J. Vis. Commun. Image R. 25 (2014) 1516–1527

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Automatic identification of fruit flies (Diptera: Tephritidae)

F.A. Faria [a,*], P. Perre [b], R.A. Zucchi [c], L.R. Jorge [d], T.M. Lewinsohn [d], A. Rocha [a], R. da S. Torres [a]

[a] Institute of Computing – University of Campinas, SP, Brazil
[b] Department of Genetics and Evolutionary Biology – University of São Paulo, SP, Brazil
[c] Luiz de Queiroz'' College of Agriculture – University of São Paulo, SP, Brazil
[d] Institute of Biology – University of Campinas, SP, Brazil

ARTICLE INFO

Article history:
Received 14 November 2013
Accepted 22 June 2014
Available online 1 July 2014

Keywords:
Anastrepha
Multimodal fusion
Machine learning
Fraterculus group
Automatic identification
Classifier selection
Fruit flies
Wing and aculeus images

ABSTRACT

Fruit flies are pests of major economic importance in agriculture. Among these pests it is possible to highlight some species of genus *Anastrepha*, which attack a wide range of fruits, and are widely distributed in the American tropics and subtropics. Researchers seek to identify fruit flies in order to implement management and control programs as well as quarantine restrictions. However, fruit fly identification is manually performed by scarce specialists through analysis of morphological features of the mesonotum, wing, and aculeus. Our objective is to find solid knowledge that can serve as a basis for the development of a sounding automatic identification system of the *Anastrepha fraterculus* group, which is of high economic importance in Brazil. Wing and aculeus images datasets from three specimens have been used in this work. The experiments using a classifier multimodal fusion approach shows promising effectiveness results for identification of these fruit flies, with more than 98% classification accuracy, a remarkable result for this difficult problem.

1. Introduction

Fruit flies are pests of major economic importance in agriculture among which we highlight some species of genus *Anastrepha*. *Anastrepha* species occur exclusively in the American tropics and subtropics, with more than 250 valid and numerous undescribed species [1], but fewer than 10 species are of agricultural importance [2]. It is the most diverse genus of fruit flies, and most of its species have been divided into several species groups. However, the taxonomy of some groups is not yet properly solved [3]. In the fraterculus group, are included major pest species such as *Anastrepha fraterculus* (South American fruit fly) and *Anastrepha obliqua* (West Indian fruit fly). Both species infest several host species throughout their distribution. For example, larvae of *A. fraterculus* and *A. obliqua* develop on 81 and 36 commercial and wild host species, respectively, in Brazil [4].

The extent of fruit fly damage to commercially produced fruit is significant. In addition, quarantine restrictions imposed by fruit importing countries are another serious economic impact caused in case one fruit flies to be found. In this scenario, species identification is crucial for the implementation of management, control programs, and quarantine restrictions.

Anastrepha species identification is mainly based on subtle differences in the shape of the aculeus (the female egg-laying "needle"), but thoracic markings, wing pattern, and microtrichia are also important taxonomically. In the *fraterculus* group, besides external morphology, molecular [5], genetic [6], and morphometric [7] studies have also been carried out to clarify the identity of cryptic species.

Given the demand, novel tools for a quick and precise identification of fruit flies, amenable to automation, need to be developed and tested. This demand for computational solutions is due to a constant quest for reducing the time and costs in performing identification tasks. Among existing solutions, there are image analysis and machine learning techniques, which have been widely used in several application areas (e.g., security, medical image analysis, biology, and agriculture). In applications to agriculture, the use of image analysis and machine learning techniques is not rare. Arriba et al. [8], for example, have proposed an automatic leaf image classification system for sunflower crops. In [9,10], image processing techniques have been applied to classify or identify wheat, spelt, and hybrid seeds. In [11], in turn, texture analysis has been performed to differentiate bark from wood chips.

For automatic identification of species, some systems have been developed, such as: (1) Digital Automated Identification SYstem

* Corresponding author. Fax: +55 19 3521 5838.
E-mail addresses: ffaria@ic.unicamp.br (F.A. Faria), pperre01@gmail.com (P. Perre), razucchi@usp.br (R.A. Zucchi), leonardorejorge@gmail.com (L.R. Jorge), thomasl@unicamp.br (T.M. Lewinsohn), anderson.rocha@ic.unicamp.br (A. Rocha), rtorres@ic.unicamp.br (R. da S. Torres).

http://dx.doi.org/10.1016/j.jvcir.2014.06.014
1047-3203/

(DAISY), which performs fish, pollen, plant, and butterfly classification [12]; (2) SPecies IDentified Automatically (SPIDA-web), which is a tool for identifying Australian spiders, that allows to distinguish 121 species [13]; (3) Automatic Bee Identification System (ABIS), which identifies bees for species of genus *Bombus*, *Colletes*, and *Andrena* [14]. In this work, we used, for the first time in fruit flies, image analysis techniques to automatically identify three species of the *fraterculus* group: *Anastrepha fraterculus* (Wied.), *Anastrepha obliqua* (Macquart) and *Anastrepha sororcula* Zucchi through wings and aculei.

The objective of this work is to perform a robust study of description and learning techniques that can serve as support for the development of the first automatic identification system of these species using wing and aculeus images. We explore complementary image features through the use of a framework for classifier selection and fusion that point out which ones are more effective to capture the properties and nuances of the fruit flies allowing us to devise and deploy an automatic classification system. Furthermore, this work can serve as a guide for implementing new modules into existing systems in the literature.

We evaluated the use of several image description approaches that encode the color, texture, and shape properties of images of fruit fly wings and aculei into feature vectors. Those features are then used to train classifiers that are later combined by a meta-learning approach that identifies fruit fly species. In addition, we also assess the feasibility of classifying fruit flies based on their wings in order to make this task more objective than it is nowadays. In this context, the main contributions of this paper are:

1. Evaluation of several image descriptors for classifying fruit flies.
2. Exploration, for the first time, of fruit fly wings as a possible morphological feature for automatic classification.
3. Design and development of an automatic classification and fusion system able to explore complementary features present in wings and aculei of fruit flies.

The remainder of this paper is organized as follows. Section 2 presents related concepts necessary for the understanding this paper. Section 3 describes the proposed classifier multimodal fusion framework. Section 4 shows the experimental procedure we adopted to conduct our experiments, while Section 5 discusses the experiments and results. Finally, Section 6 concludes the paper and points out future research directions.

2. Related work and background

This section shows important concepts used in this work.

2.1. Species identification based on fruit fly images

The species identification task consists of labeling new fruit fly images among predefined classes. In this process, a classification model is created to indicate the class to which each new image belongs.

The identification task is composed of the dataset which is the input of the problem, pre-processing, feature extraction, learning, and identification steps. Fig. 1 depicts the sequence of steps in the process of fruit fly identification.

- **Dataset**: is a set of collected and organized images to be used in the identification task and represents the input of the problem.
- **Pre-processing**: relies on algorithms such as scaling, segmentation, and dilation in order to perform noise reduction or elimination and to improve the quality of the input images.
- **Description**: relies on algorithms to encode visual properties (e.g., color, texture, and shape) of dataset images into feature vectors.
- **Learning**: relies on algorithms that use image feature vectors to learn intra- and inter-class patterns of the images.
- **Identification**: the task of labeling a new fruit fly image into one of existing known classes based on the previously learned classification model and description techniques.

It is worth mentioning that the learning step can comprise a single learning method or a set of learning methods that explore complementary properties of the fruit fly images and combine them toward a higher classification effectiveness. We explore both possibilities in this paper. Section 4 shows all setup used for each step in this work.

2.2. Image acquisition and storage

In this section, we show the acquisition and storage processes of image wing and aculeus performed in our experiments.

2.2.1. Fruit flies samples

We have used specimens of *A. fraterculus*, *A. obliqua*, and *A. sororcula* from the collection of the Instituto Biológico of São Paulo (e.g., Fig. 2a–c). Specimens have been collected through McPhail-type traps (Fig. 2b) and reared flies from fruits as well. For each species, 100 individuals with aculei and wings in good condition were selected for analysis. Because the *fraterculus* complex comprises several cryptic species [15], the name *A. fraterculus* has been used in this work.

2.2.2. Image acquisition

Fig. 3 shows the employed three-step process for wing image acquisition: (a) the right wing of each specimen is dissected; (b) it is mount on a microscope slide with Euparal; (c) the slide is covered with a glass coverslip. The slides have been photographed with a Nikon DS-Fi1 camera (resolution 2560×1920) attached to a Nikon SMZ 1500 stereomicroscope (1.5*X* objective).

Fig. 4 shows the adopted six-step process for aculeus image capture. In (a), the oviscape has been dissected; (b) oviscape is treated in a solution of 10% KOH for 12 h; (c) aculeus has been removed, placed ventral side up on a microscope slide with glycerin; (d) the slide is covered with a glass coverslip. The aculeus has been photographed with the same Nikon DS-Fi1 camera attached to a Nikon microscope ($10\times$ and $40\times$ planachromatic objectives).

Notice that fruit fly identification is either based on the female aculei or on wings. Although more reliable, the aculeus requires the extraction and preparation of a minute female terminalia to perform visual or morphometrical identification. Wings, on the other hand, are easier to prepare, but identification is less objective. Both analyses need to be manually performed by scarce specialists and an automatic learning technique exploring wings and aculei information is paramount.

Fig. 1. General pipeline for identifying a fruit fly.

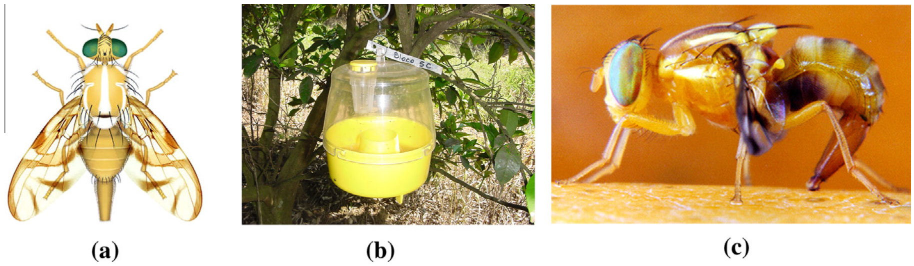


Fig. 2. (a) A fruit fly example (drawing) [54]; (b) a McPhail-type trap; and (c) a fruit fly laying eggs.

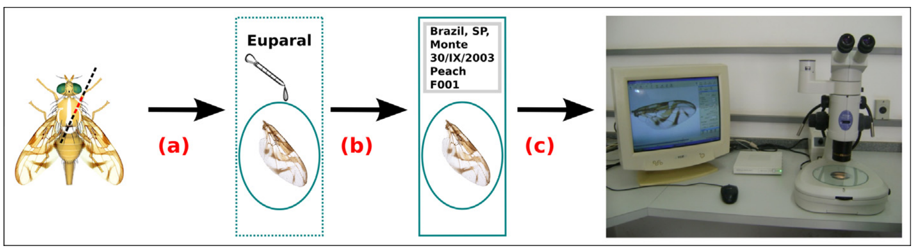


Fig. 3. The image acquisition process of wings.

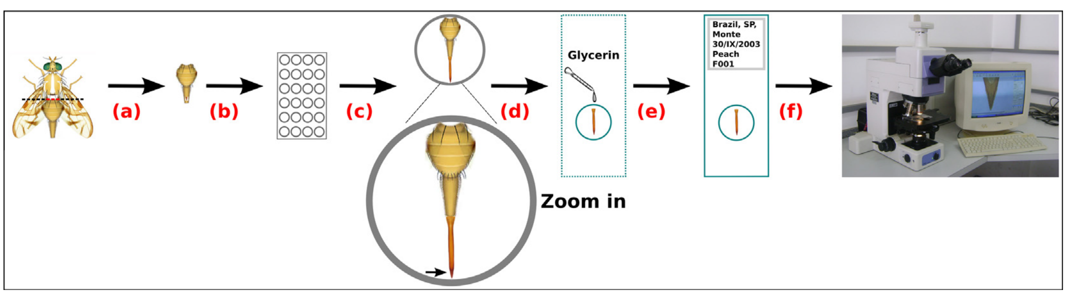


Fig. 4. The image acquisition process of aculei.

2.3. Learning techniques

This section presents some learning methods we use in this paper which are essential for a self-contained understanding of our work.

2.3.1. Decision tree (DT)

Decision tree is one of the learning techniques most intuitive that exists in the literature. It presents a simple and easy way to understand the classification process [16].

DT is composed of three kinds of nodes: root, internal, and leaf or terminal. A root node is the initial node that has zero or more outgoing edges (square in Fig. 5). Internal nodes are those that contain attributes (circles in Fig. 5). Finally, leaf nodes are the ones at the end of branches and define a class (triangles in Fig. 5) of a given input sample.

In this technique, two issues must be addressed [17]:

1. **The split policy for training records.** Typically, we use the entropy, impurity measure, (I) and the information gain of each attribute (Δ_{info}) to decide which attribute must be selected in the next recursive algorithm call.
2. **The stopping criteria for the splitting procedure.** One strategy could be the natural tree growing until each attribute is allocated in a single class. However, this strategy might result in trees that are too large to handle and lead to overfitting problems. Tree-pruning strategies are usually adopted to avoid these issues.

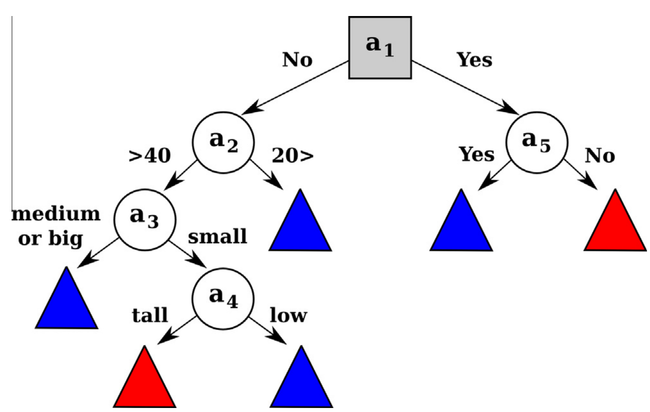


Fig. 5. A decision tree created for binary classification task (blue and red classes). Notice that a decision tree might be composed of different attributes (e.g., binary, ordinal, and continuous). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Eq. (1) shows the gain, where $I(.)$ is the impurity measure from the given node, N is the number of samples at the parent note, k is the number of attributes, and $N(a_j)$ is the number of sample associated with the node a_j. Eq. (2) shows the entropy, $p(i|a_j)$ denotes the fraction of samples from a class i at a given node a_j [17].

$$\Delta_{info} = I(parent) - \sum_{j=1}^{k} \frac{N(a_j)}{N} I(a_j) \tag{1}$$

$$Entropy(a_j) = -\sum_{i=0}^{c-1} p(i|a_j)\log_2 p(i|a_j) \tag{2}$$

Fig. 5 shows an example of decision tree, where attributes of samples from the used dataset are a_1, a_2, a_3, a_4, and a_5. The branches or edges are possible values for each attribute.

The most used algorithms in the literature are ID3 [18] e C4.5 [19].

2.3.2. k-nearest neighbors (kNN)

The k-nearest neighbor classifier is a technique based on the closest training examples in the feature space [20]. Eq. (3) illustrates an adjustment of *kNN* defined for *x*.

$$kNN(x) = \sum_{x_i \in \mathcal{N}_k(x)} y_i \tag{3}$$

where $\mathcal{N}_k(x)$ is the k-nearest neighbors from x in the training set, and y_i is a distance value among x and the current neighbor x_i (e.g., Euclidean distance). One common way to perform classification tasks might be deciding by majority voting of nearest neighbors. Fig. 6 shows an example of classification using kNN.

2.3.3. Naïve Bayes (NB)

Naïve Bayes is a simple probabilistic technique based on Bayes theorem to the problem of pattern classification. This technique assumes that the probability of each relevant attribute a_j is known and independent.

Eqs. (4) and (5) show the Bayes' formula, where $P(c_i)$ is the prior probability of the class c_i and $p(a_j|c_i)$ is a class-conditional probability density function, $p(a_j)$ is the probability density function for a_j given that the state of nature is c_i [21].

$$P(c_i|a_j) = \frac{p(a_j|c_i) \times P(c_i)}{p(a_j)} \tag{4}$$

$$p(a_j) = \sum_{i=1}^{k} p(a_j|c_i)P(c_i) \tag{5}$$

Eq. (6) shows an informal Bayes' formula from the Eq. (5).

$$Posterior = \frac{likelihood \times prior}{evidence} \tag{6}$$

2.3.4. Naïve Bayes Tree (NBT)

Naïve Bayes Tree is a hybrid technique that induces a decision tree and Naïve Bayes classifier. This technique has almost the same properties than decision trees (DT) with the additional Naïve Bayes (NB) classifier in the leaves for better deciding the class to which an input belongs. According to [22], NBT retains clean understanding of the techniques DT and NB and achieves better results in large databases.

2.3.5. Support Vector Machines (SVM)

Support Vector Machine is a machine learning method introduced in [23]. The goal is to construct an optimum hyperplane or set of hyperplanes, which can be used to separate an n-dimensional feature space. The hyperplane is calculated such that it maximizes the margin between two classes (the standard SVM is a two-class classifier). The margin can be seen as the minimum distance of one point of one class to the other. It can be interpreted as a separation measure between two classes and represents the separability degree between them (quality measure of classification). The points on borders between the classes are called support vectors.

When it is not possible to find a linear separator for the classes, the data are mapped on-the-fly onto higher dimensional spaces through a non-linear mapping using the kernel trick [24].

The important detail here is that SVMs can efficiently perform non-linear classification. The reason for choosing SVM in this work is that by using the kernel, SVMs gain flexibility in the choice of the form of the threshold separating the classes of interest, which do not need to be linear and even do not need to have the same functional form for all data. Also, SVMs deliver a unique solution, since the optimality problem is convex.

Fig. 7 illustrates the use of SVM to separate two classes. More details about this technique can be found in [23].

2.4. Multimodal and fusion techniques

According to Ross et al. [25], information fusion may be performed in four levels: sensor, feature, rank, and decision. Sensor level is the early stage of feature extraction, in which raw data are used to compose other richer data. This strategy has been widely used in biometric identification in which multiple images are combined to compose a single image with more information, as in a mosaicing scheme [26]. Feature level fusion is a strategy to handle coded data into a feature vector through some kind of description algorithm. This fusion might be a simple binding of different properties (e.g., color, texture, and shape) or more complex whether uses artificial intelligence techniques (e.g., evolutionary algorithms [27]). Rank level fusion is an approach that tries to combine different ranked lists. In Content-based Image Retrieval (CBIR) systems, rank level fusion might be used to

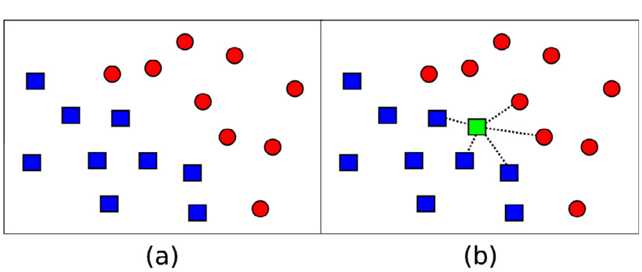


Fig. 6. (a) Samples of two classes (square and circle) in the features space. (b) Given a new object, its $k = 5$ nearest neighbors will define its class. In this case, the green point is labeled by the blue class. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

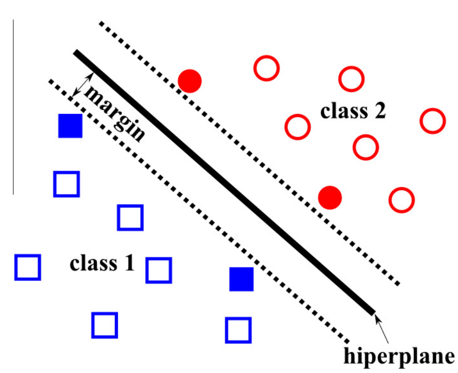


Fig. 7. The SVM classifier builds a maximum margin decision hyperplane to separate two classes.

combine ranked lists from different kinds of image descriptors (e.g., color and texture) and then produce a final rank or rank aggregation [28]. Finally, the decision-level fusion or late fusion that combines different opinions of the classifiers as confidence scores and labels. The simplest and most popular ways of combining classifiers are the majority and weighted voting approaches [29–31].

In the literature, several works have been proposed for fusion and they normally are called ensemble techniques as, for example, the well-known AdaBoost [32] and Bagging [33] approaches. AdaBoost and Bagging ensemble approaches have been used in several works in the literature due to their good results achieved in diverse applications. However, in several situations such methods have limitations in terms of efficiency, normalization, overfitting, and feature dimensionality problems. In [34], for instance, the authors have concerns with training time in Adaboost algorithms and large feature vector for face localization. In turn, in [35,36], the authors discuss the problems of feature normalization also in the context of combining classifiers.

The combination of multiple feature vectors or multimodal features in AdaBoost and Bagging approaches is usually based on their concatenation (feature binding). Normally, when performing feature binding of different nature/domain, normalization techniques should be applied to standardize all feature values in the same range which is, by itself, a hard problem.

Another common problem faced when features are combined (concatenated) refers to the "curse of dimensionality" [37]. The curse of dimensionality problem is related to the high-dimensional feature space that the available training instances become indistinguishable and not enough for allowing the definition of a good decision hyperplane [38]. In sense, our work adopted a classifier fusion framework [39] that is more robust to the aforementioned problems observed for other fusion approaches (e.g., curse of dimensionality and the time-consuming search for the most appropriate descriptors).

3. A framework for classifier selection and fusion

Given a classification problem, we have a set of image descriptors and a set of simple learning methods that will be used to learn patterns from available images in the training set. The important question then is how to automatically find the best classifiers and image properties and, more importantly, how to combine them to achieve the best possible classification results.

In this work, we denote a classifier as a tuple formed by a learning method and an image descriptor. Once we train all candidate classifiers, the learned knowledge might undergo a selection process of classifiers, which selects the most appropriate learning methods and descriptors to be combined by another learning method (meta-learning approach). The selection process aims at reducing the number of classifiers and keeping the effectiveness results as high as possible.

Those classifiers are selected in a selection process that uses diversity measures (Appendix A) calculated at training time through the use of a validation set. Diversity measures compute the degree of agreement/disagreement between involved classifiers pointing out the most interesting ones to be used in a combination scheme. The idea is that if two classifiers completely agree for all their outcomes, they do not complement each other and basically have the same opinion. The procedure is then to select complementary classifiers that although have different opinions still perform well in the classification problem. For instance, they correctly classify examples that one or the other incorrectly classifies. For more details regarding the selection process, please refer to Appendix B.

3.1. Formalization

Let C be the set of classifiers compounded by the combination of $\mathcal{L}$ learning methods and $\mathcal{F}$ image descriptors, where $|C| = |\mathcal{L}| \times |\mathcal{F}|$. Let $\mathcal{S}$ be a set of images (dataset) that will be divided into two parts, training (T) and validation (V) sets. Where $T \cup V = \mathcal{S}$ and $T \cap V = \emptyset$. As we consider a supervised learning scenario, the actual classes for training and validation data points are known *a priori*.

For all classifiers $c_j \in C$ $(1 < j \leqslant |C|)$ there is a training step in which the instances of the set T are used to learn patterns and then to predict each instance on the validation set V. These classifiers outcomes are stored into a matrix M_V, where $|M_V| = |V| \times |C|$, $|V|$ is the number of instances in a validation set V and $|C|$ is the number of classifiers.

In the next step, M_V is used as input of a selection process (Appendix B) that selects a set $C^* \subset C$ of classifiers to be combined by the meta-learning approach. This two-tier classification scheme aims at collecting classification information of each classifier in the first level and creating a new representation based on their answers to finally come up with the final classification answer. In this step, diversity measures (Appendix A) are employed to determine $|C^*|$ classifiers. Note that matrix M_V^* stores the results from the selection process of $|C^*|$ classifiers.

Given a new image I, we use each classifier from set C to determine the class of I, producing $|C|$ outcomes. The same set of classifiers C^* is used to create a matrix M_I^*, where $|M_I^*| = 1 \times |C^*|$. The matrix M_I^* is used as input of a fusion technique (meta-classifier) based on the Support Vector Machine (SVM) classifier that takes the final decision regarding the definition of the final class of I. For a more detailed description of selection process using diversity measures, the reader is referred to Appendix A and to our recent work [39]. Fig. 8 illustrates the framework for classifier selection and fusion.

3.2. Classifier multimodal fusion

Sometimes, some classification problems naturally have different modalities which can be combined to boost the classification results even further. For instance, for fruit fly classification, we have the wings (W) and aculei (A) modalities. It is reasonable to expect that although each one has good classification results as these modalities encompass different morphological features they can be complementary and devising an automatic way of detecting such complementarity and combining them is our objective. For that, we use such modalities as input to the framework previously described (Fig. 8, step (a)).

As output for step (a), two classification matrices encoding the first-layer classification are created, one for each modality ($M^*_{V_W}$ and $M^*_{V_A}$). $M^*_{V_W}$, *where* $|M^*_{V_W}| = |V| \times |C^*_W|$, is an outcome matrix and C^*_W is the set of classifiers that use Input$_W$. $M^*_{V_A}$, *where* $|M^*_{V_A}| = |V| \times |C^*_A|$ is an outcome matrix and C^*_A is the set of classifiers that use Input$_A$. Given a new instance I with those two modalities (W and A), two vectors are created as well ($M^*_{I_W}$ and $M^*_{I_A}$). Thereafter, a concatenation (merge process) of the columns of the matrices $M^*_{V_W}$ and $M^*_{V_A}$ is held and creates the matrix $M^*_{V_{WA}}$, *where* $|M^*_{V_{WA}}| = |V| \times (|C^*_W| + |C^*_A|)$.

Finally, $M^*_{V_{WA}}$ and $M^*_{I_{WA}}$ are used to feed the second-layer classification (meta level) SVM responsible for combining the different modalities and issuing the final decision for instance I (see Fig. 9).

4. Experimental protocol

This section shows the setup for each step introduced in Section 2.1 and presents the experimental methodology adopted to validate this work.

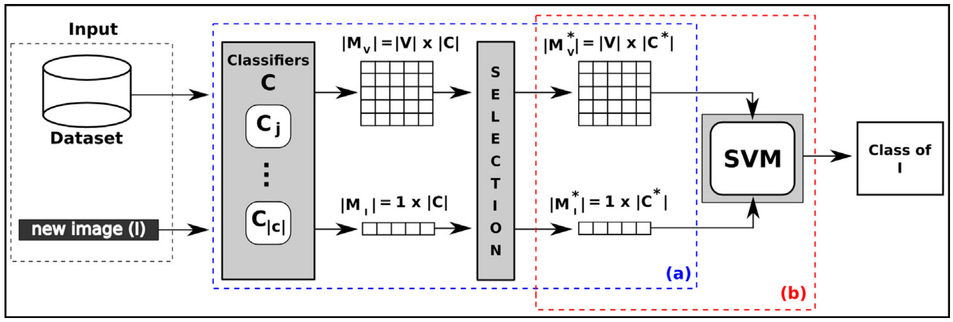


Fig. 8. Framework for classifier selection and fusion. Given a dataset, in (a), $|C|$ different classifiers are trained and the most appropriate $|C^*|$ classifiers are selected through the use of diversity measures in a selection process. In (b), $|C^*|$ classifiers are combined by an SVM technique.

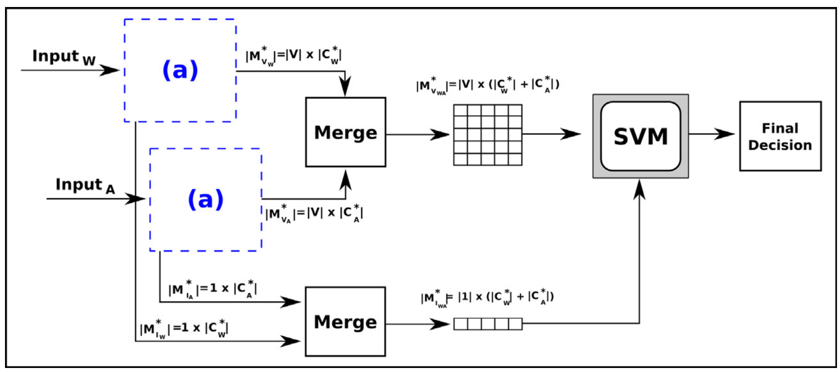


Fig. 9. Proposed extended framework for classifier multimodal fusion.

A. fraterculus *A. obliqua* *A. sororcula*

Fig. 10. Wings of the three species studied.

4.1. Dataset

We used two different datasets[1] in our experiments: (1) WING (Fig. 10) and (2) ACULEU (Fig. 11). Both datasets are composed of 249 images and divided into three different categories: *fraterculus* (83), *obliqua* (89), and *sororcula* (77). These categories have almost the same amount of images.

The datasets are composed of pictures of specimens reared from samples of fruit trees in experimental and commercial orchards in the state of São Paulo, Brazil, stored in the Department of Entomology and Acarology ESALQ, Piracicaba, SP, Brazil and in the Biological Institute, Campinas, SP, Brazil.

4.2. Pre-processing

The image pre-processing stage relies mainly on the image segmentation, a process responsible for classifying all pixels in an image as either object or background. Segmentation produces a binary map that encodes the separation between objects of interest and background. In this work, we used Otsu's method [40], which finds an optimum value (threshold) that minimizes intra-class variance of a gray image with the aim of separating object and background. Eq. (7) presents Otsu's equation.

$$\sigma^2_{within} = \omega_b(t)\sigma^2_b + \omega_o(t)\sigma^2_o, \tag{7}$$

where ω_b and ω_o are probabilities of the background and object classes separated by a threshold t, and σ^2_b and σ^2_o are variances of these classes.

The segmentation process is composed of four steps: (1) computation of Otsu's threshold from the input gray-level image; (2) creation of a binary image (Fig. 12b); (3) use of a dilation technique to increase the area of interest (Fig. 12c) and to close some holes within the target object (Fig. 13c); (4) matching of the original image with the dilated ones aiming at creating the final segmented image (Fig. 12d).

Figs. 12 and 13 show the result of each step of the segmentation approach using Otsu's method for a wing and an aculeus image, respectively.

4.3. Description

Acculei and wings naturally have different properties and we need different image descriptors to properly capture them for the optimal design of a classification system. In this context, here we

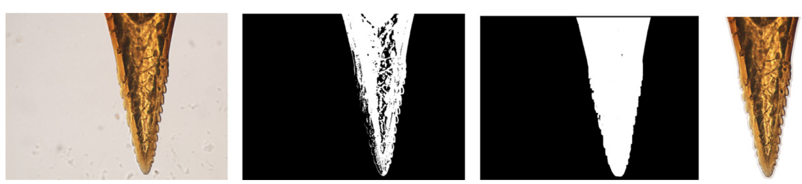
[1] The datasets will be freely available upon request after the acceptance of this paper.

A. fraterculus *A. obliqua* *A. sororcula*

Fig. 11. Aculeus of the three species studied.

(a) Original (b) Otsu (c) Dilation (d) Segmentation

Fig. 12. Images of each step of the pre-processing stage for the WING dataset.



(a) Original (b) Otsu (c) Dilation (d) Segmentation

Fig. 13. Images of each step of the pre-processing stage for the ACULEU dataset.

have considered color, texture, and shape image descriptors in order to characterize the images.

For color descriptors, we used Color Autocorrelogram (ACC) [41], Border/Interior Pixel Classification (BIC) [42], Color Coherence Vector (CCV) [43], Global Color Histogram (GCH) [44], and Local Color Histogram (LCH [44]. For texture descriptors we used Homogeneous Texture Descriptor (HTD) [45], Local Activity Spectrum (LAS) [46], and Quantized Compound Change Histogram (QCCH) [47]. Finally, as shape descriptor we used the Edge Orientation Autocorrelogram (EOAC) [48].

The criteria for choosing such image descriptors in this work are based on experiments described in [49] in which large experiments in several image retrieval tasks were performed in order to understand the behavior of different image characterization techniques and what they capture from an image.

4.4. Learning techniques

As previously discussed, in order to automate the classification process, we need to infer class properties from the input feature vectors described by various image characterization methods. As such, we employ machine learning classifiers which are, mathematically, a mapping from a feature space X to a discrete set of class labels Y (the fruit fly species). In this sense, here we have used six different simple learning methods: decision tree (DT), Naïve Bayes (NB), Naïve Bayes Tree (NBT), k-nearest neighbor (kNN) with $k = \{1, 3, 5\}$. Furthermore, we have used SVMs with normalized polynomial kernel to combine those simple learning methods in our fusion framework (see Section 3). The objective is to use simple classifiers that can provide different decision boundaries for the data and allow a better handling of difficult cases.

The implementation of the learning methods are available in the WEKA[2] data mining library. All learning methods were used with default parameters which means we did not optimize them whatsoever.

4.5. Identification

The last step of our method consists in using the previously learned classification models to classify a new input fruit fly example. This is done by using a framework for classifier selection and fusion that combines different image descriptors and learning methods through a meta-learning approach (see Section 3), our proposed two-tier classification scheme.

The outcome of such framework is the identity (labels) of the analyzed examples.

5. Experiments and results

In this section, we present results obtained with each simple classifier and all three fusion approaches that we propose herein (see Section 3). Section 5.1 discusses the impact of the segmenta-

[2] http://www.cs.waikato.ac.nz/ml/weka (As of April, 2014).

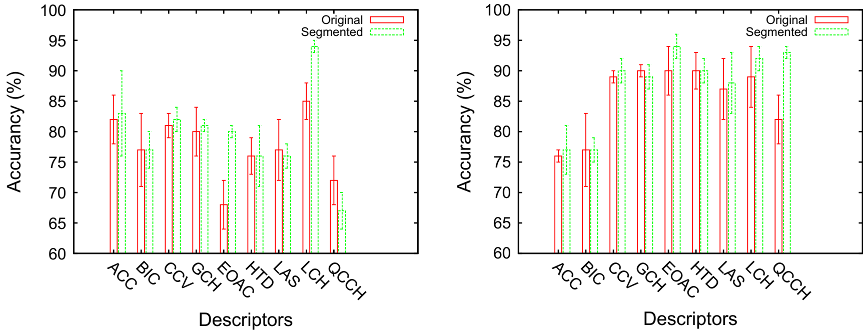


Fig. 14. Accuracy measures and their respective confidence intervals for classifiers based on the use of the SVM technique and different descriptors. Reported results consider the use of non-segmented (red bars) and segmented (green bars) images. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

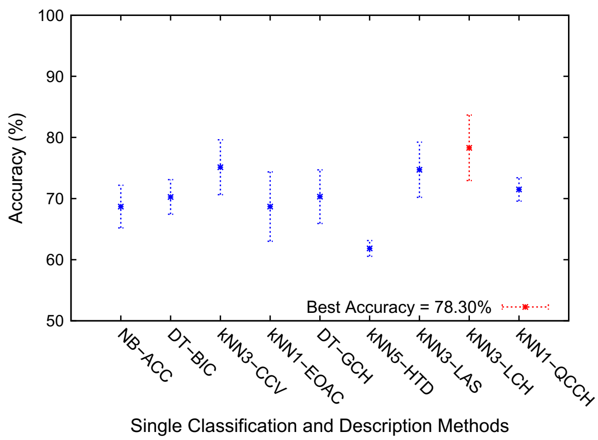


Fig. 15. The best classification results of each simple classifier using WING dataset.

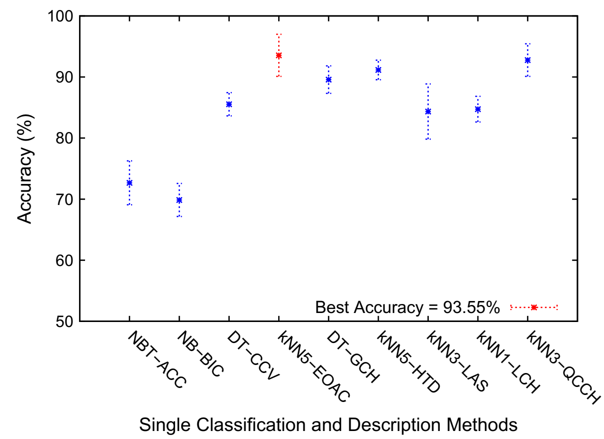


Fig. 16. The best classification results of each simple classifier using ACULEU dataset.

tion procedure employed, while Section 5.2 shows a detailed comparison of the best simple classifiers for each used image descriptor. Section 5.3 compares the classification performance of three fusion approaches and shows how the fusion outperforms any single method without it. Finally, Section 5.4 shows a detailed analysis between all approaches proposed in this work.

5.1. Impact of the segmentation process

In this experiment, we performed a study of the impact of the segmentation process in each one of the image descriptors used in this work. We used the most well-known learning technique in the literature, Support Vector Machine (SVM) and a 5-fold cross-validation protocol.

Fig. 14 depicts the obtained results for different descriptors using the WING (a) and ACULEU (b) datasets. In Fig. 14a, two visual properties must be highlighted: (1) EOAC is a shape descriptor that has a great classification gain, from 67% to almost 80%; (2) QCCH is a texture descriptor and it is a single descriptor that has a loss of performance from 72% to 67%.

As can be observed, the color and shape descriptors were able to take advantage of the segmentation process. However, the texture descriptors did not take advantage of segmentation, in special, the QCCH descriptor. A possible explanation for this phenomenon might be related to padding approaches [50,51], in which values (zero or one) are assigned to pixels not relevant in the image (in our case, background) in the segmentation process. In this process, white gaps present in the WINGS images (see Fig. 12d) are considered on the texture features extraction process and thus affect the encoding task. This problem might be partially solved with the use of local descriptors and *bag-of-visual-words* (BOW), but this is out of the scope of this work [52,53]. In Fig. 14b, all image descriptors achieved better results when the segmentation process has been

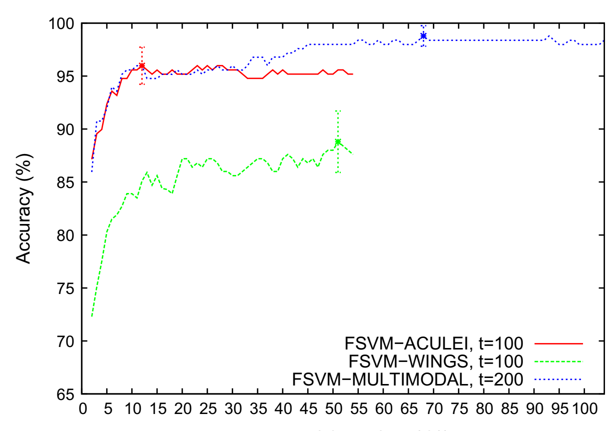


Fig. 17. Classification results of all approaches with different number of classifiers (single methods C^*_A (*FSVM − ACULEI*) and C^*_W (*FSVM − WINGS*), and fusion method C^*_{WA} (*FSVM − MULTIMODAL*)). The highlighted points are the best results of each fusion approach with their confidence intervals.

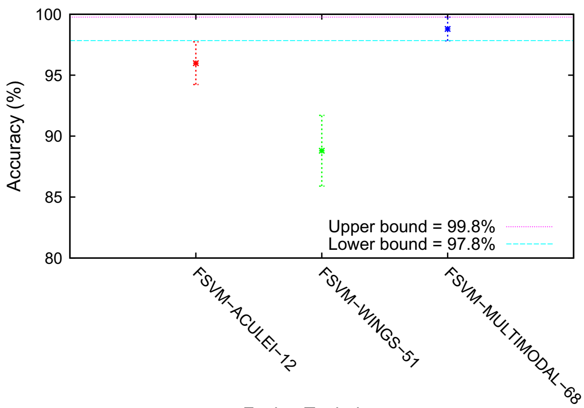


Fig. 18. The best classification results of each fusion approach.

Table 1
Classification results (accuracy) for each approach in different folds. The best results are highlighted in bold.

Folds	Approaches		
	FSVM-ACULEI-12	FSVM-WINGS-51	FSVM-MULTIMODAL-68
1	95.92	91.84	**97.96**
2	98.00	88.00	**100.00**
3	**98.00**	88.00	**98.00**
4	94.00	92.00	**98.00**
5	94.00	84.00	**100.00**
Average	95.98	88.77	**98.79**
Conf. intervals	1.75	2.90	0.97

used. In ACULEU images, there are no white gaps inside the object that may hinder the extraction process.

5.2. Classification results with single classification and description methods

This section compares the classification performance of the best learning method (Section 4.4) for each image descriptor (Section 4.3) using two different datasets (WING and ACULEU).

Fig. 15 depicts the best classification accuracy results achieved for each image descriptor using a 5-fold cross-validation protocol and WING dataset. Notice that the best accuracy result was 78.30% with the classifier kNN3-LCH, i.e., the image color descriptor LCH and the learning method kNN with $k = 3$.

Fig. 16 depicts the best accuracy results achieved for each image descriptor using a 5-fold cross-validation protocol and ACULEU dataset. Notice that the best accuracy result was 93.55% with the classifier kNN5-EOAC, i.e., the image shape descriptor EOAC and the learning method kNN with $k = 5$.

Although the previous experiments showed good classification results, both experiments need to seek the optimum learning method and image descriptor that achieve the best classification accuracy results in each target application domain.

In the next section, we show how to achieve the best results with no concern with the optimum learning method and image descriptor automatically. Our approach first explores the complementarity of different characterization and learning methods to boost the classification results. Then, we also introduce a method for combining the different modalities (aculeus and wings) towards a more robust and powerful fruit fly classification system.

5.3. Boosting the classification results through feature, classifier and modality fusion

This section describes a behavioral analysis of the extended framework (Section 3) that combines different numbers of classifiers $|C^*|$ with three modalities (ACULEI, WINGS, and MULTIMODAL) through a meta-learning SVM approach. Fig. 17 depicts the mean accuracy results achieved for each fusion approach using a 5-fold cross-validation protocol.

For experiments that use only a single modality (ACULEI and WINGS), we have $|C^*_A| = |C^*_W| = \{2, \ldots, 54\}$ (eight image descriptors $\times$ six learning methods). Note the presence of two fusion approaches here: one exploring different classifiers and descriptors on top of aculei images; and one fusion approach on top of wings images. In multimodal experiments, we have $|C^*_{WA}| = \{2, \ldots, 108\}$ which means we merge both sets ($|C^*_A|$ and $|C^*_W|$) in the fusion process considered. In this case, there is a fusion approach taking advantage of different image descriptors, classifiers and also modalities (aculei and wings).

Table 2
Classification results (accuracy) of each proposed method per class in different folds. The superiority of the multimodal fusion is apparent.

Folds	Classes	FSVM-ACULEI-12	FSVM-WINGS-51	FSVM-MULTIMODAL-68
1	fraterculus	**100.00**	87.50	**100.00**
	obliqua	94.44	**100.00**	94.44
	sororcula	93.33	86.67	**100.00**
2	fraterculus	**100.00**	88.89	**100.00**
	obliqua	**100.00**	87.50	**100.00**
	sororcula	93.75	87.50	**100.00**
3	fraterculus	**100.00**	75.00	**100.00**
	obliqua	94.12	**100.00**	94.12
	sororcula	**100.00**	88.24	**100.00**
4	fraterculus	**93.75**	**93.75**	**93.75**
	obliqua	**100.00**	95.00	**100.00**
	sororcula	85.71	85.71	**100.00**
5	fraterculus	**100.00**	70.59	**100.00**
	obliqua	88.89	94.44	**100.00**
	sororcula	93.33	86.67	**100.00**
Average	fraterculus	**98.75**	83.15	**98.75**
	obliqua	95.49	95.39	**97.71**
	sororcula	93.23	86.96	**100.00**
Conf. intervals	fraterculus	2.45	8.64	2.45
	obliqua	4.09	4.51	2.75
	sororcula	4.44	0.84	0.00

In these experiments, we can notice that the framework with ACULEI classifiers (FSVM-ACULEI) has similar results to FSVM-MULTIMODAL in almost all numbers of classifiers in the selection process between the ranges 2–28 classifiers. However, the framework with multimodal classifiers (FSVM-MULTIMODAL) is better from $|C^*| = 29$ on and the difference between FSVM-ACULEI and FSVM-MULTIMODAL approaches has a considerable increase after $|C^*| = 33$ with the latter clearly outperforming the first. Although the framework using FSVM-WINGS classifiers achieves the worst results of three approaches in all numbers of classifiers, we can observe a high confidence interval (±2.9) on the position $|C^*| = 51$. In addition, it is worth noting that we found no automatic solution in the literature so far that presented such high classification results for any method based on aculei or wings.

The best result for each fusion approach is highlighted in Fig. 17. FSVM-ACULEI (96.0%), FSVM-WINGS (88.8%), and FSVM-MULTIMODAL (98.8%) using 12, 51, and 68 classifiers, respectively.

Fig. 18 shows the best results achieved for each approach and their confidence intervals. Notice that there is no overlap between any approach proposed in this work. For best viewing, two strips (upper and lower bounds) has been placed.

5.4. Fine-grained analysis of the classification results

In this section, a more detailed analysis of the best classification results of each fusion approach (FSVM-ACULEI-12, FSVM-WINGS-51, and FSVM-MULTIMODAL-68).

Table 1 shows classification accuracy results for all approaches using the 5-fold cross-validation protocol. The best results are highlighted in bold. As can be observed, the multimodal fusion approach outperforms the other fusion methods regardless the data division/sampling and also presents the lower confidence interval (more strict and reliable classification decisions).

Table 2 shows the classification accuracy results per class for all approaches using the 5-fold cross-validation protocol. The best results are highlighted in bold. As can be observed, the multimodal approach once again achieves the best results for almost all folds with lower confidence interval for all classes confirming its superiority with respect to the other fusion and single methods.

6. Conclusions and future work

In this study, we performed several experiments with different image description and learning methods to develop a sounding understanding and basis for the design and deployment of a fruit fly recognition system considering the *fraterculus* complex. In these experiments, two datasets were used (WING and ACULEU), three approaches (FSVM-WINGS, FSVM-ACULEI, and FSVM-MULTIMODAL) have been proposed and different behaviors noticed. The first two fusion methods explore different classification and image description methods for identifying fruit flies. They explore complementary properties both in the level of description (e.g., color, texture, and shape) and in the higher classification level (different forms for carving the decision boundaries). However, it is with the third proposed method that the automatic classification shines. The FSVM-MULTIMODAL fusion explores, in addition to classification and description, the power of modality fusion (wings and aculei) to boost the classification results even further and to provide a much more reliable automatic classification method for the problem.

In the performed experiments, we showed that FSVM-MULTIMODAL approach yields an impressive 98.8% classification accuracy, against 88.8% and 96.0% of FSVM-WINGS and FSVM-ACULEI approaches, respectively. This means that when it is possible to employ different modalities to solve this problem, FSVM-MULTIMODAL represents a remarkable classification error reduction in 90% when compared to FSVM-WINGS and 70% when compared to FSVM-ACULEI. The fusion methods also allowed a huge improvement over their single characterization and classification counterparts that do not use fusion. For instance, the FSVM-MULTIMODAL reduces the classification error of the best aculei-based classification method (kNN5-EOAC) in 82%, a remarkable improvement.

These experiments confirm that it is possible to achieve good classification results in fruit fly identification tasks using the two kinds of images (wings and aculei), as well as, their combination. We also demonstrate that the automatic identification of these *Anastrepha* species based on image analysis and learning techniques is a suitable alternative to traditional laborious and error-prone methods currently employed. In fact, the classification accuracy levels obtained in this work are superior to morphological identification by experienced entomologists. Usually, those experts are not specialists in the genus *Anastrepha*, and therefore use published documentation and reference collections to perform identifications. Finally, we showed that the extended fusion framework might be a good solution to support reliable identification within the *fraterculus* group of species, which most specialist would not have deemed possible before this work.

Although exemplified herein for combining wings and aculei modalities for fruit fly classification, it is worth mentioning that this new fusion methodology we proposed herein can be used in many other classification problems with potential positive impact for applications that require complementary properties to be addressed (e.g., object recognition, scene classification, and face recognition).

Future work will investigate other species of the genus *Anastrepha*, groups of Tephritidae, image descriptors, learning methods, and strategies for feature combination as well as other classifier and modality fusion policies. Also, we intend to develop a computer aided system that can be used by non-specialist researchers in the Biology domain, who need to perform species identification in their daily work.

Acknowledgments

We thank the financial support of the FAPESP (Grants 2010/14910-0, 2010/05647-4, 2009/54806-0, and 98/05085-2), CNPq (Grants 303726/2009-1, 550890/2007-6, 309618/2010-0, and 304352/2012-8), CAPES (Grant 1260–12-0), AMD and Microsoft. We also thank Miguel Francisco Souza Filho for the samples of *Anastrepha*, and Heraldo Negri de Oliveira for the fruit fly photo.

Appendix A. Diversity measures

Diversity is the degree of agreement/disagreement between involved classifiers pointing out the most interesting ones to be further used in a combination scheme. To achieve this diversity score or quantitative value inside ensemble systems, we have explored diversity measures considering pairs of classifiers [55,56].

Let $\mathcal{M}$ be a matrix 2×2 containing the relationship between a pair of classifiers with percentage of agreement. This relationship matrix $\mathcal{M}$ has the percentage of hit and miss for each classifier c_i and c_j.

The value a is the percentage of images that both classifiers c_i and c_j classified correctly in the validation set. Values b and c are the percentage of images that c_j classified correctly but c_i missed and vice versa. The value d is the percentage of images that both classifiers missed.

In [55], Kuncheva et al. presented several measures to assess diversity, considering pairs of classifiers. Following their work, in our experiments, we have used *Correlation Coefficient p* (*COR*),

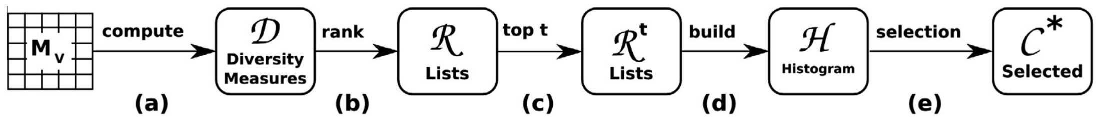


Fig. B.19. The five steps for classifier selection are: (a) computation of diversity measures from the validation matrix M_V; (b) ranking of pairs of classifiers by their diversity measures scores; (c) selection of the top t ranked pairs of classifiers; (d) computation of a histogram H that counts the number of occurrences of each classifier; (e) select the most appropriate classifiers $|\mathcal{C}^*|$ based on their occurrence in H and satisfy a defined threshold $\mathcal{T}$.

Double-Fault Measure (*DFM*), *Disagreement Measure* (*DM*), *Interrater Agreement k* (*IA*), and *Q-Statistic* (*QSTAT*). Those measures are defined as follows:

$$COR(c_i, c_j) = \frac{ad - bc}{\sqrt{(a+b)(c+d)(a+c)(b+d)}}, \tag{A.1}$$

$$DFM(c_i, c_j) = d, \tag{A.2}$$

$$DM(c_i, c_j) = \frac{b+c}{a+b+c+d}. \tag{A.3}$$

$$QSTAT(c_i, c_j) = \frac{ad - bc}{ad + bc}, \tag{A.4}$$

$$IA(c_i, c_j) = \frac{2(ac - bd)}{(a+b)(c+d) + (a+c)(b+d)}. \tag{A.5}$$

The diversity is greater if the measures *Double-Fault Measure*, *Q-Statistic*, *Interrater Agreement k*, and *Correlation Coefficient p* are lower among pairs of classifiers c_i and c_j. In the case of the *Disagreement Measure*, the greater the measure, the greater the diversity [55,56]. Ranges of *COR*, *QSTAT*, and *IA* are in $[-1, 1]$ while *DFM* and *DM* are in $[0, 1]$.

Appendix B. Classifier selection

Fig. B.19 illustrates the adopted five-step approach for selecting classifiers based on diversity measures, previously introduced in [39].

First, diversity measures (set $\mathcal{D}$ in Fig. B.19) are used to assess the degree of agreement among available classifiers in $\mathcal{C}$ by taking into account the M_V matrix previously computed. That step is represented by arrow (a) in Fig. B.19. Pairs of classifiers are then ranked according to their diversity score. Each diversity measure defines a different ranked list and, at the end of this step, a set $\mathcal{R}$ of ranked lists is produced (arrow (b)). In the following, a novel set of ranked lists $\mathcal{R}^t$ is computed by selecting the top t pairs of classifiers from each ranked list in $\mathcal{R}$ (arrow (c)), and a histogram H that counts the number of occurrences of a classifier in all ranked lists of $\mathcal{R}^t$ is computed (arrow (d)). Finally, the most frequent classifiers in H, whose accuracy is greater that a given threshold $\mathcal{T}$, are combined by a fusion approach (arrow (e)). $\mathcal{T}$ is a threshold defined in terms of the average accuracy among all classifiers using the validation set V.

References

[1] A.L. Norrbom, C.A. Korytkowski, R.A. Zucchi, K. Uramoto, G.L. Venable, J. McCormick, M.J. Dallwitz, *Anastrepha* and *Toxotrypana*: Descriptions, Illustrations, and Interactive Keys, 2012.
[2] M. Aluja, Bionomics and management of *Anastrepha*, Annu. Rev. Entomol. 39 (1994) 155–178.
[3] A.L. Norrbon, R.A. Zucchi, V. Hernandez-Ortiz, Phylogeny of the genera *Anastrepha* and *Toxotrypana* (Trypetinae: Toxotrypanini) based on morphology, in: Fruit Flies (Tephritidae): Phylogeny and Evolution of Behavior, CRC Press, Boca Raton, 1999, pp. 299–342.
[4] R.A. Zucchi, Fruit Flies in Brazil: *Anastrepha* Species and Their Host Plants and Parasitoids, URL <http://www.lea.esalq.usp.br/anastrepha/>, 2008.
[5] M.R.B. Smith-Caldas, B.A. McPheron, J.G. Silva, R.A. Zucchi, Phylogenetic relationships among species of the *fraterculus* group (*Anastrepha*: Diptera: Tephritidae) inferred from DNA sequences of mitochondrial cytochrome oxidase I, Neotrop. Entomol. 30 (2004) 565–573.
[6] D. Selivon, A. Perondini, J. Morgante, A genetic-morphological characterization of two cryptic species of the *Anastrepha fraterculus* complex (Diptera: Tephritidae), Ann. Entomol. Soc. Am. 98 (2005) 367–381.
[7] V. Hernández-Ortiz, A.F. Bartolucci, P. Morales-Valles, D. Frías, D. Selivon, Cryptic species of the *Anastrepha fraterculus* complex (Diptera, Tephritidae): a multivariate approach for the recognition of South American morphotypes, Ann. Entomol. Soc. Am. 105 (2012) 305–318.
[8] J.I. Arribas, G.V. Sánchez-Ferrero, G. Ruiz-Ruiz, J. Gómez-Gil, Leaf classification in sunflower crops by computer vision and neural networks, Comput. Electron. Agric. 78 (1) (2011) 9–18.
[9] A. Pourreza, H. Pourreza, M.-H. Abbaspour-Fard, H. Sadrnia, Identification of nine Iranian wheat seed varieties by textural analysis with image processing, Comput. Electron. Agric. 83 (2012) 102–108. ISSN 0168-1699.
[10] M. Wiwart, E. Suchowilska, W. Lajszner, L. Graban, Identification of hybrids of spelt and wheat and their parental forms using shape and color descriptors, Comput. Electron. Agric. 83 (2012) 68–76. ISSN 0168-1699.
[11] J.R. Wooten, S.D. Filip To, C. Igathinathane, L.O. Pordesimo, Discrimination of bark from wood chips through texture analysis by image processing, Comput. Electron. Agric. 79 (1) (2011) 13–19. ISSN 0168-1699.
[12] A.T. Watson, M.A. O'Neill, I.J. Kitching, A qualitative study investigating automated identification of living macrolepidoptera using the Digital Automated Identification SYstem (DAISY), Syst. Biodivers. 1 (2003) 287–300.
[13] K. Russell, M. Do, J. Huv, N. Platnick, Introducing SPIDA-web: wavelets, neural networks and Internet accessibility in an image-based automated identification system, Syst. Assoc. 74 (2007) 131–152.
[14] T. Arbuckle, S. Schrder, V. Steinhage, D. Wittmann, Biodiversity informatics in action: identification and monitoring of bee species using ABIS, in: International Symposium for Environmental Protection, 2001, pp. 425–430.
[15] V. Hernndez-Ortiz, J.A. Gmez-Anaya, A. Snchez, B.A. McPheron, M. Aluja, Morphometric analysis of Mexican and South American populations of the *Anastrepha fraterculus* complex (Diptera: Tephritidae) and recognition of a distinct Mexican morphotype, Bull. Entomol. Res. (94) (2004) 487–499.
[16] S. Lomax, S. Vadera, A survey of cost-sensitive decision tree induction algorithms, ACM Comput. Surv. 45 (2) (2013) 16:1–16:35. ISSN 0360-0300.
[17] P. Tan, M. Steinbach, V. Kumar, Introduction to Data Mining, first ed., Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2005.
[18] J.R. Quinlan, Induction of decision trees, Mach. Learn. 1 (1) (1986) 81–106.
[19] J.R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.
[20] J. Friedman, T. Hastie, R. Tibshirani, The Elements of Statistical Learning, first ed., Springer, 2001.
[21] R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, second ed., Wiley-Interscience, 2000.
[22] R. Kohavi, Scaling up the accuracy of Naive-Bayes classifiers: a decision-tree hybrid, in: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, AAAI Press, 1996, pp. 202–207.
[23] B.E. Boser, I.M. Guyon, V.N. Vapnik, A training algorithm for optimal margin classifiers, in: Proceedings of the Fifth Annual Workshop on Computational Learning Theory, COLT '92, 1992, pp. 144–152.
[24] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines and Other Kernel-based Learning Methods, Cambridge University Press, 2000.
[25] A.A. Ross, K. Nandakumar, A.K. Jain, Handbook of Multibiometrics (International Series on Biometrics), Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. ISBN 0387222960.
[26] X. Xia, L.O. Gorman, Innovations in fingerprint capture devices, Pattern Recogn. (PR) 36 (2) (2003) 361–369.
[27] R. da S. Torres, A.X. Falcão, M.A. Gonçalves, J.P. Papa, B. Zhang, W. Fan, E.A. Fox, A genetic programming framework for content-based image retrieval, Pattern Recogn. (PR) 42 (2) (2009) 283–292.
[28] D.C.G. Pedronette, R. da S. Torres, Exploiting contextual spaces for image re-ranking and rank aggregation, in: ACM International Conference on Multimedia Retrieval (ICMR), 2011, pp. 13:1–13:8.
[29] M.P. Perrone, L.N. Cooper, When Networks Disagree: Ensemble Methods for Hybrid Neural Networks, Chapman and Hall, 1993.
[30] John B. Hampshire, A. Waibel, The Meta-Pi network: building distributed knowledge representations for robust multisource pattern recognition, IEEE Trans. Pattern Anal. Machine Intell 7 (14) (1992) 751–769.
[31] J.A. Benediktsson, J.R. Sveinsson, O.K. Ersoy, P.H. Swain, Parallel consensual neural networks, IEEE Trans. Neural Networks 8 (1) (1997) 54–64.
[32] Y. Freund, R.E. Schapire, Experiments with a new boosting algorithm, Int. Conf. Mach. Learn. (ICML) (1996) 148–156.
[33] L. Breiman, Bagging predictors, Mach. Learn. 24 (2) (1996) 123–140.
[34] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features, in: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2001, pp. I-511–I-518.

[35] A. Rocha, D.C. Hauagge, J. Wainer, S. Goldenstein, Automatic fruit and vegetable classification from images, Elsevier Comput. Electron. Agric. (COMPAG) 70 (1) (2010) 96–104.
[36] A. Rocha, J.P. Papa, L.A.A. Meira, How far do we get using machine learning black-boxes?, Int J. Pattern Recognit. Artif. Intell. 26 (2) (2012) 1261001-1–1261001-23.
[37] R. Weber, H. Schek, S. Blott, A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces, in: Proceedings of the International Conference on Very Large Data Bases (VLDB), 1998, pp. 194–205.
[38] C.M. Bishop, Pattern Recognition and Machine Learning, first ed., Springer, 2006.
[39] F.A. Faria, J.A. dos Santos, A. Rocha, R. da S. Torres, A framework for selection and fusion of pattern classifiers in multimedia recognition, Pattern Recognit. Lett. 39 (0) (2014) 52–64. ISSN 0167-8655, <http://www.sciencedirect.com/science/article/pii/S0167865513002870>.
[40] N. Otsu, A threshold selection method from gray-level histograms, systems, IEEE Trans. Man Cybernet. 9 (1) (1979) 62–66.
[41] J. Huang, R. Kumar, M. Mitra, W. Zhu, R. Zabih, Image indexing using color correlograms, in: Conference on Computer Vision and Pattern Recognition, 1997, pp. 762–768.
[42] R. Stehling, M. Nascimento, A. Falcao, A compact and efficient image retrieval approach based on border/interior pixel classification, in: International Conference on Information and Knowledge Management, 2002, pp. 102–109.
[43] G. Pass, R. Zabih, J. Miller, Comparing images using color coherence vectors, in: ACM Multimedia, 1996, pp. 65–73.
[44] M. Swain, D. Ballard, Color indexing, Int. J. Comput. Vis. 7 (1) (1991) 11–32.
[45] B.S. Manjunath, W.Y. Ma, Texture features for browsing and retrieval of image data, IEEE Trans. Pattern Anal. Mach. Intell. 18 (8) (1996) 837–842.
[46] B. Tao, B. Dickinson, Texture recognition and image retrieval using gradient indexing, J. Vis. Commun. Image Representat. 11 (3) (2000) 327–342.
[47] C. Huang, Q. Liu, An orientation independent texture descriptor for image retrieval, in: International Conference on Computational Science, 2007, pp. 772–776.
[48] F. Mahmoudi, J. Shanbehzadeh, A. Eftekhari-Moghadam, H. Soltanian-Zadeh, Image retrieval based on shape similarity by edge orientation autocorrelogram, Pattern Recognit. 36 (8) (2003) 1725–1736.
[49] O.A.B. Penatti, E. Valle, R. da S. Torres, Comparative study of global color and texture descriptors for web image retrieval, J. Vis. Commun. Image Representat. 23 (2) (2012) 359–380.
[50] Z. Li, R.F. Hayward, J. Zhang, H. Jin, R.A. Walker, Evaluation of spectral and texture features for object-based vegetation species classification using support vector machines, in: ISPRS Technical VII Symposium: 100 Years ISPRS, ISPRS, Vienna, Austria, 2010, pp. 122–127.
[51] Y. Liu, D. Zhang, G. Lu, W.-Y. Ma, Study on texture feature extraction in region-based image retrieval system, in: Multi-Media Modelling Conference Proceedings, 2006 12th International, 2006, 8 pp.
[52] K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors, IEEE Trans. Pattern Anal. Mach. Intell. 27 (10) (2005) 1615–1630.
[53] J. dos Santos, O. Penatti, R. da S. Torres, P. Gosselin, S. Philipp-Foliguet, A. Falcão, Improving texture description in remote sensing image multi-scale classification tasks by using visual words, in: Pattern Recognition (ICPR), 2012 21st International Conference on, 2012, pp. 3090–3093.
[54] Plantwise, Empowering Farmers, Powering Research – Delivering Improved Food Security, URL <http://www.plantwise.org/> (accessed on October 30th, 2013).
[55] L.I. Kuncheva, C.J. Whitaker, Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy, Mach. Learn. 51 (2003) 181–207.
[56] L.I. Kuncheva, Combining Pattern Classifiers: Methods and Algorithms (Kuncheva, L.I.; 2004) [book review], vol. 18, 2007.