal of one-class classification is to define a classification boundary around the positive class (also called target) which maximizes the number of accepted true positive instances and minimizes the number of rejected true negative instances.

The one-class classification techniques used for anomaly or outlier detection can primarily be grouped in two categories: density-based and boundary-based classifiers (Tax, 2001). Within the first group are Gaussian models or Bayesian networks (Siaterlis and Maglaris, 2004, 2001). One of the boundary-based classifiers is the well-known kNN, modified for the case of one-class classification (Byers and Raftery, 1998; Eskin et al., 2002), or support vector machines (Ratsch et al., 2002; Tax and Duin, 2004).

We have used four different approaches of the latter two groups to test them in the system: the Gaussian and Parzen classifiers, which can be considered density-based classifiers, and SVDD and kNN, which are the most representative algorithms of the boundary-based classifiers.

2.4.1. Gaussian model

The training dataset being a set of p-dimensional instances $x_i, i = 1, \ldots, n$, this model simply calculates a Gaussian one-class classifier by estimating the mean $\bar{x}$ and the covariance matrix S_n of the dataset distribution. The classifier uses the Mahalanobis distance $(x - \mu_n)^T S_n^{-1} (x - \mu_n)$ to estimate the fitness of each instance to the target class. In addition, a threshold $\theta_{p,n}$ needs to be defined during the training phase by means of a target acceptance rate which is normally given as a parameter.

Each new testing instance z will be evaluated for acceptance or rejection as target by the following Eq. (4):

$$z \text{ is accepted as target if } (x - \mu_n)^T S_n^{-1} (x - \mu_n) \leqslant \theta_{p,n} \qquad (4)$$

2.4.2. Parzen classifier

The goal of this classifier is to fit a Parzen density estimation on the training dataset x. The Parzen density estimation (Parzen, 1962) is a mixture of Gaussian kernels centred on each training instance x_i with diagonal covariance matrices. The only parameter to be estimated in the model is the width of the kernel h, equal for all kernels. This means that the Parzen density estimator assumes equally weighted features. The optimization of this h parameter is usually performed by maximum likelihood estimation.

2.4.3. Support vector data description (SVDD)

The main idea of SVDD is to obtain a spherical shaped boundary around the training dataset x which can enclose as many samples as possible while having the minimum volume. The sphere is characterised by its centre c and radius $R > 0$. Minimization of the sphere volume is achieved by minimizing its square radius R^2. To improve the generality of the model, some samples are located outside the sphere but larger distances from the centre should be penalized. Thus, slack variables are included in the optimization of the variables which will determine the hypersphere. For the mathematical details, please refer to seminal paper of Tax and Duin (2004).

Given a new sample z, we compare its distance to the centre of the sphere with the radius of the sphere R. If z is inside the hypersphere, it belongs to the target class. Otherwise, it is an outlier.

2.4.4. One-class kNN

Distance-based one-class classifiers are based on the assumption that normal data instances occur in dense neighbourhoods while anomalies occur far from their closest neighbours. kNN, originally provided by Dasarathy (1991), is the best-known distance classifier. The basics of the algorithm for one-class classification is that the anomaly score of a data instance is defined as the distance with its kth nearest neighbour in a given dataset.

Nearest neighbour classifiers always require the definition of distance or similarity measures between two data instances. For continuous features, the Euclidean distance is the most popular choice. For our problem we have chosen $k = 1$ as the parameter of the algorithm; i.e., each new instance z will be considered as target or outlier depending on the classification of its closest neighbour in the training data.

In multi-class anomaly detection, which is the taxonomy of our problem, training data contains only labelled instances belonging to multiple normal classes. A test instance is considered anomalous if it is not classified as normal by any of the classifiers (see plots of Fig. 4). A confidence score with the prediction made by the classifier is normally provided. If none of the classifiers are confident in classifying the test instance, the instance is declared to be anomalous (Chandola et al., 2009). We have followed this kind of approach, as will be described in Section 2.5.

2.5. Fusion of the classifiers

C being a set of known local bee pollen load types, the training data will contain instances belonging to $\|C\|$ classes. In order to use one-class classifiers and be able to reject unknown pollen load types, the system in $\|C\|$ binary sub-problems must be decomposed. Thus, $\|C\|$ one-class classifiers $f_1, f_2, \ldots f_{\|C\|}$ based on densities or distances must be trained, and an ensemble scheme has to be built to fuse them in a multi-class prediction.

Therefore, for each pollen colour instance x we first map each one-class classifier output $f_i(x)$ to a posterior probability $P(y = c|x)$. These probabilities are also normalised in the range [0,1]. The posterior probability of the each classifier's target can be considered as the confidence $CF_{oc}(y = c|x)$ that one instance x belongs to the class c.

In order to classify an incoming pollen load sample as one of the $|C|$ possible pollen types a multi-classifier is constructed. It compares the confidence $CF_{oc}(y|x)$ of all the one-class classifiers and provides a global prediction from the most reliable one-class classifier. The multi-classifier prediction ω is given by:

$$\omega = \max_{1 \leqslant c \leqslant |C|} CF_{oc}(c|x) \tag{5}$$

However, it is also necessary to estimate the confidence of the multi-classifier prediction. To do this we first introduce two parameters as done in Goh et al. (2005):

$$T_{oc} = CF_{oc}(\omega|x) \tag{6}$$
$$T_m = T_{oc} - max_{1 \leqslant c \leqslant |C|, c \neq \omega} CF_{oc}(c|x) \tag{7}$$

Although T_{oc} is the highest confidence factor from the $|C|$ binary one-class classifiers and determines the multi-classifier prediction class ω, T_{oc} might not be sufficient to estimate the global confidence of the multi-classifier prediction. For this reason, we introduce the use of the multi-class margin T_m. Wrong predictions could have high T_{oc} but small T_m but correct predictions must have higher multi-class margin values T_m.

There is a better separation of correct from erroneous predictions if the multi-class margin variable is used (Goh et al., 2001; Schapire and Singer, 1999). After preliminary experimentation, we set both parameters T_{oc} and T_m to 0.5 and 0.01, respectively for all the paper experimentation, to be used in the final decision of the multi-classifier as in the rule of Eq. (8).

$$\begin{cases} \omega \text{ is accepted,} & \text{if } T_{oc} \geqslant 0.5 \quad \text{and} \quad T_m \geqslant 0.001, \\ \text{outlier,} & \text{otherwise} \end{cases} \tag{8}$$

2.6. Ambiguity discovery

Sometimes one or more bee pollen types could have exactly the same colour description as another. In that case, the multi-classification system must be able to detect, during the training phase, that one incoming local pollen type is identical to one already existing. This mechanism is called ambiguity discovery.

The ambiguity discovery process starts if a new pollen type is included in the training process and its corresponding one-class classifier is trained and fused in the multi-classifier scheme, increasing not only the previous error rate, but also the mis-classification with an existing pollen type. Mathematically, ε_{m_1} being a sensitivity–specificity error of a multi-classifier before the inclusion of the new pollen type, and ε_{m_2} the error of a multi-classifier after the inclusion of the classifier of the new pollen type, Δ_ε will be the difference between them. The ambiguity discovery process is launched every time the Δ_ε parameter is higher than a fixed value. In our case we have used the F-measure (van Rijsbergen, 1979) as the ε_{m_1} error measure.

Where Δ_ε exceeds a threshold value, ambiguity discovery is triggered and the process works as follows:

1. The confusion matrix of the new multi-classifier for the testing data is computed.
2. The maximum value of (Λ_i, L_j) is calculated with $i \neq j$, Λ being the vector of real classes and L the vector of predicted classes.
3. The user is consulted about merging the conflicting classes c_i and c_j into a unique class c_i.
4. The multi-classifier is trained according to the response of the user in the third step.

3. Results and discussion

We first show the software prototype used to test and validate the system as well as the results of the image processing algorithms applied (Section 3.1). Then, and using the features
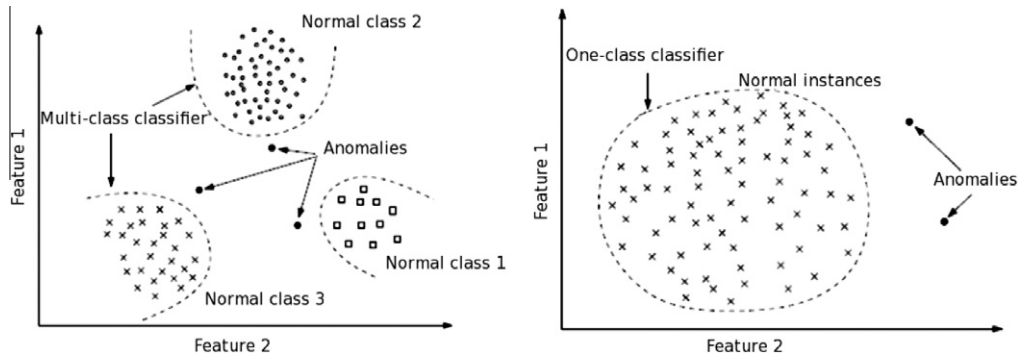
Fig. 4. On the right, a graph showing a one-class classification problem is presented. The left graph shows a multi-class classification based anomaly detection problem (Chandola et al., 2009).

resulting from the latter algorithms, an experiment testing the one-class classifiers and the final multi-classifiers is run. The performance indicators used to compare the classification algorithms are given in Section 3.2. The analysis of the results are commented on Sections 3.3 and 3.4.

3.1. Framework prototype and image processing results

A software prototype was developed to manage the created local pollen types, train and adjust the models, and validate the whole system (see Fig. 5). All the algorithms tested in this section are embedded in this platform. This software was programmed in MATLAB using some functions of the dd_tools library (Tax, 2011).

A set of images in TIFF format and resolution of 1024 x 768 pixels were loaded and processed by the software following the steps described in Section 2.3: pre-processing, segmentation of the image, and mean shift filtering.

As explained in Section 2.3.2, the selection of an appropriate bandwidth parameter for the mean shift algorithm is not trivial. This selection depends on the kind of images used in the application. Basically, the implemented mean shift algorithm receives the spatial bandwidth h_s, the range bandwidth h_r, and the minimum segment area in number of pixels. The last parameter is fixed to a high value (20 pixels) in order not to segment a pollen load into different parts.

The selection of the other two parameters is more complicated. High spatial bandwidths merge different bee pollen loads because they are normally close to each other. Low range bandwidths do not effectively aggregate the entire colour information of the pollen loads. We conducted a preliminary experiment with different values, and the set of bandwidths $h = (h_s, h_r) = (15, 20)$ is the best approach according to this study. Original and filtered pollen load images are shown in Fig. 6.

3.2. Evaluation of the one-class authentication system

From the processed and filtered images, a set of 3146 colour instances, each with three input features (corresponding to the $L^* u^* v^*$ colour space values) and its class (one of the four pollen types or an outlier class), is created. Of this set of colour instances, 400 are used for the training of the four one-class classifiers (100 each pollen type), 800 to test the one-class classifiers (400 belonging to the four known pollen types and 400 were outliers), and the remaining 1946 instances to validate the multi-classification system. The three sets of instances are independent. We have summarised the used data in Table 1.

As remarked in the text, different pollen load types can have the same colour. In these cases, the system's ambiguity discovery process is brought into play, creating a new pollen type with the same colour properties. However, to validate the classifiers, the outlier instances are compounded by pollen types that can have similar, but not identical, colour properties to those of the local pollen types, in which case the ambiguity discovery process is not started.
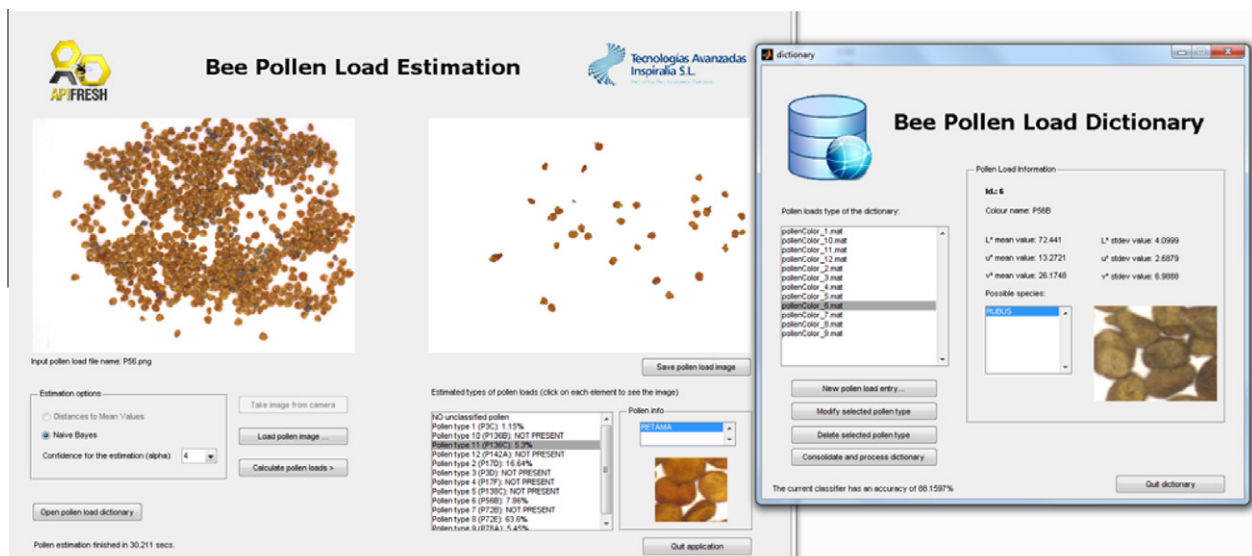

Fig. 5. Screenshot of the software built in MATLAB to test the algorithms. This software could be also used by pollen experts and bee-keepers.

Fig. 6. Result of applying the mean shift algorithm ($h = (h_s, h_r) = (15, 20)$). On the left, the original image is shown. On the right, we present the filtered image.

Table 1
Description of the three independent datasets used to train, test the classifiers, and validate the whole system.

Dataset name	Total size	Pollen types instances	Outlier instances
Training	400	400	0
Test	800	400	400
Validation	1946	1016	930

Table 2
Evaluation measures obtained by the one-class classifiers for each of the four pollen types. Lower values mean better performance.

	Rubus			Echium		
	FN rate	FP rate	F-measure	FN rate	FP rate	F-measure
Gaussian	0.1	0	0.9474	0.19	0	0.8950
Parzen	0.2	0	0.8889	0.27	0	0.8439
SVDD	0.1	0	0.9474	0.1	0	0.9474
kNN	0.05	0	0.9744	0.09	0	0.9529
	Cistus ladanifer			Quercus ilex		
	FN rate	FP rate	F-measure	FN rate	FP rate	F-measure
Gaussian	0.06	0	0.9691	0.06	0	0.9691
Parzen	0.25	0	0.8571	0.06	0	0.9691
SVDD	0.13	0	0.9305	0.09	0	0.9529
kNN	0.06	0	0.9691	0.01	0.0028	0.9851

In order to validate the classifier performance, classification accuracy, false negative and positive rates, F-measure, and confusion matrix are calculated. A false negative (FN) occurs when the outcome of the classifier is incorrectly predicted as outlier when it is actually a target. A false positive (FP), on the other hand, occurs when the outcome is incorrectly predicted as target when it is actually an outlier. The FN rate measures the number of FNs out of the total number of negatives or outliers, and the FP rate calculates the fraction of FPs divided by the total number of positives or target instances (Witten and Frank, 2005). The F-measure provides a relation between the precision and recall of the classification results (van Rijsbergen, 1979).

We have also used the receiver operating characteristic (ROC) curve analysis and the area under the curve (AUC) (Provost and Fawcett, 1997). The ROC curve provides a way to represent the trade-off between false and true positives for different values of the rejection threshold in anomaly detection and one-class classification problems (Bradley, 1997). It does not depend on the specific values of the classification rejection threshold. Also, the AUC summarizes the classification performance of the classifier in the entire range [0,1] of the FP rate, and can be interpreted as the probability of authenticating non-local pollen types (outliers) higher than local pollen types. It is calculated from the ROC curve.

3.3. Analysis of the one-class classifiers

In this section, we have analysed the performance of the four one-class classification paradigms, defined in Section 2.4, in the authentication of each of the selected local bee pollen types. These results were obtained by classifying the test dataset with the trained one-class classifiers. The classifiers were trained without rejecting any training instance as outlier (rejection threshold equal to 0%). In Table 2 the evaluation measures of the classifiers are shown for each pollen type.

By observing the figures of Table 2, we arrive at the following analysis:

- The kNN classifier is the best classifier in comparison with the other models for the authentication of the four pollen types, although for the classification of *Cistus ladanifer* the same results are achieved by the Gaussian classifier.
- The SVDD is the second best classifier according to some metrics. However, it does not clearly outperformed the simple Gaussian classifier. The Parzen model does not reach the results of the other classifiers.
- In general, the FP rate is almost null. This means that the classifiers are able to correctly identify all the outliers (non-local pollen types) without misclassifying them as local pollen types.
- The FN rate is higher than the FP rate. Nevertheless, the FN rate is low, between 5% in the best classifier (the kNN) and 27% in the worst case (the Parzen model).
- The F-measure shows similar results. The kNN classifier is the best except in one problem, the authentication of the *Cistus ladanifer* type.

The ROC analysis will allow us to understand the performance of the classifiers without taking into account the rejection threshold of the targets considered as outliers which biased the measures of the previous comparison. In Fig. 7 and Table 3 the ROC curves and the AUC values are presented. The ROC curves are shown for the most complicated pollen types: *Rubus* and *Cistus ladanifer*. In these cases, the Gaussian classifier (blue solid line) has the worst behaviour while the kNN and SVDD are always in the best position. In the *Rubus* problem, practically the same results are obtained by the kNN, SVDD, and the Parzen classifier (they are represented by means of the same dashed line). The AUC table (Table 3) also presents these results. The Gaussian classifier has the lowest performance and the kNN and SVDD the best. However, the difference between all of them is low.

3.4. Results of the multi-classification system

Different multi-classification models, following the description of Section 2.5, are built using the previous one-class classification paradigms, as their performance indicators were very similar. Then, we compare four multi-classifiers in two different training conditions. The first condition is defined to use one-class classifiers having a 0% threshold rejection. The second one is devoted to use a 10% rejection. The results of this section are calculated by applying the multi-classifiers to the validation dataset, totally independent of the training and test datasets.

In Table 4 the performance measures of the multi-classifiers are shown. In the first block of figures, we have shown the accuracy and FP–FN rates when rejection threshold equals 0%. In the second block, a rejection threshold of 10% is used for obtaining the results.

The kNN, with a rejection of 10%, is the model with the best results as it has low FP and FN rates, and also the highest accuracy. The multi-classifier formed by SVDD one-class classifiers is the second-best model, outperforming the remaining two.

Thus, it seems clear that the performance differences of the one-class models are higher when solving the multi-classification problem, as it is of higher complexity. In this case, more powerful models as kNN and SVDD outperform the rest of the one-class classifiers.

In Tables 5 and 6, the confusion matrices of the best multi-classifiers, SVDD with no rejection and kNN with a rejection of 10%, are shown. It can be seen that generally there is no mis-classification between the known pollen load types, just one instance in which the kNN multi-classifier is misclassified. The highest number of errors is located in the right column. These are FNs as real pollen load

Table 3
AUC measure of the four one-class classifiers. The higher the value, the better the classifier performance.

	AUC			
	Rubus	Echium	Cistus	Quercus
Gaussian	0.4482	1	0.9891	0.9999
Parzen	1	1	0.9936	0.9999
SVDD	1	1	0.9939	0.9999
kNN	1	1	0.9939	0.9999

Table 4
Evaluation measures of the final multi-classifier. Two cases are listed; considering 0%, and considering 10% of data as outliers when training the one-class classifiers. FN and FP rates are calculated taking all the known pollen types as the positive class and the outliers as the negative. Best values are highlighted in bold.

Models of the multi-classifier	0% Rejected during training		
	Accuracy (%)	FN rate	FP rate
Gaussian	92.6002	0.1516	**0.0030**
Parzen	88.0267	0.2441	0.0059
SVDD	**94.3988**	0.0978	0.0177
kNN	92.5488	**0.0108**	0.1161
	10% Rejected during training		
	Accuracy	FN rate	FP rate
Gaussian	88.0267	0.2441	0.0059
Parzen	92.6002	0.1516	**0.0030**
SVDD	93.7821	0.1269	**0.0030**
kNN	**94.6043**	**0.0935**	0.0167

type instances are classified as outliers (non-local pollen types). Nevertheless, as can be seen in Table 4, the FN rate is not high.

In addition to the previous analysis, other remarkable facts can be obtained from the results:

- The trade-off between the FP and FN rates is better than in the isolated one-class classifiers. This fact could be justified because of the confidence mechanism of the multi-classifier, which is able to discard instances classified by one of the one-class classifiers as local pollen types when they are not clearly confident about their decision.
- Although accuracy is obtained when classifying all the classes (four known pollen types and outliers), the FP and FN rates are calculated between the classification of the instance as known pollen type (one of the four known types) or as outlier. There is almost no error among local pollen types.
- If we compare the validation measures of the multi-classifiers and one-class classifiers we can observe how the overall results of the multi-classifiers are better than the isolated one-class classifiers. This indicates good behaviour of the fusion scheme of the known pollen type classifiers.
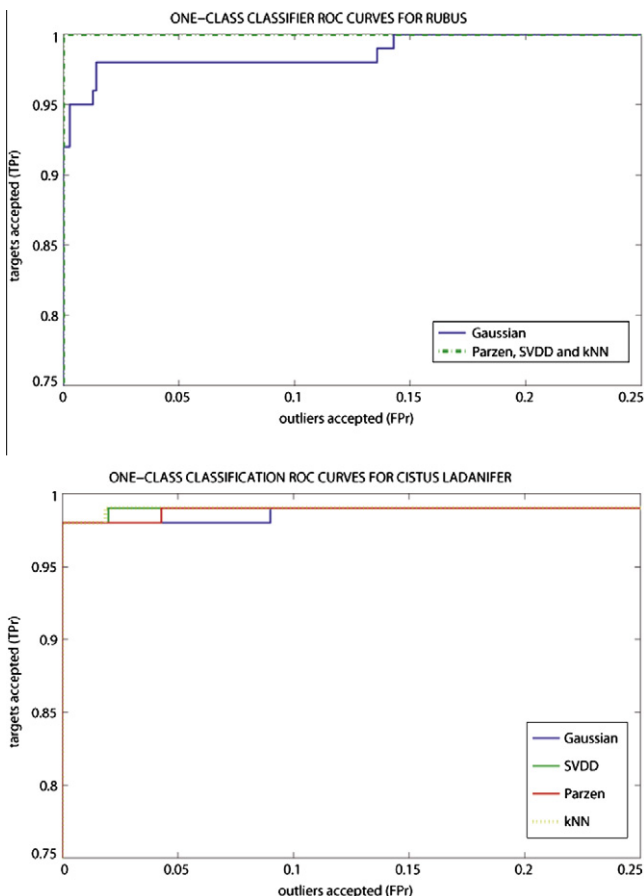

Fig. 7. ROC curves of the one-class classification for *Rubus* and *Cistus ladanifer* types.

Table 5
Confusion matrix of the multi-classification system formed by the four one-class SVDD classifiers. 0% of the training data were considered as outliers during the training phase.

		Predicted pollen type					Total
		Rubus	Echium	Cistus	Quercus	Outlier	
Real pollen type	Rubus	243	0	0	0	38	281
	Echium	0	370	0	0	30	400
	Cistus	0	0	267	0	19	286
	Quercus	0	0	0	45	4	49
	Outlier	2	0	9	7	912	930
	Total	245	370	276	52	1003	1946

Table 6
Confusion matrix of the multi-classification system formed by the four one-class kNN classifiers. 10% of the training data were rejected as outliers during the training phase of the one-class classifier.

		Predicted pollen type					Total
		Rubus	Echium	Cistus	Quercus	Outlier	
Real pollen type	Rubus	248	0	0	0	33	281
	Echium	0	357	0	0	43	400
	Cistus	0	0	275	1	10	286
	Quercus	0	0	0	48	1	49
	Outlier	4	0	5	8	913	930
	Total	252	357	280	57	1000	1946

- One of the distance-based multi-classifiers, kNN, achieves better results when discarding the 10% of the target instances. The remaining methods perform in different ways when rejecting subsets of target instances.

4. Conclusion

A complete novel chain of methods based on computer vision and classification techniques has been applied for the origin authentication of bee pollen loads. Colour properties have been filtered by image processing algorithms, and different one-class classification models have been successfully used. The kNN one-class classification model obtained the best results in comparison with density classifiers and SVDD.

In addition, a multi-classifier scheme has also been designed, improving the accuracy of the isolated one-class classification methods and merging the outputs of the classifiers. The best multi-classifier, formed by kNN one-class classifiers and rejecting some instances in the training phase, has achieved 94% accuracy. These models have been compared in the authentication of four Spanish pollen types against different outlier samples, and validated in 1946 colours instances. An ambiguity discovery mechanism was also presented to include robustness in the final system.

The results are very promising, and the use of the application prototype with the developed methods drastically reduce the time and effort spent by experts to several seconds. It also can be used by non-experts to detect fraudulent products. Future work will be devoted to: (a) studying the benefits of using another macroscopic features of pollen loads, (b) developing a low-cost system to analyse the microscopic features of the pollen grains to be used in conjunction with the system presented in this work, and (c) considering other colour spaces for the proposed methodology.

Acknowledgement

References

Allen, G., 2006. An automated pollen recognition system. Master's thesis. Institute of Information Sciences and Technology, Massey University.

Augusteijn, M., Folkert, B., 2002. Neural network classification and novelty detection. International Journal on Remote Sensing 23, 2891–2902.

Barbara, D., Couto, J., Jajodia, S., Wu, N., 2001. Detecting novel network intrusions using Bayes estimators. In: Proceedings of the First SIAM International Conference on Data Mining, Chicago, USA.

Bishop, C., 1994. Novelty detection and neural network validation. In: IEEE Proceedings on Vision, Image and Signal Processing, Special Issue on Applications of Neural Networks, pp. 217–222.

Boucher, A., Hidalgo, P.J., Thonnat, M., Belmonte, J., Galan, C., Bonton, P., Tomczak, R., 2002. Development of a semi-automatic system for pollen recognition. Aerobiologia 18, 195–201.

Bradley, A.P., 1997. The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern Recognition 30, 1145–1159.

Byers, S., Raftery, A., 1998. Nearest neighbor clutter removal for estimating features in spatial point processes. Journal of the American Statistical Association 93, 577–584.

Campos, M., Markham, K., Mitchell, K., da Cunha, A., 1997. An approach to the characterization of bee pollens via their flavonoid/phenolic profiles. Phytochemical Analysis 8, 181–185.

Chandola, V., Banerjee, A., Kumar, V., 2009. Anomaly detection: a survey. ACM Computing Surveys 41, 15:1–15:58.

Chen, H., Chien, W., Wang, S., 2004. Contrast-based color image segmentation. IEEE Signal Processing Letters 11, 641–644.

Comaniciu, D., Meer, P., 2002. Mean shift: a robust approach toward feature space analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence 24, 603–619.

Dasarathy, B.V., 1991. Nearest Neighbor (NN) Norms: Nearest Neighbor Pattern Classification Techniques. IEEE Computational Society, Los Alamitos, CA.

Du, C., Sun, D., 2004. Recent developments in the applications of image processing techniques for food quality evaluation. Trends in Food Science and Technology 15, 230–249.

Du, C., Sun, D., 2006. Learning techniques used in computer vision for food quality evaluation: a review. Journal of Food Engineering 72, 39–55.

Eskin, E., Arnold, A., Prerau, M., Portnoy, L., Stolfo, S., 2002. A geometric framework for unsupervised anomaly detection: detecting intrusions in unlabeled data. In: Applications of Data Mining in Computer Security. Kluwer Academics, pp. 78–100.

France, I., Duller, A.W.G., Duller, G.A.T., Lamb, H.F., 2000. A new approach to automated pollen analysis. Quaternary Science Reviews 19, 537–546.

Fukunaga, K., Hostetler, L., 1975. The estimation of the gradient of a density function, with applications in pattern recognition. IEEE Transactions on Information Theory 21, 32–40.

Goh, K., Chang, E., Cheng, K., 2001. SVM binary classifier ensembles for image classification. In: Proceedings of the ACM Conference Information and Knowledge Management, New York, USA, pp. 395–402.

Goh, K., Chang, E., Li, B., 2005. Using one-class and two-class SVMs for multiclass image annotation. IEEE Transactions on Knowledge and Data Engineering 17, 1333–1346.

Gökmen, V., Şenyuva, H., Dülek, B., Çetin, A., 2007. Computer vision-based image analysis for the estimation of acrylamide concentrations of potato chips and french fries. Food Chemistry 101, 791–798.

Gonzalez, R.C., Woods, R.E., 2008. Digital Image Processing, third ed. Prentice Hall.

Japkowicz, N., Myers, C., Gluck, M., 1995. A novelty detection approach to classification. In: Proceedings of International Joint Conference on Artificial Intelligence. Morgan Kaufmann, San Francisco, USA, pp. 518–523.

Kang, S., Sabarez, H., 2009. Simple colour image segmentation of bicolour food products for quality measurement. Journal of Food Engineering 94, 21–25.

Kirk, W., 1994. A Colour Guide to Pollen Loads of the Honey Bee, International Bee Research Association, New York, USA.

Lucchese, L., Mitra, S., 2001. Colour image segmentation: a state-of-the-art survey. In: Proceedings of the Indian National Science Academy (INSA-A), New Delhi, India, pp. 207–221.

Mery, D., Chanona-Pérez, J., Soto, A., Aguilera, J., Cipriano, A., Veléz-Rivera, N., Arzate-Vázquez, I., Gutiérrez-López, G., 2010. Quality classification of corn tortillas using computer vision. Journal of Food Engineering 101, 357–364.

Moya, M., Koch, M., Hostetler, L., 1993. One-class classifier networks for target recognition applications. In: Proceedings on World Congress on Neural Networks, International Neural Network Society, Portland, USA, pp. 797–801.

Otsu, N., 1979. A threshold selection method from grey-level histograms. IEEE Transactions on Systems, Man and Cybernetics 9, 62–66.

Parzen, E., 1962. On the estimation of a probability density function and mode. Annals of Mathematical Statistics 33, 1065–1076.

Phua, C., Alahakoon, D., Lee, V., 2004. Minority report in fraud detection: classification of skewed data. In: SIGKDD Explorer Newsletter, pp. 50–59.

Pokrajac, D., Lazarevic, A., Latecki, L., 2007. Incremental local outlier detection for data streams. In: Proceedings of IEEE Symposium on Computational Intelligence and Data Mining, pp. 504–515.

Provost, F., Fawcett, T., 1997. Analysis and visualization of classifier performance: comparison under imprecise class and cost distributions. In: Proceedings of the Third International Conference on Knowledge Discovery and Data Mining, Huntington Beach, CA, USA, pp. 43–48.

Ratsch, G., Mika, S., Scholkopf, B., Muller, K., 2002. Constructing boosting algorithms from SVMs: an application to one-class classification. IEEE Transactions on Pattern Analysis and Machine Intelligence 24, 1184–1199.

Ritter, G., Gallegos, M., 1997. Outliers in statistical pattern recognition and an application to automatic chromosome classification. Pattern Recognition Letters 18, 525–539.

Rodríguez-Damián, M., Cernadas, E., Formella, A., Fernández-Delgado, M., Sá-Otero, P.D., 2006. Automatic detection and classification of grains of pollen based on shape and texture. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews 36, 531–542.

M.P. de Sá Otero, E., Díaz Losada, S.B.S.B., 2002. Método de determinación del origen geográfico del polen apícola comercial, Lazaroa 23, 25–34.

Schapire, R., Singer, Y., 1999. Improved boosting algorithms using confidence-rated predictions. Machine Learning 37, 297–336.

Siaterlis, C., Maglaris, B., 2004. Towards multisensor data fusion for dos detection. In: Proceedings of the 2004 ACM symposium on Applied Computing. ACM Press, Nicosia, Cyprus, pp. 439–446.

Taniguchi, M., Haft, M., Hollmn, J., Tresp, V., 1998. Fraud detection in communications networks using neural and probabilistic methods. In: Proceedings of IEEE International Conference in Acoustics. Speech and Signal Processing. IEEE Computer Society, pp. 1241–1244.

Tax, D., 2001. One-class classification; concept-learning in the absence of counter-examples, Ph.D. thesis. Delft University of Technology.

Tax, D., 2011. DDtools, the data description toolbox for Matlab. Version 1.9.0.

Tax, D., Duin, R., 2004. Support vector data description. Machine Learning 54, 45–66.

van Rijsbergen, C., 1979. Information Retrieval. Butterworth, London.

Witten, I.H., Frank, E., 2005. Data Mining: Practical Machine Learning Tools and Techniques, second ed. Morgan Kaufman, San Francisco, USA.

Wyszecki, G., Stiles, W., 1982. Color Science: Concepts and Methods, Quantitative Data and Formulae. Wiley.