A reference process for automating bee species identification based on wing images and digital image processing

Fabiana S. Santana a,b, Anna H. Reali Costa b, Flavio S. Truzzi b, Felipe L. Silva b, Sheila L. Santos a, Tiago M. Francoy c, Antonio M. Saraiva b

a Centro de Matemática, Computação e Cognição, Universidade Federal do ABC, Avenida dos Estados, 5001, Santo André, SP CEP: 09210-580, Brazil
b Escola Politécnica da Universidade de São Paulo, Av. Prof. Luciano Gualberto, travessa 3, 158, São Paulo, SP CEP: 05508-970, Brazil
c Escola de Artes, Ciências e Humanidades, Universidade de São Paulo, Rua Arlindo Béttio, 1000, São Paulo, SP CEP: 03828-000, Brazil

A R T I C L E   I N F O

Article history:
Received 1 March 2013
Received in revised form 29 November 2013
Accepted 1 December 2013
Available online xxxx

Keywords:
Reference process
Digital image processing
Machine learning
Bee wing classification
Taxonomic classification

A B S T R A C T

Pollinators play a key role in biodiversity conservation, since they provide vital services to both natural ecosystems and agriculture. In particular, bees are excellent pollinators; therefore, their mapping, classification, and preservation help to promote biodiversity conservation. However, these tasks are difficult and time consuming since there is a lack of classification keys, sampling efforts and trained taxonomists. The development of tools for automating and assisting the identification of bee species represents an important contribution to biodiversity conservation. Several studies have shown that features extracted from patterns of bee wings are good discriminatory elements to differentiate among species, and some have devoted efforts to automate this process. However, the automated identification of bee species is a particularly hard problem, because (i) individuals of a given species may vary hugely in morphology, and (ii) closely related species may be extremely similar to one another. This paper proposes a reference process for bee classification based on wing images to provide a complete understanding of the problem from the experts’ point of view, and a foundation to software systems development and integration using Internet services. The results can be extended to other species identification and taxonomic classification, as long as similar criteria are applicable. The reference process may also be helpful for beginners in this research field, as they can use the process and the experiments presented here as a guide to this complex activity.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

The well-being of human populations of the world depends, among others, on the so called ecosystem services. These services can be defined as the benefits people obtain from ecosystems and they can be divided into four categories: supporting systems, such as nutrient cycling and seed dispersal; provisioning services, such as food, water, minerals and others; regulating services, such as pollination and carbon sequestration; and cultural services, such as ecotourism and recreational experiences. That is why the worldwide loss of biodiversity has been a major concern at least since the last century, and it was recognized as such by the United Nations (Brundtland, 1987). It brought to attention that population, energy, industry, food security, and human settlements, are connected to the loss of species and genetic resources. Different factors may cause biodiversity depletion, such as destruction and reduction of natural habitats, pollution of ground water and of the atmosphere, and global warming (Sala et al., 2000).

Pollination is a key ecosystem service that represents a very important tool for environment conservation (Daily, 1997). A serious decline of the pollinator populations is being noticed since the end of the 20th century (Buchmann et al., 1997). It has recently been suggested that the main causes to this decline are linked to the rapid expansion of human activities, such as loss and fragmentation of natural habitat, aggressive agricultural practices, pathogens and climate changes (Potts et al., 2010; Schweiger et al., 2010). In particular, bees are major pollinators; therefore, their preservation helps to promote biodiversity conservation.

Identification of bee species is part of their preservation. It may be obtained by studying, processing, and analyzing images of bee wings (Bueno et al.; Giannini et al., 2011). Several studies have shown that features extracted from patterns of wing venation are good discriminatory elements to differentiate among species of insects (Francoy et al., 2008, 2009; Nielsen et al., 1999; Schröder et al., 2002; Weeks et al., 1997). The use of geometric morphometrics is very helpful, since it applies a set of computational techniques for analyzing shapes with applications in species identification, in genetic variability, among others.
However, automated species identification is a particularly hard problem and the entire process related to image acquisition, plotting of landmarks, feature extraction and statistical analysis is still mostly manual, demanding a considerable amount of time. Therefore, the development of auxiliary tools for bee identification is of great importance. They can reduce the work load for the small number of taxonomists that spend a long time identifying species for non-taxonomist scientists.

Morphometrics, DNA barcode and cuticular hydrocarbons are important alternatives to assess biodiversity, enriching the information obtained in alpha taxonomy (Francisco et al., 2008).

This paper presents a reference process for automating bee identification based on their wing images. Reference processes provide a complete understanding of the problem from the experts’ point of view (Bass et al., 2003). In Internet-based software systems, as is the case with most biodiversity databases and tools, system development and integration are greatly improved by using SaaS, Software as a Service, and Cloud Computing (Fox and Patterson, 2012). Reference processes represent an adequate approach for the development of systems in this scenario because the steps and the communication among different steps are clearly identified.

The reference process is divided into four main sub-processes, namely image acquisition, digital image processing, classification, and validation. Image acquisition is the step in which wing images are obtained and stored. Live bees or dissected bee wings may be used in this step. Digital image processing is the second step, its purpose is to recognize landmarks and to extract relevant and discriminatory features so that bees can be classified. In the classification step, classifiers have the extracted features as input and they decide which species each wing image belongs to. The basic criteria for initial classification are defined by experts in the field. Though they do not always agree about which criteria should be applied in each case, an approach must be chosen in order to perform the classification sub-process properly. The last step is validation; it comprehends the interactions with users, researchers and experts to decide whether the criteria for classification should be revised, or classification results are acceptable, or if further analysis should be performed.

Morphological analyses are performed in the process. They help with bee identification by defining and extracting homologous landmarks and relevant and discriminatory features (such as shape, color and patterns) from the wing images, so that bees can be classified. Landmarks are extracted from Cartesian coordinates plotted at intersections between veins on the bee wing.

The reference process is presented in BPMN, Business Process Modeling Notation (White and Miers, 2008). BPMN is a standard to simplify the understanding of businesses. It provides a graphical notation to facilitate performance and collaboration studies, allowing the identification of gaps and potential improvements.

BPMN permits bridging the gap between business process design and implementation. BPMN is an agreement between multiple modeling tool vendors to use the same notation for the benefit of end-users, and it is supported by many of the largest software companies in the world (BPMN Information Home, 2013). It is also applied by several companies to model their businesses, being one of the most accepted notations by the community (Intalio Business Process Management System Home, 2013). The education community and the aerospace and defense, business service, and construction and operations industries are among their main users.

A BPMN-based process can also be automatically converted into BPEL, Business Process Execution Language. This feature is very desirable for integrating software services on the Internet because it addresses the integration of REST-compliant web services and other web services compliant to W3C standards, which are the majority of services available on SaaS-based systems in the cloud. In order to properly apply BPMN, in this paper bee classification is considered as the business.

The application of reference processes to this problem certainly contributes to its clarification while the knowledge gained with this work allows improvements in the development of software methods and packages for bee identification and classification. The main steps of the reference process were implemented in order to assess its viability and the viability of digital image processing to this problem. The results are presented as case studies, showing its effectiveness. If similar computational techniques can be applied, the approach presented here may be extended to the identification and taxonomic classification of other species, as long as their own characteristics are well defined, extending the contributions of this work. Finally, as it is presented, the reference process may be very helpful for beginners in this research field, as they can use the process and the experiments presented here as a guide to this complex activity.

2. Reference processes and BPMN

The main purpose of this paper is to establish the foundations to construct software systems for automated bee identification based on wing images and digital image processing. A recommendation to achieve this challenge is to design reference architecture for this domain (Bass et al., 2003). A reference architecture can be defined as a set of engineering and design principles within a specific domain. It defines the structure of the system, the responsibilities of system components, and templates, among others, mapping the main functionalities that must be addressed by the software system. Reference architecture offers a basis for developing specific software architectures; because the main issues of the domain are already identified and dealt with, the architect only needs to invest time and effort in the adequacy of the reference architecture for each specific implementation.

Mapping the detailed information contained in an existing standard or model is necessary to define a reference architecture (Bass et al., 2003). This can be obtained by a reference process. Reference processes document and formalize current practices and the experts’ knowledge about a subject (Bass et al., 2003), proving to be a reliable guide in the identification of existing gaps and failures, and of extra work and improvements needed in the informal processes adopted (Santana et al., 2008). They represent important tools for the software architect because they have the ability to capture and to summarize the knowledge in a relatively simple way, as an ordered sequence of business activities and supporting information (White, 2004). They can also be applied to construct specific processes for each application. In service oriented architectures, it is usually recommended to design processes to identify the system issues, such as communication requirements (Huhns and Singh, 2005). Since this work considers SaaS and Cloud Computing as a natural part and the future of biodiversity information systems, a reference process is recommended to achieve the desired purposes.

In order to present the reference process, a notation is required. BPMN is a standard developed to provide a flow-chart based notation understandable by all business users (White, 2004), including business analysts, technical developers responsible for implementing the technology, and business people. Business Process Modeling (BPM) comprises the activities of representing, analyzing and improving the process of an enterprise, in order to achieve better levels of quality and efficiency.

A software package to design processes in BPMN is called BPMS, Business Process Management System. There are several of these systems available that implement the same notation (BPMN Information Home, 2013; Bizagi, 2013). The differences among BPMN packages are mainly related to functionalities other than design, such as performance evaluation and process analysis. Since the purpose here is only to design and to introduce the process, most of them would be suitable. The Bizagi Process Modeler™ was chosen to design the reference process for bee classification presented in this work because it is a free, standard-based, BPMS, which would thus be easily accessible by the biodiversity community.

Another advantage of BPMN is the existence of resources to map the graphical representation of a process to a services instantiation and
execution language in order to enable process execution (White and Miers, 2008). Most BPMS packages implement colored BPEL as their execution language. BPEL is a XML-based language to execute web services compliant with the W3C standards. In BPEL, the activities of the process are invoked in a specific sequence in order to achieve the business goal. Services orchestration and choreography are performed, and service-oriented solutions are easily implemented.

BPMN is based on a flowcharting technique adapted to create graphical models of business processes (White, 2004; White and Miers, 2008). In this context, a business process is a set of graphical objects (symbols) representing the activities, events and supporting information of a process, according to a specific order. The main BPMN symbols are presented in Fig. 1.

Symbols A, B, C, and D represent events (White, 2004; White and Miers, 2008). An event is something that happens during the course of a business process, potentially affecting the process flow. Events usually have a trigger or a result; they can start, interrupt, or end the flow. In BPMN, events are circles and the boundaries of the figures determine the type of event. A represents a start event and it indicates the beginning of a process. B represents an intermediate event; it may occur in between the beginning and the end of the process. C represents an end event and it indicates where a process will end. Start, intermediate and end events can be associated to triggers or expected results that indicate the specific circumstances of the event in the process flow. For example, the event described by D is associated with a message, indicating that the trigger is a message, in the case of start and intermediate events, or a message is the expected result of the process, in the case of an end event. There are several other triggers in addition to messages, such as timers, rules, and errors, among others. It is also possible to define triggers based on multiple events.

Symbols E, F, G, and H represent gateways (White, 2004) and they are used to control sequence flows. They are represented by diamonds and internal markers indicate different behaviors. E is an exclusive gateway, which means that only one of the outgoing paths will be taken when the process is executed. The decision mechanism can be based on data and conditions’ expressions, or events. F is a parallel gateway. When it is positioned at the beginning of a sequence flow fragment, it means that one or more paths can be defined as outgoing paths that might be taken when the process is executed. At the end of a sequence flow fragment, it can be applied to synchronize parallel paths. G is an inclusive gateway, which means that more than one outgoing path is possible. Those paths are usually followed by another inclusive gateway to merge the results of previous outcomes. H is a complex gateway, which means that more complex behavior definitions can be applied in the decision process. It can be defined for modeling both merging and splitting behavior.

I, J, K, and L are named activities. Activities represent work to be performed within the process. Activities may represent tasks or subprocesses, and in BPMN they are drawn as rectangles. They can be performed once or have internally defined loops. I represents a task, and is basically an atomic activity (White, 2004; White and Miers, 2008). J represents a sub-process, which can be a set of activities or even other sub-processes (White, 2004; White and Miers, 2008). The sub-process should be expanded and detailed eventually as part of the process which is being designed. The Bizagi Process Modeler has different symbols to identify different types of tasks (Bizagi, 2013), such as the ones represented by K and L. K represents a user task. User tasks are the ones that require human interaction with a software application (Bizagi, 2013). L represents a manual task, which means the ones that must be completed by a person without software interaction (Bizagi, 2013). Besides user and manual tasks, there are specific symbols for service, script, send, receive, and reference tasks.

While A to L are flow objects, M to P are artifacts (White, 2004; White and Miers, 2008). M is a symbol to establish bounds for an embedded sub-process. An embedded sub-process is fully contained in the parent process, from which it is also instantiated. The activities have access to the same data as the parent process, and they are reusable. N represents a data object, applied to show how data and documents are used within processes. They are applied to define inputs and outputs of activities. O represents text annotations, which can be applied to explain or to clarify a process, providing additional information about the process. They can be connected to flow objects, such as tasks. P represents a group. Groups are used to indicate relationships between process elements.

Besides flow objects and artifacts, BPMN also has connectors and swim lanes (White, 2004; White and Miers, 2008). There are three types of connectors: sequence flow, message flow and association.

---

Please cite this article as: Santana, F.S., et al., A reference process for automating bee species identification based on wing images and digital image processing, Ecological Informatics (2014), http://dx.doi.org/10.1016/j.ecoinf.2013.12.001
Sequence flows are applied to show the order in which activities should be performed; they are drawn by solid arrows. Message flows present the flow of messages between the entities that send and receive them; they are drawn by dashed arrows. Associations are applied to associate data, information, and artifacts to flow objects; they are drawn by dotted arrows or lines. Swim lanes help partitioning and organizing activities in a more detailed way. They are classified as pool and lanes and are drawn similarly as an Olympic swimming pool with its lanes. In a process in which different participants are required, pools represent the participants in the process, e.g., buyer and seller in a commercial transaction. Lanes are subpartitions for objects within a pool, e.g., manager and associate in the buyer or in the seller. For the sake of simplicity, swim lanes are not being used in the reference process presented herein, but they can be easily incorporated if necessary.

3. The reference process for automating bee species identification

The design of the reference process was based on research on tools and methods for bee identification and interviews with experts in both bee classification and image processing areas. The purpose was to synthesize the experts’ knowledge and to understand their difficulties, in order to propose a seamless and clear enough process to support the development of a reliable tool for bee classification. Fig. 2 introduces the reference process for bee classification. It is divided into sub-processes, described as follows. This process was designed in BPMN.

3.1. Image Acquisition

Image Acquisition is the first sub-process. The purpose of this step is to obtain the image of a bee wing so that analysis can be performed to obtain the bee classification, either by using live bees or by dissecting the bee wing. The wing can be put under a microscope or on a very white and clean background in very bright lighted conditions. Some researchers mount the wing in glass photographic frames. It is important to minimize glare and shadows when taking the digital images. Some known size reference or scale is recommended to be included in the image. Wing images can also be captured by scanning the wings. The wings should be well framed in the images, occupying the largest area possible without causing occlusion of the regions of interest. Images can be in grayscale or colored, and stored in JPEG, TIFF, or other formats.

A proper image resolution should be pursued. Unnecessary margins of the images can be eliminated by defining the region of interest (wing) with a bounding rectangle, and then cropping each image. If necessary, alignment operations, superimposition methods, and image registration must be performed so that the wings follow positioning specifications in relation to the boundaries of the images or previous models of the bees (Bookstein, 1997; Gonzalez and Woods, 2002; Snyder and Qi, 2004; Zitova and Flusser, 2003). Following this acquisition protocol, the images of bee wings can be processed. Inappropriate images should be discarded.

Examples of procedures for acquiring images of bee wings can be found in the literature. In the bee identification system developed by Schröder et al. (Schröder et al., 2002), the image acquisition works with live bees as well as with mounted collection specimens without removal of any body parts. The system consists of an electronic notebook connected with a CCD camera, which is mounted on a stereomicroscope. The user clips the forewing of the bee under the microscope slide and captures its image, which is transferred from the camera to the notebook. In Roggero and Entrèves (Roggero and d’Entrèves, 2005), each wing was cut off at the tegulae by a scalpel. The wings were put in glass microvials and cleared by 5% KOH watery solution for almost 10 min, then washed with distilled water before being transferred to 70% ethanol. The cleared wing was placed in 90% ethanol, then in Euparal essence, and mounted in Euparal mounting medium. Once fixed, the wing slides were examined under a microscope to describe the venation of both wings. A digital camera Olympus DP11 attached to a stereoscopic microscope Leica MZ28 captured forewing and hindwing images, stored in JPEG format. Tofliski (Tofliski, 2008) used 900 left forewings of three honeybee subspecies (300 workers per subspecies, collected from 30 colonies) in his investigation. Each wing was dissected, mounted in glass photographic frames (Rowi 260) and scanned with a Nikon Coolscan 5000 ED scanner equipped with a SF-210 slide feeder, with image resolution 2400 dpi.

3.2. Digital Image Processing

Following the Image Acquisition sub-process, the Digital Image Processing sub-process is presented in Fig. 3. It is divided into two sub-processes: Image Preprocessing and Feature Extraction. The Image Preprocessing sub-process aims to improve images and its output is a modified image. The Feature Extraction sub-process aims to extract sets of measurements which characterize either the entire image or some component thereof, and the output consists of information about the image content.

The Image Preprocessing sub-process receives an image as input and generates a modified image as output, which should be suitable for the next step, Feature Extraction. The Image Preprocessing sub-process may perform noise removal, enhancement, and segmentation. The BPMN diagram shown in Fig. 4 indicates that one or more of these functions can be applied in parallel or individually, and that they can be performed several times until the quality of the image is satisfactory.

Digital images often contain noise due to imperfect sensors, problems with acquisition, compression, storage or transmission of the image. Noise can interfere with the data of interest. There are appropriate methods for removing each type of noise; therefore, each image must be filtered with the most suitable filter to remove or to reduce noise. For example, a convolution of the original image with a Gaussian kernel can reduce Gaussian noise and a median filter can reduce salt-and-pepper noise type (Gonzalez and Woods, 2002; Snyder and Qi, 2004). The grain effect is a noise resulting from the use of a CCD sensor to capture the image; this effect can be reduced by applying a convolution with a Gaussian kernel; however, this process will insert blur in the image.

![Fig. 2. The reference process for bee classification.](image-url)
Image enhancement is a process whose goal is to provide better input, for either a human observer or an automated or semi-automated image processing system, by modifying attributes of the image to make it more suitable for the feature extraction task. Enhancement performs operations such as contrast stretching (including functions such as histogram equalization), brightness scaling, and edge sharpening, among others (Gonzalez and Woods, 2002; Snyder and Qi, 2004).

Segmentation is the operation of separating meaningful regions from the background. Popular segmentation functions include threshold-based, region-based (or connected component analysis), and edge-based algorithms. Segmentation of an image is then a partitioning into connected regions, where each region is homogeneous in some aspect and is identified by a unique label. For example, DrawWing (Tofinski, 2004) is a freely-distributed computer program to automatically extract landmarks from an insect wing. DrawWing is a threshold-based approach that uses two threshold values to segment the wing image: the first determines the outline of the wing, and the second extracts the venation outline. The venation outline is then converted to its skeleton by a thinning algorithm (Gonzalez and Woods, 2002), and all veins shorter than a specified value can be removed so that landmarks can be extracted.

The Image Preprocessing sub-process is more relevant in the case of fully automatic extraction of features (e.g. (Tofinski, 2008)). In other cases, when the process is performed interactively with the specialist in either a totally manual or a semi-automatic mode, it is only necessary for the specialist to approve the image available and then to demarcate the features of interest (e.g. (Schröder et al., 2002)).

The main component of the Digital Image Processing sub-process is the step of extracting features, taking measurements, geometric or otherwise, of meaningful regions (possibly segmented) in the image. In this case, features are a set of numbers that characterize some property of the wings captured in the images. They are used in a classifier. The Feature Extraction sub-process may be a totally manual process, a semi-automatic process, or a fully automatic process.

Some alternative methods for feature extraction in bee-wing classification include multivariate morphometric methods, outline methods, and geometric morphometric methods (Adams et al., 2004), as shown in Fig. 5.

In multivariate morphometric methods for bee-wing classification, linear distance measurements are applied, sometimes in conjunction with angles, ratios, and counts, so that covariation in the morphological measurements is quantified and variations within and among samples can be assessed.

Outline methods use the bounding edge of a structure or region, which can be considered homologous across specimens, i.e., structures with the same embryological origin. Examples include the value of changes in the angle of tangents at each point along an outline and lengths of equally spaced radii from a central point.

The application of geometric morphometric methods is a successful approach in many cases. It is based on outlining structures and also on landmarks, as it is the case of this study. Landmarks are sets of point coordinates of biologically definable landmarks. Since the point coordinates depend on the position, orientation, and scale of the sample, non-shape variation must be removed before analysis. Then, the calculation of the differences in coordinates of corresponding landmarks between objects can assess shape.

For example, in the bee identification system developed by Schröder et al. (2002), there is a manual first step in which the user marks defined vein junctions with a mouse-click on the forewing image, and then the system automatically connects the junctions using a line-following approach.
algorithm. The bee identification is based on the venation features of the forewing, such as vein length, width, curvature, angles and the area of all cells. The system calculates about 200 features of the venation that are used in the classification process.

Roggero and d’Entrèves (2005) collected 16 landmarks on the venation of the forewing and 12 on the hindwing using the tpsDig software (Rohlf, 2010). However, their interest did not lie in classifying bees, but rather in assessing attributes and variations between two populations of *Scythris obscurella*.

Toﬁlski (2008) used the DrawWing software (Toﬁlski, 2004) to automatically determine the coordinates of 18 vein junctions used as landmarks for geometric morphometrics. The landmarks were aligned according to generalized orthogonal least-squares procedures using tpsSuper software (Morphometrics at SUNY Stony Brook, 2013), and centroid size was used as a feature. Standard morphometry was based on 4 distances and 11 angles. 13 out of 15 variables were used in the classification model on standard morphometry, and 22 out of 37 variables were used in the geometric morphometrics classification model. Several other studies (Aytekin et al., 2007; Francoy et al., 2012b; Meulemeester et al., 2012; Michez et al., 2009; Wappler et al., 2012) aiming fossil and currently extant bee species identification were conducted using landmark based methods. The number of landmarks and extracted features varied according to the study, but they have all presented good discrimination power among the studied groups and provided important information for the understanding of several questions.

### 3.3. Classification

Classification is the sub-process in which the image features should allow the classification of the species. The features extracted from the images are initially labeled by the expert in their categories, creating a sample set that may be applied for training the classifier.

In the case of the subject of this paper, the type of learning procedure widely used to generate the output value is the Supervised learning (see Fig. 6). Supervised learning assumes that a set of training data has been provided, consisting of a set of instances that have been properly labeled by hand with the correct output. A learning procedure then generates a model that attempts to meet two, sometimes conflicting, objectives: to deliver good performance with training data and deliver good generalization with new data.

---

Please cite this article as: Santana, F.S., et al., A reference process for automating bee species identification based on wing images and digital image processing, Ecological Informatics (2014), http://dx.doi.org/10.1016/j.ecoinf.2013.12.001
In the first goal of the learning procedure, a preliminary fitting procedure optimizes the model parameters to make the model fit the training data as well as possible. However, if an independent sample of the validation data is taken from the same population from which the training data were extracted, it probably will be observed that the model will not fit so well in the validation data as it fits in the training data. This effect is known as overfitting, and it is particularly likely to occur when the size of the set of training data is small or when the number of model parameters is large. Hence, the second goal of the learning procedure consists of generalizing the model so as to perform as well as possible on new data. A conflict between the goals may occur when attempting to perform as well as possible on both, the training data and the new data.

A popular solution to get around this problem is to use the technique of cross-validation. Cross-validation is applied to predict the fit of a model to a new data set by dividing the labeled data with the correct output into two sets: one used to train a model – the training set – and the other used to validate the model – the validation set. A commonly applied type of cross-validation method is the k-fold cross-validation, in which the data is initially divided into k folds of approximately equal size. Then k rounds of training and validation are performed, so that at each iteration a different data fold is used for validation, while the remaining k−1 folds are used for training. The k results from the folds then can be averaged (or otherwise combined) to produce a single estimation.

Some very popular algorithms for classification can be organized as classifiers based on supervised learning predicting categorical labels either using parametric model, such as Linear Discriminant Analysis (LDA) and Logistic Regression, or using nonparametric models, such as Decision Trees, K-nearest-neighbor (KNN) algorithms, Naive Bayes classifier, Multi-Layer Perceptrons (MLP), and Support Vector Machines (SVM).

Linear discriminant analysis (LDA) is a method used to find a linear combination of features which characterizes or separates two or more classes of objects or events. The resulting combination may be used as a linear classifier, or for dimensionality reduction before later classification. LDA works when the measurements of independent variables for each observation are continuous quantities. Logistic regression is similar to LDA, but it is used in applications in which it is not reasonable to assume that the independent variables are normally distributed, which is a fundamental assumption of the LDA method.

A Decision Tree is a decision support tool that uses a tree-like graph to represent decisions and their possible consequences. A decision tree works iteratively, at different levels of a hierarchy of classification space. The system developed by Roth et al. (1999) is quite elaborate and works iteratively, at different levels of a hierarchy of classifiers, performing all the steps automatically. Initially, it extracts a morphologically and geometrically robust set of substructures (three wing cells are used as the initial substructure). Numerical attributes (compactness, relative positions, relative angles of principal axes of inertia) of these three cells are used as input for a LDA classifier (first level of the hierarchy) that selects a template from a precomputed set of about 30 templates. Templates represent groups reflecting the genera structure of the initial set that contained 550 specimens, with approximately 15 species per genera. The templates were computed by averaging a manually selected representative training set of wing images, and they contain information about systematic disturbances for the given bee group. The selected template is used to guide the search for other structures in the image. Having all the structures of interest, a 110 dimensional feature vector that represents shape features (lengths, angles, and area descriptions) is computed; then eigenvalue decomposition is carried out to select the principal components that contained 99% of the original variance (this process reduced the 110-dimensional feature vector into a 50-dimensional vector). The system then rotates and scales the images so that they have the same orientation and the same distance between two reference points. This process is done across the entire image because pixel information is also used in the classifier. Then, a rectangular frame box is placed around one of the points to always choose the same clipping of the image. After an intensity normalization procedure based on the mean gray value within the clipping window, the images are downsampled to 12 × 20 pixels whose corresponding gray values are used as the
components of a 240-dimensional feature vector. This vector is combined with the 50-dimensional vector, resulting in a 290-dimensional feature vector, which is used as input for the classification (last hierarchical level), which is performed either with a SVM classifier or with a Kernel Discriminant Analysis classifier, leading to very good classification results (i.e., misclassification rate of only 0.7%).

Tofilski (2008) used forward stepwise discriminant function analysis (S. INC, 2013) to determine classification functions based on the 18 aligned landmark coordinates and centroid size (geometric morphometrics) or 4 distances and 11 angles (standard morphometry), followed by canonical analysis. Differences between subspecies were tested using MANOVA of partial warp scores produced by the tpsRelw software package (Morphometrics at SUNY Stony Brook, 2013). The cross validation test was used to verify the accuracy of both methods, geometric morphometrics and standard morphometry. In this test, half of the colonies were used as a training set and the other half of the colonies were used as a validation set.

Several other methods for each subprocess shown in Fig. 2 have been described in the literature (Francoy et al., 2008, 2009, 2012a; Nielsen et al., 1999; Roth et al., 1999; Schröder et al., 2002; Tofilski, 2008; Weeks et al., 1997). They vary according to the system specifications, to the purpose of the system, and to the needs of the user. The adequacy of the system must be evaluated by a final process known as validation.

3.4. Validation

Validation is the process of evaluating a system during or at the end of the development process to determine whether it satisfies specified user requirements and that it fulfils its intended purpose, i.e., validation checks that the product design satisfies or fits the intended usage and ensures that the product actually meets the user’s needs. This is done through dynamic testing and other forms of review. It often involves acceptance and suitability with external customers. The most tested attributes in validation tasks may include, but are not limited to: selectivity/specificity, accuracy and precision, reproducibility, false positive, limit of detection, and system suitability.

In the validation process, several labeled samples should be tested and, if the result is not adequate, the various stages of the process should be reviewed in order to achieve the result expected.

4. Experiments

Experiments were conducted to illustrate the reference process. RGB images, with 1360 × 1024 pixels and 150 ppi resolution, were obtained using a digital camera coupled to a magnifying glass. Images correspond to the forewing of males of five species of the *Euglossa* genus (Fig. 7 is one example of such images). Table 1 shows the dataset specification.

In order to extract features for classification, 18 landmarks were chosen using the tool tpsDig (Rohlf, 2010). The landmarks are plotted as red diamonds on Fig. 7.

In this paper, two case studies are presented. The first one uses features based only on the landmarks, whereas the second one adds features extracted from pixel values of the wing images, similarly to the features used by Roth et al. (1999). The main goal of the experiments was to evaluate whether the addition of pixel-based features to landmark-based features in Case Study 2 can improve the classification performance shown by Case Study 1, which uses only landmark-based features. The same dataset summarized in Table 1 was used for both case studies.

In both cases, all the classifiers presented in Section 3.3, namely LDA, Logistic, Decision Tree, KNN, Naive Bayes, MLP and SVM, were evaluated with a 10-fold cross-validation, where the percentage of correct classifications for each classifier is estimated as the mean of the ten tests performed.

In this paper, the pruned C4.5 (Quinlan, 1993) algorithm was used as the implementation of Decision Trees, tuned with the parameter confidence factor $C = 0.25$. The KNN classifier was tested with the parameter $k$ ranging from 1 to 10, and the best result was selected; the Euclidean distance in the feature space was used as distance metric. The MLP was trained with learning rate $\alpha = 0.3$ and the number of neurons of the hidden layers was set to the sum of the number of classes and attributes divided by two. Empirically, those parameters achieved a good performance, however, any technique might be used to calculate the number of neurons in the hidden layer (Statthakis, 2009). The SVM algorithm was evaluated with a polynomial kernel, an efficient kernel for multiclass classification (Sangeetha and K.B., 2011).

In order to test the classifiers, the WEKA data mining software (Hall et al., 2009) was adopted. It implements a popular collection of machine learning algorithms for data preprocessing, classification, regression, clustering, association rules, and visualization. Fig. 8 presents an example of the MLP classifier run on Case Study 2, where it is possible to see the classification rate obtained by cross-validation, the confusion matrix, and other statistical measures. All classifiers were executed on WEKA, except the LDA, that was executed on MATLAB (The MathWorks Inc., 2012).

The following subsections show how the feature vectors were obtained for both case studies and describe the results of the experiments.

4.1. Case Study 1

The feature vector used for classification in this case study consists only of landmark-based features. That is the case of many studies on the literature (Francoy et al., 2008; Tofilski, 2008).

The first feature extracted is the Centroid Size, a widely used feature from geometric morphometrics. The set of landmarks $L$ is composed of $|L|$ landmark coordinates $(x_i, y_i)$, $i = 1, 2, \ldots, |L|$. The centroid of $L$ is $(x, y)$, given by:

$$
  x = \frac{1}{|L|} \sum_{i=1}^{1} x_i \quad \text{and} \quad y = \frac{1}{|L|} \sum_{i=1}^{1} y_i.
$$

| Table 1 |

<p>| Number of specimens per species on the data set, totaling 138 images. |</p>
<table>
<thead>
<tr>
<th>Species</th>
<th>Images</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Euglossa flavmea</em></td>
<td>29</td>
</tr>
<tr>
<td><em>Euglossa ignita</em></td>
<td>26</td>
</tr>
<tr>
<td><em>Euglossa imperialis</em></td>
<td>29</td>
</tr>
<tr>
<td><em>Euglossa orellana</em></td>
<td>26</td>
</tr>
<tr>
<td><em>Euglossa chalybeata</em></td>
<td>28</td>
</tr>
</tbody>
</table>

Fig. 7. *Euglossa imperialis* forewing with landmarks.

Please cite this article as: Santana, F.S., et al., A reference process for automating bee species identification based on wing images and digital image processing, Ecological Informatics (2014), http://dx.doi.org/10.1016/j.ecoinf.2013.12.001
The Centroid Size is the squared root of the sum of the squared distances between each landmark and their centroid (Bookstein, 1997):

\[ CS = \sqrt{\sum_{i=1}^{n} (x_i - \bar{x})^2 + (y_i - \bar{y})^2} \]  

As the images were taken without concern about rotation, the following method was applied to determine the wing’s rotation.

The landmarks shown in Fig. 9a were taken as vertexes of a regular polygon, and it was generated a binary image where any position inside the polygon is set to white, while all the other positions are set to black (e.g. Fig. 9b). Note that, in order to generate this image, the original wing image is not necessary, the landmark’s coordinates are enough.

Then, the following second moments of the binary image were computed (Grimson, 1990):

\[
\begin{align*}
    m_{2,0} &= \sum_{x} \sum_{y} x^2 b(x, y), \\
    m_{1,1} &= \sum_{x} \sum_{y} x y b(x, y), \\
    m_{0,2} &= \sum_{x} \sum_{y} y^2 b(x, y),
\end{align*}
\]

where \(b(x,y)\) is the intensity of pixel \((x,y)\) (the white color stands for the intensity of 1, while the black color stands for 0), and \(x’\) and \(y’\) are:

\[
\begin{align*}
    x’ &= x - x_c, \\
    y’ &= y - y_c,
\end{align*}
\]

with \((x_c,y_c)\) the position of the center of mass of the image:

\[
\begin{align*}
    x_c &= \frac{\sum_{x} \sum_{y} x b(x, y)}{\sum_{x} \sum_{y} b(x, y)} \quad \text{and} \quad y_c = \frac{\sum_{x} \sum_{y} y b(x, y)}{\sum_{x} \sum_{y} b(x, y)}.
\end{align*}
\]

Now, the moments from Eq. (3) can be applied to determine the angle of rotation \(\theta\) of the wing (in radians):

\[
2\theta = \arcsin \left( \frac{2m_{1,1}}{\sqrt{4m_{1,1}^2 + (m_{2,0} - m_{0,2})^2}} \right)
\]

Once the angle is computed, rotation invariance can be achieved by rotating all the configurations of landmarks by their correspondent angles. It corresponds to a dot product of the landmark matrix with the following rotation matrix:

\[
\text{RotationMatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}
\]

The application of the dot product with the described Rotation Matrix corresponds to a rotation around the point \((0, 0)\); hence, for a rotation...
around the center of mass \((x_i, y_i)\), it is necessary to subtract the center of mass from the points that should be rotated before applying the dot product, and add the center of mass after the dot product, as follows:

\[
\text{NewPoint} = (\text{Point} - \text{CenterOfMass}) \times \text{RotationMatrix} + \text{CenterOfMass}.
\]  
(7)

In order to provide scale and translation invariance, an Orthogonal Procrustes Analysis (Rohlf and Slice, 1990) was performed after the rotation by \(\theta\) of the image. The Bookstein coordinates (Bookstein, 1997) can be viewed as a variation of the method applied in this paper, and could also be used to obtain the aligned landmarks.

The resulting aligned landmarks can be obtained with the equation:

\[
L' = \frac{(I - P)L}{s},
\]

where

\[
s = \sqrt{\text{tr}((I - P)L'(I - P))}
\]

and \(L\) is the matrix with the \((x,y)\) coordinates of all the landmarks, \(I\) is an \(|L| \times |L|\) identity matrix (\(|L|\) stands for the number of landmarks), \(P\) is an \(|L| \times |L|\) matrix with all elements equal to \(\frac{1}{|L|}\) and \(\text{tr}(A)\) refers to the sum of all principal diagonal elements of \(A\).

Finally, the feature vector was built with the \((x_i, y_i)\) position of each aligned landmark from Eq. (8) and the centroid size \(CS\) from Eq. (2), resulting in a vector with 37 features:

\[
\text{FeatureVector} = [x_1, y_1, x_2, y_2, \ldots, x_{18}, y_{18}, CS, k_1, k_2, \ldots, k_{256}].
\]

**4.2. Results for Case Study 1**

Table 2 shows the percentage of correct classifications in this experiment for all the classifiers. The LDA was the best classifier in Case Study 1. It means that the species evaluated in this paper can be reasonably well described linearly, since the linear discrimination had a classification rate of 81.21%.

**4.3. Case Study 2**

In the second case study, pixel-based features were used in addition to the landmark-based features described in Section 4.1. This approach was applied in Roth et al.’s system (1999).

After calculating the angle of rotation in Eq. (6), the entire image was rotated in the same way as the landmarks, thus assuring that all the images are in the same rotation. After this procedure, the minimal bounding box that holds all the landmarks was calculated with a 5 pixel margin, then the image was cropped using the bounding box as a mask. The resulting image was divided into 256 quadrants, as illustrated in Fig. 10, and the following value was extracted for each quadrant to be used as a feature:

\[
k_i = T_i + \sigma r_i,
\]

where \(T_i\) and \(\sigma r_i\) are, respectively, the mean value and the standard deviation of the blue channel of the pixels inside the quadrant \(i\). A gray-scale image could also be used to extract this feature; however, in these experiments, the blue channel provided a slightly better performance.

**4.4. Results for Case Study 2**

The results of this case study are presented in Table 3. Although the LDA still presented a good performance (classification rate of 84.18%), the MLP had a better classification rate, indicating that the ability of handling non-linearly separable data can be beneficial for pixel-based features. These results agree with (Roth et al., 1999), though in that case a good performance was not achieved with LDA and the SVM was applied to improve the results. C4.5 and Naive Bayes displayed a decrease in the classification performance, which might indicate divergences in handling the increase of the dimensionality of the feature vector. This result was to be expected for Naive Bayes, since it relies on independent features. This fact is not assured in the feature vector used in this case study.

By analyzing the two case studies, it is possible to observe that there was a performance gain when adding pixel-based features to the feature vectors composed of landmark-based features. In Case Study 1, the best performance was 81.21%, while in Case Study 2, the best performance was 87.68%, thus indicating that pixel-based features are beneficial.

**4.5. Data set minimum size**

In order to evaluate the minimum number of specimens required when using the features from Case Study 2, the following experiment was conducted. Initially, the data set was split randomly in 60% of the specimens intended to training and all the remaining to testing. The percentage of specimens for training was progressively reduced by 5% per iteration, until this reduction was not possible anymore. For all the different splits, the MLP (the best classifier defined in Section 4.4) was executed with the features from Case Study 2.

Fig. 11 shows the results obtained. As expected, the classification performance fell down with the reduction of the training set. To achieve more than 70% of hit rate, about 30% of the entire data set was used (approximately, 41 specimens).

**5. Discussion**

The reference process for bee classification based on wing images and digital image processing presented in this work introduces the main steps required to perform automatic bee recognition applying
computational techniques related to image processing and machine learning. It is part of a major effort to improve the techniques, tools and software packages to make biodiversity studies more effective. However, several aspects of this problem should be addressed in order to perform bee identification and morphological classification properly. They are discussed as follows.

Species and subspecies identification and classification are not trivial activities to perform, even when software complexity is not involved. The taxonomic impediment was discussed in Section 1, regarding the lack of trained taxonomists and auxiliary tools to obtain more taxonomic data, but this is not the only problem. Different identification methods can be applied to identify or to classify the same species, like identification of diagnostic features, morphometrics (Francoy et al., 2012a), cuticular hydrocarbons (Francisco et al., 2008), SNPs (Whitfield et al., 2006) and others. Sometimes the same method may not be applied to all cases, e.g., in the case of species that are morphologically similar, the appliance of more than one technique is often required. Even if they lead to the same results, it is still a problem when it comes to developing automatic tools for species identification and classification, because developing not only a method, but several methods to study the same species is required. The amount of available methods is directly proportional to the effort towards building automatic tools for this problem. That is probably part of the reason why the number and variety of such tools are still not enough, and are part of the taxonomic impediment problem.

A reference process mainly introduces a general approach to a problem, disregarding technical details that might be necessary to build specific solutions. Therefore, before starting the system development, the system analyst must understand that the reference process should be used as a starting point, but system requirements unrelated to the reference process should also be considered, and added when necessary. It is not a purpose of processes, or of this paper for that matter, to establish all the technical details of system development for bee identification, and the software tools that are casually referred here have mainly illustrative character. This is actually a desired feature of reference processes because it confers more versatility and flexibility to their concept and application.

In the first sub-process, Image Acquisition, a proper image should be pursued, or the system will not be able to process it. For example, consider the two images in Fig. 12. The first image, labeled as (a), is clear and is recognizable by automatic methods. In the second image, labeled as (b), some spots are observed due to dirt in the process of acquisition, polluting the image. As can be observed in the processed version of the image, the spots might be considered as proper features of the wing and, if it happens, they will be included in all further analysis performed by the system. Therefore, the automatic identification and classification will not be possible unless a bee with the same features introduced by the spot exists — of course, the result would not be accurate in this case. Some methods to obtain an adequate image and a list of features that are important to make automatic identification possible are presented in Section 3.1.

In the second sub-process, Digital Image Processing, described in Section 3.2, the image is submitted to a preprocessing step to improve its quality, e.g., by applying enhancement and denoising methods. The preprocessing is followed by the extraction of the features that will be necessary for classification. This sub-process can be performed with or without user interaction. In order to have a solution without user interaction, the quality of the image is even more important for image processing because the system might, for example, enhance the wrong point, as observed in Fig. 12 (processed image (b), on the right), where the spot was processed as a real part of the wing. In this case, the algorithms must be more sophisticated and able to decide when enough processing was conducted, meaning the image is good enough to have its features extracted. Concerning features extraction, the multiplicity of classifying methods is the problem here, because different methods might require different features.

In the third sub-process, Classification, described in Section 3.3, the extracted features are submitted to machine learning algorithms so as to classify the wing image. Supposing the image is adequate and the features have been correctly extracted, the problem here is to match the extracted features with the features that describe a species in a set of species previously established, or to identify the image as not pertaining to any species in this set. There are several possible strategies to apply here, but not all of them can be applied to the same problems with the same efficiency. Most of the time, several strategies must be tried in the search for a reliable classification, e.g., when the classification fails, or the accuracy of the classifier is not good enough to accept the results, other strategies should be applied. The most recommended strategies are to try a different method, or the same method with a different set of features, which can be obtained by adding, removing or exchanging features. It is also possible to improve a method by adding training...
samples to the machine learning algorithms, in order to supposedly improve their effectiveness. Of course, it depends on the availability of samples to do so. Misclassification of a wing image of a new species that was not present in the training data set is another problem that might be expected to happen here, because most machine learning algorithms try to match the image with another one in the established set, and sometimes a negative answer is not possible because it is not predicted by the algorithm. These algorithms assign a new wing image to one of a set of previously known bee classes. Therefore, once the image enters the classification sub-process, it will be identified even when it should not. In this case, we say it is a false positive. The one-class classification algorithms (Moya and Hush, 1996) are exceptions to this procedure. The task of the one-class classification is to define a boundary around the target class, such that it accepts as much of the target objects as possible, while it minimizes the chance of accepting outlier objects.

The last sub-process of the reference process, Validation, described in Section 3.4, is rather difficult to define. In some cases, further statistical analyses might be required; in others, the decision of a researcher might be enough. However, if the idea is to have a fully automatic classification system, it is desirable that, once the classifier was trained and validated with the training set, the expert evaluates the classifier’s performance with new specimens that were not available at the moment of the training and, if the performance is not acceptable, returns to the training adding the new specimens to the data set. It is possible to develop a system that would continuously return to previous steps of the reference process until an acceptable solution is reached, but not before clearly defining a criteria of reliability. Until these criteria are not clearly described in the literature, this step will be highly dependent on human interaction and on the researcher’s knowledge (Francoy et al., 2012a). Finally, note that the process, methods, recommendations, and even some of the software tools used to classify bees based on the image of their wings, introduced in this paper, are easily extended to other species, since the same basic principles, e.g., the existence of morphometric methods for classification, are applicable. The technique can be very efficient if the necessary precautions are taken, as shown by the experiments presented in Section 4.

6. Conclusion

This work introduced a reference process for automating bee species identification based on wing images and digital image processing, detailing the necessary steps, methods, and recommendations to perform this complex activity. Bees were chosen because of their relevance for biodiversity conservation by providing vital services to agriculture and natural systems. The general purpose was to contribute to the effort towards providing techniques, software packages and tools to improve studies in biodiversity. Experiments were conducted to evaluate the process, assessing its adequacy to the problem, and to verify the application of image processing methods and machine learning algorithms to solve the problem proposed. Results show that the process is effective and accurate. However, there are aspects of the process that must be observed in order to improve the results.

The knowledge gained with this work is a relevant contribution to the development of an automatic classification software system. It is particularly important because this solution can be extended to the classification of other species, as long as similar criteria are applicable. The process can also be useful as a guide for beginners in this research field, as it explains all the required steps to perform the activity.

Acknowledgments

The authors are grateful to the Universidade de São Paulo for the support to the Biodiversity and Computing Research Center, BioComp; to CAPES for the support to Flavio Sales Truzzi and to Sheila Leal Santos; to FAPESP (proc. n. 2011/07857-9 and 2011/2980-8) for the support to Tiago Mauricio Francoy and to Anna Helena Reali Costa; and to CNPq (proc. n. 311058/2011-6, 303826/2010-5 and 131238/2013-2) for the support to Anna Helena Reali Costa, Antonio Mauro Saraiva, and Felipe Leno da Silva.

References


Roggero, A., d’Entrèves, P., 2005. Geometric morphometric analysis of wings variation between two populations of the Syrphus obscurella species-group: geographic or


S. INC, 2013. Statistica (data analysis software system).


