A Relevance Feedback Method based on Genetic Programming for Classification of Remote Sensing Images

J. A. dos Santos, C. D. Ferreira, R. da S. Torres *,

Institute of Computing, University of Campinas, Campinas, SP, Brazil

M. A. Gonçalves

Department of Computer Science, Federal University of Minas Gerais, Belo Horizonte, MG, Brazil

R. A. C. Lamparelli

Center for Research in Agriculture, Univesity of Campinas, Campinas, SP, Brazil

---

Abstract

This paper presents an interactive technique for remote sensing image classification. In our proposal, users are able to interact with the classification system, indicating regions of interest (and those which are not). This feedback information is employed by a genetic programming approach to learn user preferences and combine image region descriptors that encode spectral and texture properties. Experiments demonstrate that the proposed method is effective for image classification tasks and outperform some recent and effective as well as traditional baselines for the problem.

Key words:

content-based image retrieval, region descriptors, relevance feedback, genetic programming, remote sensing image classification

1 Introduction

Brazilian agriculture has obtained efficient, competitive, and dynamic results. In the last decade, agriculture has increased its contribution to the Brazilian Gross Domestic Product (GDP), representing around 10% of the total GDP. In this scenario, there is a huge demand for information systems to support monitoring and planning of agriculture activities in Brazil. One of the most used approaches for crop monitoring is based on the use of Remote Sensing Images (RSIs).

RSIs provide the basis for the creation of information systems that support the decision-making process based on soil occupation changes. In these systems, two important issues need to be addressed: how to identify (recognize) regions of interest and, later, how to extract/define polygons around these regions.

The identification and polygon extraction tasks usually rely on applying classification strategies that exploit visual aspects related to spectral and texture patterns identified in RSI regions. These tasks can be done automatically or manually.

The "manual" approach is based on image editors with which users can define or

* Corresponding author. Address: Av. Albert Einstein, 1251, CEP 13084-851 Campinas, SP, Brazil. Tel +55 19 3521-5887; Fax +55 19 3521-5847.

Email addresses: jefersson@lis.ic.unicamp.br (J. A. dos Santos), crferreira@lis.ic.unicamp.br (C. D. Ferreira), rtorres@ic.unicamp.br (R. da S. Torres), mgoncalv@dcc.ufmg.br (M. A. Gonçalves), rubens@cpa.unicamp.br (R. A. C. Lamparelli).

draw polygons that represent regions of interest using the raster image as background. The extraction of polygons from raster images is called *vectorization*.

In general, automatic approaches use classication strategies based on pixel information [1]. The main drawback of these approaches is concerned with its sensitivity to noise in the images (for example, distortions that can be found in mountainous regions). Another important problem in the automatic approaches is concerned with the fact that they usually fail to correctly identify borders between distinct regions within the same image. Thus, in practice, the results obtained need to be manually revised. As these revisions may take a lot of time, it is sometimes more convenient to the user to perform recognition manually.

This paper addresses these shortcomings by presenting a semi-automatic approach for RSI classification. The proposed solution relies on the use of an interactive strategy, called *relevance feedback* [2], based on the idea that a classification system can learn which are the regions of interest. The proposed image classification process with relevance feedback is comprised of four steps: (i) showing a small number of retrieved image regions to the user; (ii) user indication of relevant and non-relevant regions; (iii) learning the user needs from her feedback; (iv) and selecting a new set of regions to be shown. This procedure is repeated until a satisfactory result is reached. Notice that compared to other manual or semi-automatic approaches, ours only require the user to identify relevant (or irrelevant) regions, being potentially very easy to use.

In this paper recently proposed relevance feedback methods for interactive image search [3,4] are extended and adapted for image classification, more specifically for RSI classification. This method adopts a genetic programming approach to learn user preferences in a query session. Genetic programming (GP) [5] is a Machine

Learning technique used in many applications, such as data mining, signal processing, and regression [6,7,8]. This technique is based on the evolution theory and aims to find near optimal solutions. The use of GP is motivated in this work by the previous success of using this technique in information retrieval [7] and Content-Based Image Retrieval (CBIR) [9] tasks.

In [3] a RF approach which exploits the indication of relevant (positive) images are introduced. This method was extended in [4] to deal with image region features. The main objective was to use GP to find a function that combines the region similarity values (instead of global features, as presented in [3]) computed by different descriptors, and then learn the user needs. In this paper, we extend both approaches for a new application: classification of remote sensing images. Furthermore, we discuss how to incorporate the user indication of non-relevant regions in the relevance feedback process, an issue not explored in both previous works. These extensions are original contributions of this work.

This article is organized as follows. Section 2 covers related work. Section 3 presents the background concepts necessary to understand our proposed approach. Section 4 introduces our region-based similarity model using GP, including the extensions required to incorporate negative feedback. Section 5 discusses how the proposed model is applied to the problem of RSI vectorization. Section 6 describes our experimental evaluation and is follwed by Section 7 that concludes the paper.

2 Related Work

2.1 Classification of RSIs

Images provided by satellite sensors have been used in large scale for crop monitoring and production predictions. However, there is not a satisfactory fuly automatic method to classify RSIs so far. Terrain distortions and the interference of clouds, for example, make classification a hard problem. Another issue is to provide effective classification strategies considering the different evolution stages of a crop. Traditional classification methods are based on pixel analysis. The most used pixel classification algorithm, MaxVer [1], however, is not very effective. Several new methods have been proposed to improve the performance of MaxVer-based techniques. In [10], a new method considering image segmentation, GIS, and data mining algorithms was presented. Compared with pixel-based classification, the results showed best agreement with visual interpretation. The work proposed in [11] applied a morphological filter in an image which was classified by MaxVer algorithm. The results were compared with the other classification algorithms (Fisher linear likelihood, minimum Euclidean distance and ECHO). In [12], three Land Cover Classification Algorithms are compared for monitoring North Korea using multi-temporal data.

2.2 CBIR and Relevance Feedback

CBIR systems provide efficient and effective means to retrieve images. In these systems, the searching process consists in, for a given image, computing the most similar images stored in the database. The searching process relies on the use of image *descriptors*. A descriptor can be characterized by two functions: *feature vector extraction* and *similarity computation*. The feature vectors encode image properties, like color, texture, and shape. Therefore, the similarity between two images is computed as a function of their feature vectors distance.

In some CBIR approaches the descriptors are statically combined, that is, the descriptors composition is fixed and used in all retrieval sessions. Nevertheless, different people can have distinct visual perception of a same image. Motivated by this limitation, *relevance feedback* approaches were incorporated into CBIR systems [13,14,15]. This technique makes possible the user interaction with the retrieval systems.

Relevance feedback (RF) [13,14,15,16] is a technique initially proposed for document retrieval that has been used with great success for human-computer interaction in CBIR. RF addresses two questions referring to CBIR process. The first one is the semantic gap between high-level visual properties of images and low-level features used to describe them. Usually, it is not easy for a user to map his/her visual perception of an image into low level features such as color and shape. Another issue is concerned with the subjectivity of the image perception. Different people can have distinct visual perceptions of the same image. Different images may have different meanings or importance for different users. For example, given a picture showing a “car in front of a house”, while a user may be interested in cars, others may be interested in houses.

Most of the RF methods that are being applied to CBIR uses global information, i.e., they consider the image as a monolithic unit. One of the first relevance feedback-based CBIR method was proposed in [13]. In this work, the learning process is based on assigning weights to each descriptor (*interweight*), and also to each feature vector bin, that is, to each position in this vector (*intraweight*). The learning algorithm heuristically estimates the weight values that best encodes the user needs in the retrieval process. In [17], the weight assignment is again employed. However, an optimization framework is applied to estimate the weights. Another pioneer work in this area is the *PicHunter* system, presented in [14]. The PicHunter uses a Bayesian framework in the learning process. This mechanism tries to predict the image closer to the user needs. In [18], another approach for *relevance feedback* using Bayesian inference is proposed: the *rich get richer (RGR)*. This method considers the consistency among successive user feedbacks provided in the learning process. In [19], the query pattern is a set of images, instead of a single one.

In [20] Stejic et al. proposed a "local" approach for image retrieval based on the similarity of region features. They proposed a genetic algorithm (GA)-based relevance feedback method and a new method, Local Similarity Pattern (LSP), for computing image similarity. LSP is defined as a structure containing R and F_R, where R is a set with $N \times N$ regions obtained by the image uniform partitioning, and F_R is a set of image features that are extracted from each region and used for similarity computation. GA and relevance feedback are used to determine the feature that best describes each LSP region. In [21], Stejic et al. proposed the RFSP (Region and Feature Saliency Pattern). The RFSP is defined like a structure such as LSP. But, instead of using GA to determine the feature that best describes each image region, in the RFSP method, there is a weight associated to each region and the GA is used to find the best weights for all region features. A new GA-based relevance feedback technique was proposed in [22]: Local Aggregation Pattern (LAP). In LAP, Stejic et al. used mathematical aggregation operators to combine the similarity regions. So, in this approach, GA-based RF is used to find the best set of mathematical aggregation operators.

There are also RF methods based on SVM (*Suport Vector Machine*) using local information. Jing et al. [23] proposed two relevance feedback algorithms based on region representations. One is inspired from the query point of positive examples together and reweinghting the regions to emphatize the latest ones, a pseudo image is formed as the new query. The other propose a new SVM kernel so as enable the algorithms to be applicable to region-based representations. In another approach, Lin et. al. [24] proposed to carry out the recognition task with adaptative ensemble kernel machines, each of which is derived from proper localization and regularization for object category recognition.

3 Background

3.1 CBIR model

This paper uses the CBIR model proposed in [25,26], described in the following.

Definition 1 *An* **image** $\hat{I}$ *is a pair* (D_I, $\vec{I}$)*, where:* D_I *is a finite set of* pixels *(points in* $\mathbb{Z}^2$*, that is,* $D_I \subset \mathbb{Z}^2$*), and* $\vec{I} : D_I \rightarrow \mathsf{D}'$ *is a function that assigns to each pixel* p *in* D_I *a vector* $\vec{I}(p)$ *of values in some arbitrary space* D' *(for example,* $\mathsf{D}' = \mathbb{R}^3$ *when a color in the RGB system is assigned to a pixel).*

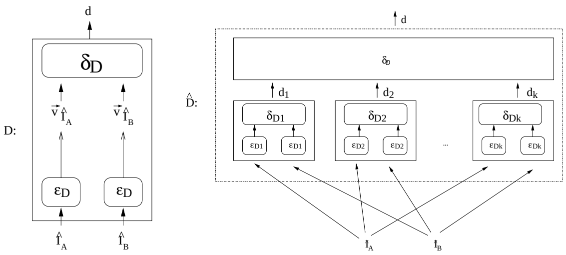
Definition 2 *A* ***simple descriptor*** *(briefly,* **descriptor***)* D *is defined as a pair* (ϵ_D, δ_D)*, where:* $\epsilon_D : \hat{I} \rightarrow \mathbb{R}^n$ *is a function, which extracts a* feature vector $\vec{v}_{\hat{I}}$ *from an* image $\hat{I}$. $\delta_D : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ *is a* similarity function *(e.g., based on a distance metric) that computes the similarity between two images as a function of the distance between their corresponding* feature vectors.

Definition 3 *A* ***feature vector*** $\vec{v}_{\hat{I}}$ *of an image* $\hat{I}$ *is a point in* $\mathbb{R}^n$ *space:* $\vec{v}_{\hat{I}} =$

$(v_1, v_2, ..., v_n)$, where n is the dimension of the vector. They essentially encode image properties, such as color, shape, and texture. Note that different types of feature vectors may require different similarity functions.

Figure 1a illustrates the use of a simple descriptor D to compute the similarity between two images $\hat{I}_A$ and $\hat{I}_B$. First, the extraction algorithm ϵ_D is used to compute the feature vectors $\vec{v}_{\hat{I}_A}$ and $\vec{v}_{\hat{I}_B}$ associated with the images. Next, the similarity function δ_D is used to determine the similarity value d between the images.

Definition 4 *A **composite descriptor** $\hat{D}$ is a pair $(\mathcal{D}, \delta_{\mathcal{D}})$ (see Figure 1b), where: $\mathcal{D} = \{D_1, D_2, \ldots, D_k\}$ is a set of k pre-defined simple descriptors. $\delta_{\mathcal{D}}$ is a similarity combination function which combines the similarity values d_i obtained from each descriptor $D_i \in \mathcal{D}$, $i = 1, 2, \ldots, k$.*



(a) (b)

Figure 1. (a) Simple and (b) Composite descriptors.

3.2 Region-Based Image Similarity Model

This paper uses a RF approach based on local imagem features. In the following, the Region-based Image Similarity Model (RISM) used is described.

In general, RISMs express the image similarity as a combination of region similarities [21]. There are many ways to model image similarity based on regions.

Some approaches are based on a segmentation-process step. However, to partition the image into grids with the same size is easier. Segmentation needs more complex algorithms which, in general, are application dependent.

Thus, in this work the RISM used is based on Stejic et al. methods [20,21,22]. A formalization of RISM is explained in the following.

Let I be a *set of images* that represents the image database. Each image is partitioned into a *set of regions* $R = \{r_1, r_2, ...r_{n_R}\}$. Figure 2 ilustrates the partition of an image into 9 regions. From each region, a set of feature vectors $F_{r_i} = \{f_{1_{r_i}}, f_{2_{r_i}}, ...f_{n_D}\}$ are extracted using a set $\mathcal{D} = \{D_1, D_2, \ldots, D_{n_D}\}$ of n_D descriptors.

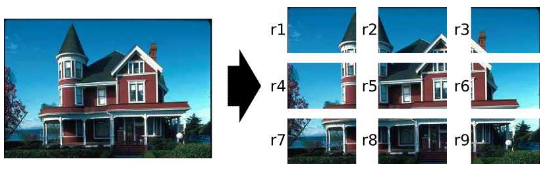


Figure 2. Example of image partition.

A descriptor D_i returns the feature similarity degree $d_i r_{j I_a I_b}$, using a simililarity function δ_{D_i}, from a pair of images I_a and I_b with respect to the image feature f_i of the image region r_j. Given a colection of feature similarity values, a composite descriptor $\mathcal{D}$ returns the image similarity value $d_{i I_a I_b}$ of a pair of images I_a and I_b.

Figure 3 shows the Region-based Image Similarity Model used. For each region of the image, the features vectors and the similarities are calculated by using the k descriptors available. A δ_D function is used to combine the region similarites. It is possible to observe that this structure is a typical composite descriptor like described in Section 3.1.

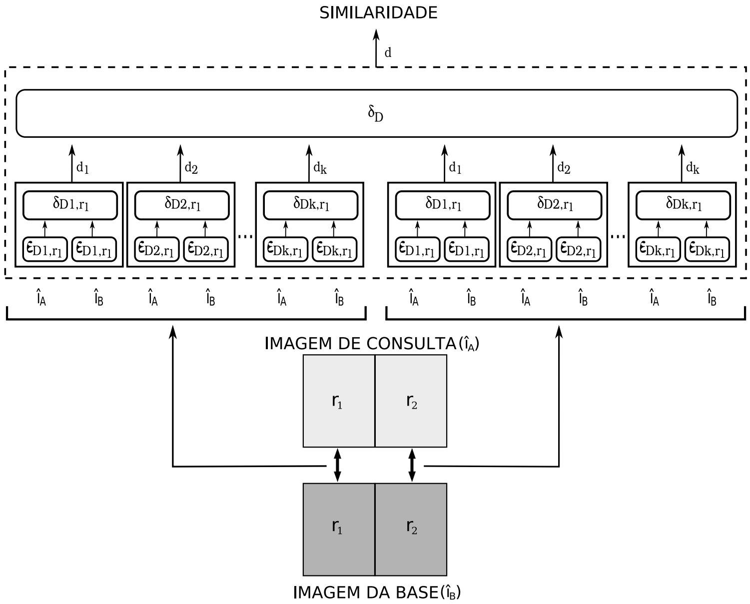


Figure 3. Example of the image similarity computation using the proposed model.

3.3 Genetic Programming

Genetic programming (GP) [5], such as other evolutionary computation algorithms, is an artificial intelligence problem-solving technique based on the principles of biological inheritance and evolution. In GP approach, the individuals represent programs that undergo evolution. The *fitness* evaluation consists in executing these programs, and measuring their degrees of evolution. Genetic programming, then, involves an evolution-directed search in the space of possible computer programs that best solve a given problem.

At beginning of the evolution, an initial population of individuals is created. Next, a loop of successive steps are performed to evolve these individuals: the fitness calculation of each individual, the selection of the individuals, based on their fitness, to breed a new population by applying genetic operators. In the following, these steps are presented in more details.

Usually, a GP individual represents a program and is encoded in a tree. In this enconding, an individual contains two kinds of nodes, *terminals* (leaf nodes) and *functions* (intern nodes). Terminals are usually program inputs, although they may also be constants. Functions take inputs and produce outputs. A function input can be either a terminal or the output of another function.

The fitness of an individual is determined by its effectiveness in producing the correct outputs for all cases in a *training set*. The training set is a set containing inputs and their correspondent previously known outputs.

To evolve the population, and optimize the desired objectives, it is necessary to choose the correct individuals to be subject to genetic operators. Thus, *selection operators* are employed to select the individuals, usually, based on their fitness. Examples of selection method are *roulette wheel*, *tournament* and *rank-based* selections [27].

Genetic operators introduce variability in the individuals and make evolution possible, which may produce better individuals in posterior generations. The *crossover* operator exchanges sub-trees from a pair of individuals, generating two others. *Mutation* operator replaces a randomly chosen sub-tree from an individual by a sub-tree randomly generated. The *reproduction* operator simply copies individuals and insert them in the next generation.

4 Region-Based Image Similarity Model Using GP

This section presents the GP-based CBIR framework used. This framework uses relevance feedback which exploits image similarity based on regions (GP_{LSP}). In this method, a composite descriptor $\hat{D} = (\mathcal{D}, \delta_{\mathcal{D}})$ (see Section 3.1) is employed to rank N database images defined as $DB = \{db_1, db_2, \ldots, db_N\}$. The set of K simple descriptors of $\hat{D}$ is represented by $\mathcal{D} = \{D_1, D_2, \ldots, D_{n_\mathcal{D}}\}$. Database images are partitioned into a set of regions $R = \{r_1, r_2, ...r_{n_R}\}$ (see Section 3.2). The similarity between two image regions $I_{a,rj}$ and $I_{b,rj}$, computed by D_i, is represented by $d_i r_{j I_a I_b}$. All similarities $d_i r_{j I_a I_b}$ are normalized between 0 and 1. A Gaussian normalization [13] is employed to normalize these values. So, the similarity between two images I_a and I_b are obtained combining the $n_\mathcal{D} \times n_R$ image regions similarities.

Let L be a number of regions displayed on each iteration. Let Q be the query pattern $Q = \{q_1, q_2, \ldots, q_M\}$, where M is the number of elements in Q, formed by the query image q_1 and all images defined as relevant during a retrieval session.

4.1 Basic Algorithm

Algorithm 1 presents an overview of the retrieval process used in this paper. The user interactions are indicated in italic. At the beginning of the retrieval process, the user indicates the query image q_1 (line 1). Based on this image, a initial set of images is selected to be shown to the user (line 2). Thus, the user is able to indicate the relevant images, from this initial set, starting the relevance feedback iterations. Each iteration involves the following steps: user indication of relevant regions (line 4); the update of the query pattern (line 5); the learning of the user preference by using GP (line 6); ranking of image regions (line 7); and the exhibition of the most similar regions (line 8).

The selection of the initial set of regions, the use of GP to find the best similarity composition functions and the algorithm to rank regions are the same as presented in [3,4].

Algorithm 1 The GP-based relevance feedback process.

User indication of query image q_1
Show the initial set of regions
while the user is not satisfied **do**
    User indication of the relevant regions
    Update query pattern Q
    Apply GP to find the best individuals (similarity composition functions)
    Rank the image regions
    Show the L most similar regions
end while

4.2 Learning from the indication of non-relevant images

The framework presented in [3,4] uses only the information provided by the images labeled as relevant. One natural extension would be to incorporate non-relevant images/regions to the learning and ranking processes.

Two components of the described framework were adapted to cope with such an extension: the training set definition (Section 4.2.1) and the similarity function employed to rank regions (Section 4.2.2).

4.2.1 Redefinition of the training set

The new training set is composed by relevant, non-labeled, and non-relevant regions. Let IRR be the set of regions labeled as non-relevant over all iterations. Remember that M is the size of the query pattern Q and L is the number of regions displayed on each iteration. The new training set $\mathcal{T}^{\pm}$ is defined as follows.

Definition 5 *The **training set** is defined as a pair* $\mathcal{T}^{\pm} = (T, r^{\pm})$ *where:*

- $T = \{t_1, t_2, \ldots, t_{N_T}\}$ *is a set composed of* N_T *distinct* training regions.
- $r^{\pm} : T \rightarrow \{-1, 0, 1\}$ *is a function that indicates the user feedback about each*

region $t \in T$.

The function $r^{\pm}(t_i)$, where $t_i \in T$, is defined as

$$r^{\pm}(t_i) = \begin{cases} 1, \text{ if } t_i \text{ is relevant.} \\ -1, \text{ if } t_i \text{ is non-relevant.} \\ 0, \text{ if } t_i \text{ is unlabeled.} \end{cases} \tag{1}$$

The fitness computation process is similar to that presented in [3,4]. However, for the $GP^{\pm}$ framework, the highest fitness values are assigned to those individuals that by rank relevant regions at the first positions and the non-relevant ones at the last positions.

4.2.2 Sorting regions

The process of sorting should also consider the regions labeled as non-relevant. This is achieved by defining a new similarity function, $Sim^{\pm}_{\delta_i}(Q, IRR, db_j)$. This function is defined as

$$Sim^{\pm}_{\delta_i}(Q, IRR, db_j) = \frac{max_{\delta_i}(Q, db_j)}{max_{\delta_i}(IRR, db_j)} \tag{2}$$

where the max function is defined as

$$\begin{gathered} max_{\delta_i}(IMG, I) = \{\delta_i(img_k, I) \,|\, \delta_i(img_k, I) > \delta_i(img_l, I) \\ \forall\, img_k, img_l \in IMG \wedge k \neq l\} \end{gathered} \tag{3}$$

Note that the function $Sim^{\pm}_{\delta_i}(Q, IRR, db_j)$ assigns the highest similarity values to regions which are similar to a relevant and not similar to a non-relevant region indicated by the user.

5 The Semi-automatic Vectorization Approach

The proposed vectorization approach can be divided into four main steps: (i) image partition and region feature extraction, (ii) identification of the partitions which are of interest, (iii) image segmentation, and (iv) region vectorization. Figure 4 ilustrates the steps of the vectorization process.

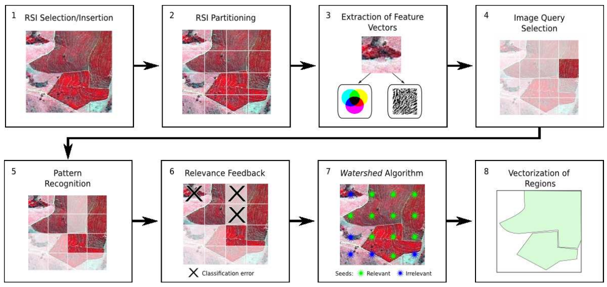


Figure 4. Steps of the proposed vectorization process.

Let I be an RSI and $I_{i \times j}$ a sub image of I composed by $n \times n$ pixels. The *image partition process* consists of creating a grid of $n \times n$ sub images (tiles) from I. The value of n is based on the estimated size of a region of interest. This way, an ideal value of n is that one which makes the sub images be found inside regions of interest. For each subimage of I, spectral and texture features are extracted using pre-defined image descriptors.

Given the definitons above, the process of *identifying relevant partitions* can be naturally performed by small adaptations of the the relevance feedback strategy described before. Each tile is considered as an independent image and this process starts by the indication of a query image by the user. This query image is assumed to present the same texture and spectral properties of the RSI regions which are of interest. A similarity search is performed and the most similar tiles are returned to the user. The user then indicates if the returned tiles are relevant or non-relevant. By using this feedback, the classification system learns the user needs and tunes itself in order to improve the results in the next iteration. This process is repeated until the user is satisfied with the result. The identification of relevant partitions uses the relevance feedback approach described in Section 4.

After the tiles of interest are identified, the next step is concerned with the *segmentation* of relevant regions. The segmentation process of the image is performed by using a “watershed”-based [28] algorithm. This algorithm segmentates images using seeds. The seeds are based on areas of interest identified in the last step.

Finally, the vectorization process consists of using the segmented image to extract the polygons of the region of interest.

This work describes the results obtained for the first two main steps, namely: *partition/extraction of image features* and *recognition of regions of interest*.

6 Experiments

This section describes in details the experiments performed to validate our framework. Two different sets of experiments were conducted. The first one aims at evaluating the impact of using non-relevant partitions into the RF approach. In these experiments, we also constrast our proposed approach against a recent and effective region-based image retrieval approach. The second set of experiments evaluates the proposed method with regards to its effectiveness on classifying remote sensing images.

Table 1

Descriptors used in the experiments.

Descriptor	**Sim. Function**	**Type**
Color Histogram [29]	$L1$	Color
Color Moments [30]	d_{mom} [30]	Color
BIC [31]	$dLog$ [31]	Color
Gabor Wavelets [32]	Euclidean	Texture
Spline Wavelets [33]	Euclidean	Texture

6.1 Impact of non-relevant images

6.1.1 Setup

- Image descriptors

  The proposed method was presented in a generic way, since there are no restrictions with regards to the descriptors that can be used to characterize the images. Color and texture based descriptors are the most common ones and were in fact used in the experiments described next. Table 1 presents the used descriptors.

- Baselines

  We compare our method against the LAP approach proposed by Stejic et al. [22]. As mentioned in section 2, LAP [22] is a current and effective method to compute image similarity based on the similarity of the regions. LAP is also based on an evolutionary approach making the comparison fairer. Experiments with other baselines are left for future work.

  LAP computes the similarity of two regions based on local information. This process is comprised of two steps: first, the similarity of regions are computed; second, the region similarity values are combined by means of *mathematical aggregation operators*. Stejic et al. [22] defined a mathematical aggregation operator a as a function of the form $a : [0,1]^n \mapsto [0,1]$. They used genetic algorithms to find a good set of operators to combine the similality values. The complete set is composed by 67 aggregation operators.

- Image Database

  The Image Database used in the experiments was a subset of the heterogeneous colection of 20000 images from the Corel GALLERY Magic — Stock Photo Library 2. The used subset is composed of 3906 images, distributed among 85 classes. These distribution of images per class is also skewed, with classes sizes varying from 7 to 98 images.

- GP_{LSP} Implementation

  We implement a CBIR system with the minimal requirements to validate our method. The configuration parameters used in the implementation are shown in Table 2. These parameters were determined empirically through several experiments. As can be seen in the table, only crossover and mutation operators were used in the search process. Both uses 2-tournament as selection method. Due to the small population size, the use of reproduction operator makes the population diversity fall down quickly. Thus, this operator was not employed. The protected division used in the function set returns 1 if divisor value was zero. The maximum number of generations adopted was 10, but if a individual has normalized fitness value (between 0 and 1) equal to 1 before the last generation, the GP run is finished earlier. The used fitness function (FFP2) is presented in [34].

- Effectiveness measures

  We use *precision-recall* curves to evaluate performance in the experiments. Precision-Recall curve is a common performance evaluation criterion used in

Table 2

Configuration parameters.

Population size	30
Maximum number of generations	10
Maximum tree depth	6
Function set	$+, \times, /$ (*protected*)
Terminal set	similarity by simple descriptors
Initialization	half and half
Initial ramp	$2-6$
Crossover rate	0.80
Mutation rate	0.20
Selection method	tournament (size 2)
Fitness function	FFP2 (see [3,4])
Training set size	80 (3.6$\%$ of DB)
Voting selection ratio threshold	1.00

information retrieval systems that have been employed to evaluate CBIR systems. Precision $Pr(q)$ can be defined as the number of retrieved relevant images $R(q)$ over the total number of retrieved images $N(q)$ for a given query q, that is $Pr(q) = \frac{R(q)}{N(q)}$. Recall $Re(q)$ is the number of retrieved relevant images $R(q)$ over the total number of relevant images $M(q)$ present in the database for a given query q, that is $Re(q) = \frac{R(q)}{M(q)}$.

6.1.2 Experimental design

The user behavior was simulated by computer. At each iteration, all images belonging to the same class of the query are labeled as relevant. Experiments considered 10 iterations for each query. At each iteration, 20 images were displayed. The first set of images displayed, for a given query, are based on the average of the similarity values measured by each employed descriptor. We refer to our approach as GP_{LSP}. GP_{LSP}^{+} considers only relevant regions while $GP_{LSP}^{\pm}$, in turn, considers also non-relevant regions.

6.1.3 Results

Figure 5 shows the precision-recall curves of the GP_{LSP} method using different partitionings of the image area with resolution 3×3, 4×4, 5×5, 6×6, and 7×7 regions. The $GP_{LSP(3\times3)}$ presents best results for recall values greater than 0.3. Similar results are seen in Figure 6 which shows the results for the method with positive and negative feedback.

Figure 7 compares the best GP-Based RF methods (GP_{LSP} and $GP_{LSP}^{\pm}$ (3×3)) with the LAP method. As can be observed, the proposed methods have a significant better effectiveness for all recall values. In fact for several recall levels the difference in performance is more than 100%. We can also see that in the first levels of recall (e.g., when the first percentages of relevant images are retrieved) the precision is very high, meaning that there is, in average, very few irrelevant images in the top of the rankings. Finally, when compared to each other, we can see that there is a slight advantage of $GP_{LSP}^{\pm}$ over GP_{LSP} in almost all recall levels.

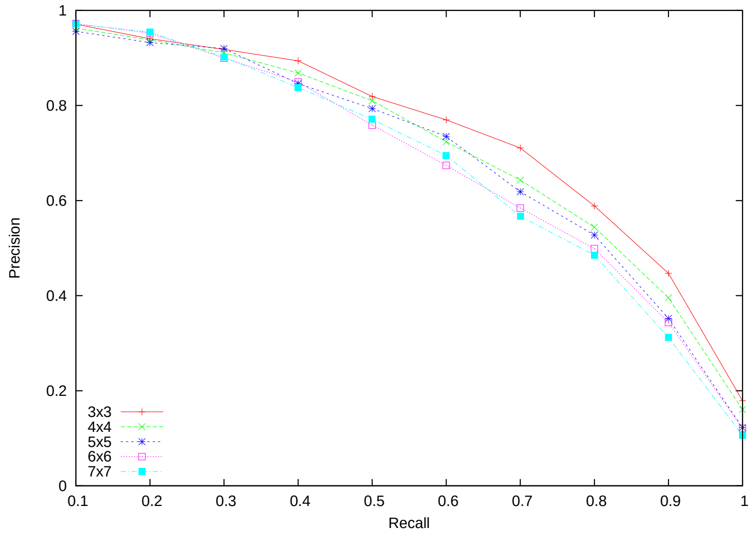


Figure 5. Precision-recall curves showing GP^{+}_{LSP} method in 3×3, 4×4, 5×5, 6×6 and 7×7 grids partitions effectiveness in the Corel database.

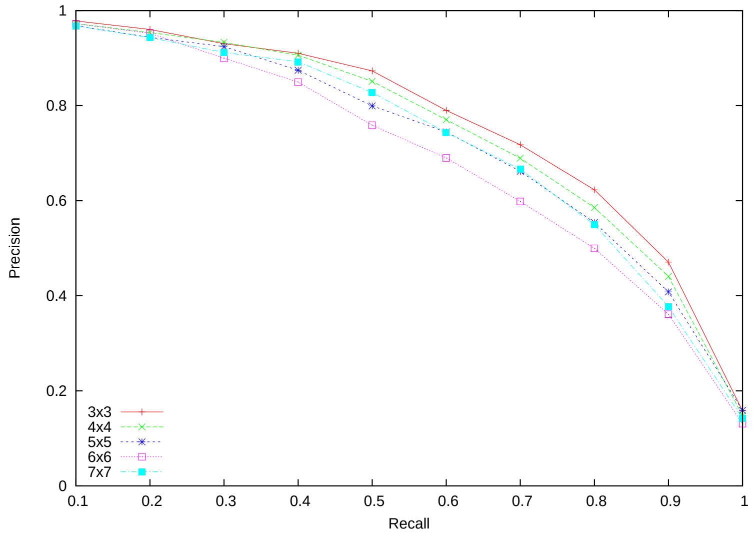


Figure 6. Precision-recall curves showing $GP^{\pm}_{LSP}$ method in 3×3, 4×4, 5×5, 6×6 and 7×7 grids partitions effectiveness in the Corel database.

6.2 Validation of the proposed GP-based RF strategy

This section describes the experiments performed to validate our method to identify and classify regions.

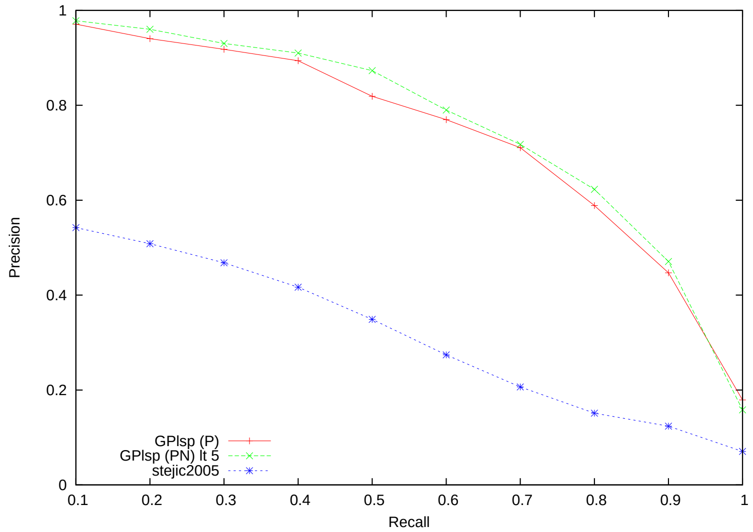


Figure 7. Precision-recall curves showing GP_{LSP} (including positive approach and positive-negative approach) and LAP methods effectiveness in the Corel database.

6.2.1 Setup

- Remote Sensing Images

  Two RSIs are used to validade our method. One can be considered as of “easy recognition” (pasture image) while the other is “hard recognition” (coffee), due to its higher level of noise. Information about used RSIs is showed in Table 3. In particular, the sets of regions extracted from each RSI are seen as a whole data set of test images. In this sense, theses bases are even larger, i.e., they have more images, than the Core Database used before.

- Image Descriptors

  The descriptors described in Table 1 were used in this experiment.

- Baselines

  We compare our method against *Maximum Likelihood (MaxVer) Classification* [1]. It is the most common supervised classification method used with remote sensing image data. LAP could also have been used, but this would require adaptations of the method for the case of RSI. Moreover, given also the poor

Table 3

Remote Sensing Images used in the experiments.

	Image1	**Image2**
Recognition level	easy	hard
Region of interest	pasture	coffee
Terrain	plain	mountainous
Satelite	CBERS	SPOT
Spatial resolution	20 meters	2,5 meters
Bands composition	R-IR-G (342)	IR-NIR-R (342)
Acquisition date	08–20–2005	08–29–2005
Location	“Laranja Azeda” Basin, MS	Monte Santo County, MG
Dimensions (px)	1310×1842	2400×2400

performance of LAP when compared to our method in the previous Section for the exact scenario for which it was developed, in here we prefer to use a standard baseline for RSI which would allow an indirect comparison of other methods against our approach.

MaxVer is considered as a parametric algorithm and it assumes a particular class statistical distribution, commonly the normal distribution. The implementation of MaxVer algorithm requires the computation of the probability that each pixel belongs to each of the defined classes. Pixels are then assigned to the class with highest probabity.

- Implementation

As before, the system is implemented with the minimal requirements to validate our method. The recognition of partitions of interest requires the definition of several GP parameters (e.g., mutation rate, population size). We used the same parameters as showed in Table 2.

- Effectiveness Measure

The classification results of the proposed method are associated with the number of user interactions. Since in here we are interested in the relative performance of the methods to classify or identify RSI regions in the map at each iteration, (in contrast to the overall performance of the methods as in the previous experiment), and with different configurations, to analyze the results we use *kappa–interactions* curves. Kappa is an effective index to compare classified images, commonly used in the RSI retrieval area with feedback. To calculate it is necessary to create an *error matrix*. An *error matrix* is a square array of numbers set out in rows and columns. It expresses the number of sample units (pixels, clusters, or polygons) assigned to a particular category in one classification relative to the number of sample units assigned to a particular category in another classification [35]. The matrix error is a very effective way to represent map accuracy in that the individual accuracies of each category are plainly described along with both the errors of inclusion (commissions errors) and errors of exclusion (omission errors) present in the classification. A commission error is simply defined as including an area into a category when it does not belong to that category. An omission error is excluding that area from the category in which it truly does belong. The kappa index does not use just the elements from the main diagonal of the error matrix, but includes all of them.

6.2.2 Experimental Design

In our experiments we fixed the tile size according to the common extension value of a *region of interest*. Coffee crops are normally in small parcels on the same farm. We defined, based on previous studies, that 75×75 $meters$ is a good value to the size of the partition. To pasture parcels, that are larger, the chosen value was 400×400 $meters$. The dimension of partitions are fixed in experiments. We used 30×30 $pixels$ to partition the coffee image and 20×20 $pixels$ for the pasture image. The number of partitions for the pasture and coffee images were 5980 and 6400, respectvively.

We used a "mask" contained all regions of interest from the RSIs used in the experiments. A "mask" is a binary image where value 1 represents pixels of regions of interest. The "masks" used in our experiments were manually classified by agricultural specialists.

As before, the user interaction was simulated. To do it, we created a *groundtruth* based on the "mask". The *groundtruth* is also a binary image. A relevant partition within it corresponds to a set of pixels with value 1, in which the number of relevant pixels is higher than a given percentual value. In our experiments this value was fixed in 50%. This value was empirically shown to produce a reasonable number of relevant partitions for the feedback experiments. With this percentual value, in average, between 6% and 10% of all partitions in the groundtruth are relevant in the experiments. The number of partitions showed to the user in each iteration was 20.

The proposed technique to recognize regions (section [36,37]) creates a ranking of partitions based on their similarity with regard the reference image defined by the user. On the other hand, the results have to be a binary image representing relevants and irrelevants partitions. Thus, we made experiments using different *thresholds* to separate relevant and irrelevant partitions: 5%, $7,5\%$, 10%, 15%, 20%, and 30% of the top ranked partitions. For the experiment to recognize coffee, we also tested the threshold value 40% because the number of relevant images in this case is higher.

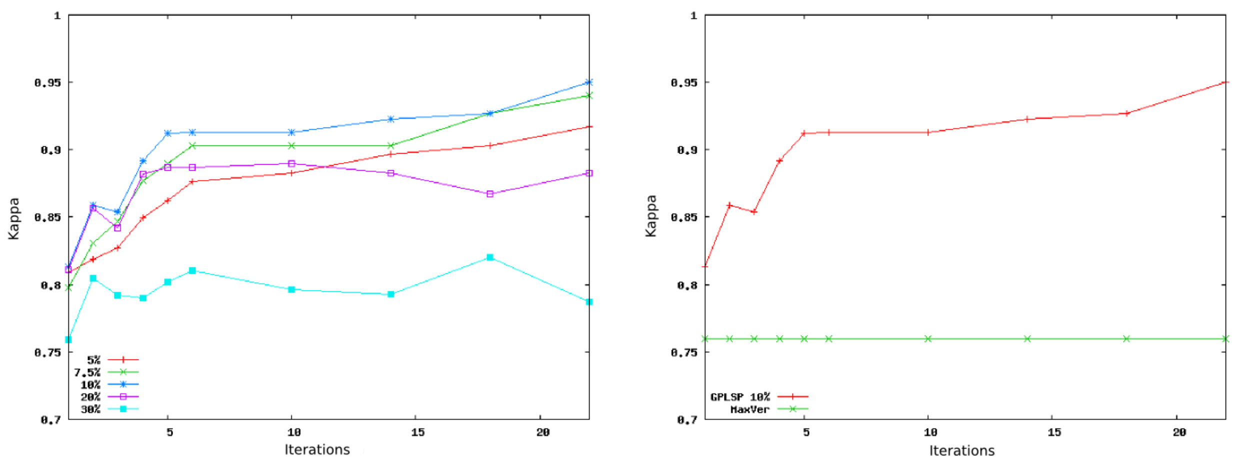
Finally, as mentioned before, the proposed method is compared with *MaxVer*. Image 1 was classified by *MaxVer* with probability threshold 0.8 and using 20.580 points of the pasture sample. Image 2 was classified with probability threshold 0.98 and using 43.630 points of the coffee sample.

6.2.3 Results

Figure 8(a) shows the results referring to the pasture image for the GP_{LSP} method. It shows curves related to the kappa index variation along iterations, considering different thresholds applied to the ranked partitions: 5%, 7.5%, 10% , 20%, and 30%. Figure 8(b) shows the best curve in the Figure 8(a) and the kappa index obtained using the *MaxVer* image classification. We can see that the kappa values of the GP method start already with higher values than MaxVer and thta by the end of the last iteration the relative gains are around 25%.

Figure 9 ilustrates the original RSI, the mask (ground truth) and the classifications considering different threshold values to pasture recognition.

In Figure we show the corresponding results for the $GP^{\pm}_{LSP}$ method. In this case, the best results were obtained with the 10/besides being slightly better in the end of the whole process when compared to GP_{LSP}, this method converges much faster achieving performance close to the maximum after 10 iterations while GP_{LSP} needs more than 20 iterations to achieve similar results. In a real setting with real



(a) (b)

Figure 8. Kappa-Iterations curves considering the pasture image. (a) The proposed classification method considering 5%, 7.5%, 10%, 20%, and 30% threshold values. (b) The best curve showed in (a) – considering a threshold value of 7.5% – and the $MaxVer$ classification accuracy.

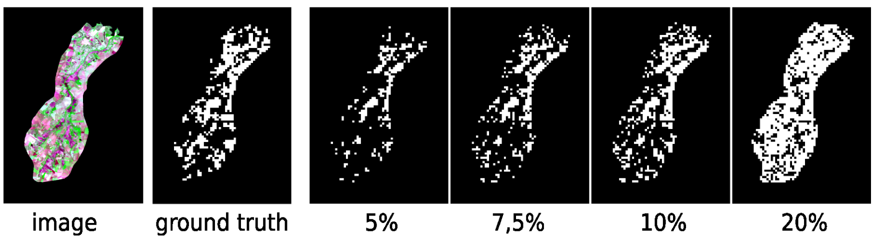
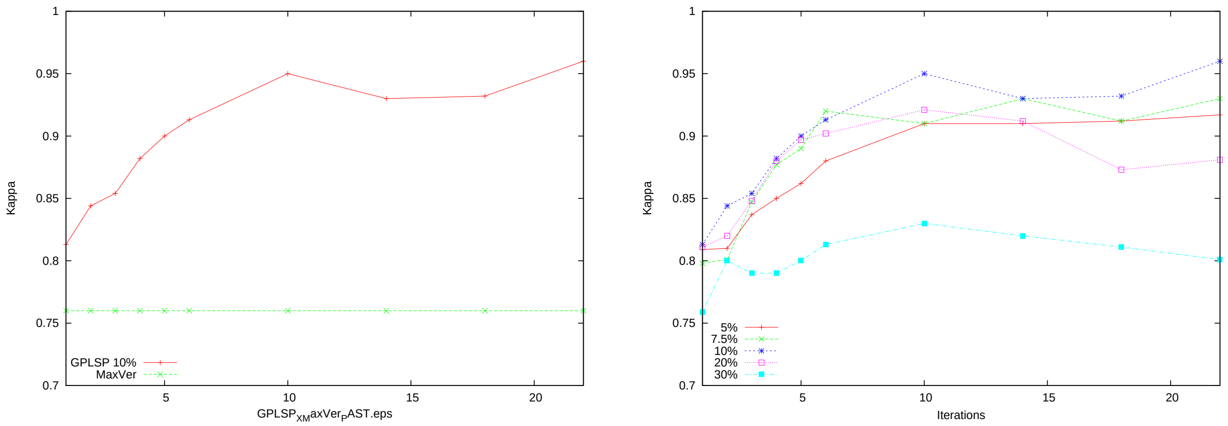


Figure 9. Steps of the proposed vectorization process.

users, it is important to achieve the best performance with as few iterations as possible..

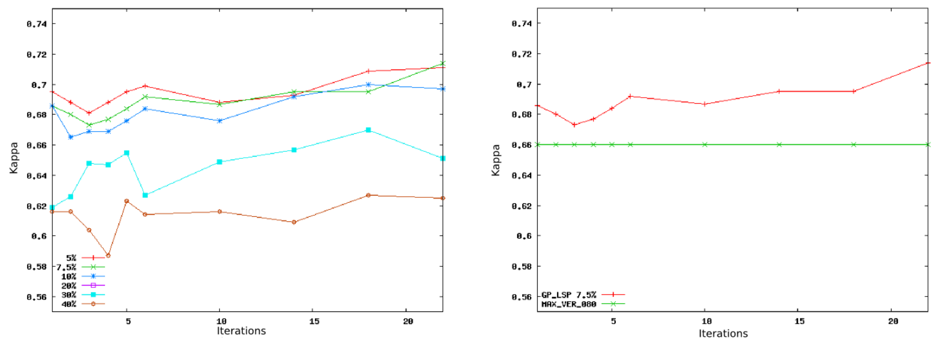
Figure 11 (a) shows the kappa-iterations curves for the coffee image for GP_{LSP}. Figure 11(b) shows the best line curve in the Figure 11 (a) and the kappa index obtained using the *MaxVer* image classification. Again the GP method starts slightly higher than the baseline. However, differently than before, the results remain almost steady for a number of iterations, but improve a bit more by the end of the process. The overall relative gain is around 7.5% in the end of the process. Regard-



(a) (b)

Figure 10. Kappa-Iterations curves considering the pasture image to $GP^{\pm}_{LSP}$. (a) The proposed classification method considering 5%, 7.5%, 10%, 20%, and 30% threshold values. (b) The best curve showed in (a) – considering a threshold value of 7.5% – and the $MaxVer$ classification accuracy.

ing $GP^{\pm}_{LSP}$, the results are qualitatively similar to the pasture experiments, thus leading to similar conclusions.



(a) (b)

Figure 11. Kappa-Iterations curves considering the coffee image. (a) The proposed classification method considering 5%, 7.5%, 10%, 20%, and 30% threshold values. (b) The best curve showed in (a) – considering a threshold value of 7.5% – and the $MaxVer$ classification accuracy.

Figure 12 ilustrates the original RSI, the mask (ground truth) and the classifications using considering different threshold values to coffee recognition.

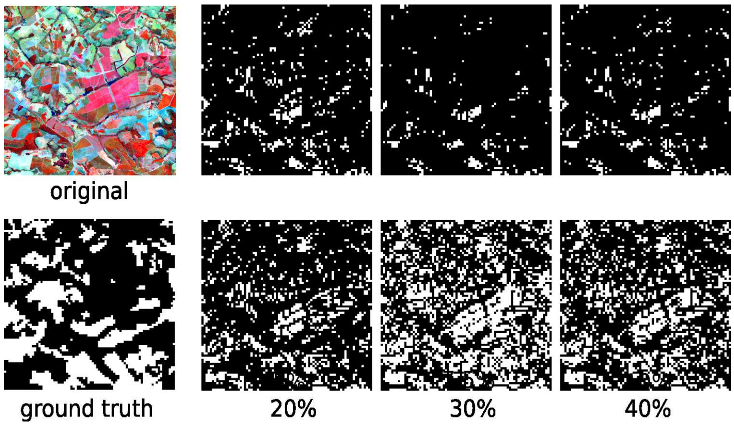


Figure 12. Steps of the proposed vectorization process.

7 Conclusions

We have presented a relevance feedback approach to classifiy remote sensing images. This method explores a genetic programming approach and local properties to learn the user preferences and combine region similarity features. The method explores not only positive feedback but can also take advantage of negative (non-relevant) examples.

Experiments showed that the proposed method is effective to recognize regions of interest, presenting better accuracy than a strong baseline for region-based image retrieval, in general, and than the traditional MaxVer method, for RSI.

The next stage of our work is to implement a watershed-based segmentation algorithm and to compare our method with other region-based classification approaches. We also plan to run new experiments using different image collections.

8 Acknowledgments

Authors are grateful to FAPESP, CAPES, CNPq, and Microsoft for financial support.

References

[1] R. Showengerdt, Techniques for Image Processing and Classification in Remote Sensing, Academic Press, New York, 1983.

[2] X. S. Zhou, T. S. Huang, Relevance feedback in image retrieval: A comprehensive review, Multimedia System 8 (6) (2003) 536–544.

[3] C. D. Ferreira, R. da S. Torres, M. A. Goncalves, W. Fan, Image Retrieval with Relevance Feedback based on Genetic Programming, in: Brazilian Symposium on Data Bases, Campinas, SP, 2008, pp. 120–134.

[4] J. A. Santos, C. D. Ferreira, R. da S. Torres, A Genetic Programming Approach for Relevance Feedback in Region-based Image Retrieval Systems, in: XXI Brazilian Symposium on Computer Graphics and Image Processing, Campo Grande, MS, 2008, pp. 155–162.

[5] J. R. Koza, Genetic Programming: On the Programming of Computers by Means of Natural Selection, MIT Press, Cambridge, MA, USA, 1992.

[6] B. Bhanu, Y. Lin, Object Detection in Multi-Modal Images Using Genetic Programming, Applied Soft Computing 4 (2) (2004) 175–201.

[7] W. Fan, M. D. Gordon, P. Pathak, A generic ranking function discovery framework by genetic programming for information retrieval, Information Processing & Management 40 (4) (2004) 587–602.

[8] B. Zhang, M. A. Gonçalves, W. Fan, Y. Chen, E. A. Fox, P. Calado, M. Cristo, Combining structural and citation-based evidence for text classification, in: Proceedings of the 13th ACM Conference on Information and Knowledge Management, 2004, pp. 162–163.

[9] R. da S. Torres, A. X. Falcão, M. A. Goncalves, J. P. Papa, B. Zhang, W. Fan, E. A. Fox, A genetic programming framework for content-based image retrieval, Pattern Recognition 42 (2) (2009) 283–292.

[10] D.-K. Mo, H. Lin, J. Li, H. Sun, Y.-J. Xiong, Design and implementation of a high spatial resolution remote sensing image intelligent interpretation system, Data Science Journal 6 (2007) S445–S452.

[11] I. Yildirim, O. K. Ersoy, B. Yazgan, Improvement of classification accuracy in remote sensing using morphological filter, Advances in Space Research.

[12] S. Kim, E. A. Fox, W. Fan, C. North, D. Tatar, R. da S. Torres, Design and Evaluation of Techniques to Utilize Implicit Rating Data in Complex Information Systems, Tech. Rep. TR-07-20, Computer Science Department, Virginia Tech (2007).

[13] Y. Rui, T. S. Huang, M. Ortega, S. Mehrotra, Relevance Feedback: A Power Tool for Interactive Content-Based Image Retrieval, IEEE Transactions on Circuits and Systems for Video Technology 8 (5) (1998) 644–655.

[14] I. J. Cox, M. L. Miller, T. P. Minka, T. V. Papathomas, P. N. Yianilos, The Bayesian Image Retrieval System, PicHunter: Theory, Implementation, and Psychophysical Experiments, IEEE Transactions on Image Processing 9 (1) (2000) 20–37.

[15] P. Hong, Q. Tian, T. S. Huang, Incorporate support vector machines to content-based image retrieval with relevant feedback, in: Proceedings of the 7th IEEE International Conference on Image Processing, 2000, pp. 750–753.

[16] S. Tong, E. Y. Chang, Support vector machine active learning for image retrieval, in: Proceedings of 9th ACM international conference on Multimedia, ACM Press, New

York, NY, USA, 2001, pp. 107–118.

[17] Y. Rui, T. Huang, Optimizing learning in image retrieval, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2000, pp. 236–245.

[18] L. Duan, W. Gao, W. Zeng, D. Zhao, Adaptive relevance feedback based on Bayesian inference for image retrieval, Signal Processing 85 (2) (2005) 395–399.

[19] M. Cord, J. Fournier, S. Philipp-Foliguet, Exploration and search-by-similarity in cbir, in: XVI Brazilian Symposium on Computer Graphics and Image Processing, 2003, pp. 175–182.

[20] Z. Stejic, Y. Takama, K. Hirota, Genetic algorithms for a family of image similarity models incorporated in the relevance feedback mechanism, Appl. Soft Comput. 2 (4) (2003) 306–327.

[21] Z. Stejic, Y. Takama, K. Hirota, Relevance feedback-based image retrieval interface incorporating region and feature saliency patterns as visualizable image similarity criteria, Industrial Electronics, IEEE Transactions on 50 (5) (Oct. 2003) 839–852.

[22] Z. Stejic, Y. Takama, K. Hirota, Mathematical aggregation operators in image retrieval: effect on retrieval performance and role in relevance feedback, Signal Processing 85 (2) (2005) 297–324.

[23] F. Jing, M. Li, H.-J. Zhang, B. Zhang, Relevance feedback in region-based image retrieval, Circuits and Systems for Video Technology, IEEE Transactions on 14 (5) (May 2004) 672–681.

[24] Y.-Y. Lin, T.-L. Liu, C.-S. Fuh, Local ensemble kernel learning for object category recognition, Computer Vision and Pattern Recognition, 2007. CVPR ’07. IEEE Conference on (17-22 June 2007) 1–8.

[25] R. da S. Torres, A. X. Falcão, M. A. Goncalves, B. Zhang, W. Fan, E. A. Fox, P. Calado, A New Framework to Combine Descriptors for Content-based Image

Retrieval, in: Proc. of the 14th ACM Conference on Information and Knowledge Management, 2005, pp. 335–336.

[26] R. da S. Torres, A. X. Falcão, M. A. Goncalves, B. Zhang, W. Fan, E. A. Fox, A New Framework to Combine Descriptors for Content-based Image Retrieval, Tech. Rep. IC-05-21, Institute of Computing, University of Campinas, Campinas, Brazil (2005).

[27] T. Bäck, D. B. Fogel, Z. Michalewicz, Evolutionary Computation 1 Basics Algorithms and Operators, Institute of Physics Publishing, 2002.

[28] R. Lotufo, A. Falcão, The ordered queue and the optimality of the watershed approaches, in: In Mathematical Morphology and its Applications to Image and Signal Processing, Kluwer Academic Publishers, 2000, pp. 341–350.

[29] M. Swain, D. Ballard, Color Indexing, International Journal of Computer Vision 7 (1) (1991) 11–32.

[30] M. A. Stricker, M. Orengo, Similarity of Color Images, in: Storage and Retrieval for Image and Video Databases (SPIE), 1995, pp. 381–392.

[31] R. Stehling, M. Nascimento, A. Falcão, A Compact and Efficient Image Retrieval Approach Based on Border/Interior Pixel Classification, in: Proceedings of the 11th ACM International Conference on Information and Knowledge Management, ACM Press, McLean, Virginia, USA, 2002, pp. 102–109.

[32] T. S. Lee, Image representation using 2d gabor wavelets, IEEE Transactions Pattern Analysis Machine Intelligence 18 (10) (1996) 959–971.

[33] M. Unser, A. Aldroubi, M. Eden, A family of polynomial spline wavelet transforms, Signal Process. 30 (2) (1993) 141–162.

[34] W. Fan, E. A. Fox, P. Pathak, H. Wu, The Effects of Fitness Functions on Genetic Programming-Based Ranking Discovery for Web Search, Journal of the American Society for Information Science and Technology 55 (7) (2004) 628–636.

[35] R. G. Congalton, K. Green, Assessing the Accuracy of Remotely Sensed Data: Principles and Practices., Lewis Publishersr, Washington, DC, 1977.

[36] C. Ferreira, R. Torres, M. Gonalves, W. Fan, Image retrieval with relevance feedback based on genetic programming, in: 23nd Brazilian Symposium on Database, Campinas, SP, Brasil, 2008.

[37] J. A. dos Santos, C. Ferreira, R. Torres, A genetic programming approach for relevance feedback in region-based image retrieval systems, in: Brazilian Symposium on Computer Graphics and Image Processing, 21 (SIBGRAPI), Campo Grande, MS, Brasil, 2008.

Jefersson Alex dos Santos concluded his undergraduate course in Computer Science in 2006 at State University of Mato Grosso do Sul. He is a master student at University of Campinas. His research interests include image retrieval, machine learning, and image classification.

Cristiano Dalmaschio Ferreira concluded his undergraduate course in Computer Science in 2005 at Federal University of Viosa. He received a MSc in Computer Science from the University of Campinas in 2007. His research interests include image retrieval, machine learning and data analysis.

Ricardo da Silva Torres received a BSc in Computer Engineering from the University of Campinas, Brazil, in 2000. He got his doctorate in Computer Science at the same university in 2004. He has been Professor at the Institute of Computing, University of Campinas, since 2005, and his research interests include image analysis, content-based image retrieval, image databases, and geographic information systems.

Marcos André Gonçalves concluded his doctoral degree in Computer Science at Virginia Tech in 2004. He earned a Master degree from University of Campinas (UNICAMP) in 1997 and a Bachelor degree from the Federal University of Cear (UFC) in 1995, both in Computer Science. He has published more tahn 20 journal papers and 80 conference/workshop papers in the digital library, databases, and information retrieval fields.

Rubens Lamparelli