Improving Graph-Based Image Classification by using Emerging Patterns as Attributes

Niusvel Acosta-Mendoza[a,*], Andrés Gago-Alonso[a], Jesús Ariel Carrasco-Ochoa[b], José Francisco Martínez-Trinidad[b], José Eladio Medina-Pagola[a]

[a] *Advanced Technologies Application Center (CENATAV), 7a ♯ 21406 e/ 214 and 216, Siboney, Playa, CP: 12200, Havana, Cuba*
[b] *Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE), Luis Enrique Erro No. 1, Sta. María Tonantzintla, Puebla, CP: 72840, Mexico*

Abstract

In recent years, frequent approximate subgraph (FAS) mining has been used for image classification. However, using FASs leads to a high dimensional representation. In order to solve this problem, in this paper, we propose using emerging patterns for reducing the dimensionality of the image representation in this approach. Using our proposal, a dimensionality reduction over 50% of the original patterns is achieved, additionally, better classification results are obtained.

Keywords: Approximate graph mining, approximate graph matching, emerging patterns, graph-based classification.

1. Introduction

Several authors have turned their attention to graph-based mining techniques where frequent subgraphs are detected allowing some distortions in the data (Holder et al., 1992; Anchuri et al., 2013; Flores-Garrido et al., 2014b; Acosta-Mendoza et al., 2015). Since in practical applications, it is not common to have two instances exactly equal, several algorithms for frequent approximate subgraph (FAS) mining have been developed (Song and

*Corresponding author
Email addresses: nacosta@cenatav.co.cu (Niusvel Acosta-Mendoza), agago@cenatav.co.cu (Andrés Gago-Alonso), ariel@ccc.inaoep.mx (Jesús Ariel Carrasco-Ochoa), fmartine@ccc.inaoep.mx (José Francisco Martínez-Trinidad), jmedina@cenatav.co.cu (José Eladio Medina-Pagola)

Chen, 2006; Zhang and Yang, 2008; Zou et al., 2009; Jia et al., 2011; Acosta-Mendoza et al., 2012a; Anchuri et al., 2013; Flores-Garrido et al., 2014a,b). The usefulness of the patterns computed by these algorithms has been shown in different classification tasks (Holder et al., 1992; Jia et al., 2011; Acosta-Mendoza et al., 2012c; Gago-Alonso et al., 2013; Morales-González et al., 2014); but only a few of them have been applied to image classification (Acosta-Mendoza et al., 2012a,c, 2013; Morales-González et al., 2014). In these works, the authors have reported good results; however, they use a large number of graphs (patterns) as attributes for describing the images. This fact affects the performance of the classifiers due to the high dimensionality of the representation. Moreover, some of these patterns do not provide useful information for classification. To solve this problem, several strategies to reduce the set of patterns in several contexts have been proposed (Jin and Wang, 2011; Acosta-Mendoza et al., 2013; Kong et al., 2013; Aridhi et al., 2015; Moradi and Rostami, 2015). In this paper, we propose to use emerging approximate graph patterns for image classification. An emerging pattern is a discriminative pattern whose support increases significantly into a class regarding to the remaining classes. Therefore, we will focus on holding only emerging patterns from the set of frequent patterns obtained by a FAS mining algorithm and using them as attributes for image classification. Our proposal reduces the number of patterns to be considered as attributes, and as we will show in our experiments, it allows increasing the efficiency and effectiveness of image classification, compared against the best methods reported in the literature (Acosta-Mendoza et al., 2012a,c, 2013). Furthermore, to the best of our knowledge, this is the first work that uses FAS mining and emerging pattern selection for image classification.

The organization of this paper is the following. In Section 2, some basic concepts are presented. In Section 3, the related work on image classification methods based on FAS, as well as, the FAS mining algorithms are presented. In Section 4, our proposal for reducing the representation of images in the collection through the use of emerging patterns is introduced. Later, in Section 5, through some experiments, the efficiency and effectiveness of our proposal are shown. Finally, our conclusions and future work directions are discussed in Section 6.

2. Basic concepts

This work is focused on collections of simple undirected labeled graphs. Henceforth, when we refer to a graph, we are assuming this type of graphs.

A *labeled graph* in the domain of all possible labels $L = L_V \cup L_E$, where L_V and L_E are the label sets for vertices and edges respectively, is a 4-tuple, $G = (V, E, I, J)$, where V is a set whose elements are called *vertices*, $E \subseteq \{\{u, v\} \mid u, v \in V, u \neq v\}$ is a set whose elements are called *edges* (the edge $\{u, v\}$ connects the vertex u with the vertex v), $I : V \to L_V$ is a *labeling function* for assigning labels to vertices and $J : E \to L_E$ is a *labeling function* for assigning labels to edges.

Let $G_1 = (V_1, E_1, I_1, J_1)$ and $G_2 = (V_2, E_2, I_2, J_2)$ be two graphs, G_1 is a *subgraph* of G_2 if $V_1 \subseteq V_2$, $E_1 \subseteq E_2$, $\forall u \in V_1, I_1(u) = I_2(u)$, and $\forall e \in E_1, J_1(e) = J_2(e)$. In this case, we use the notation $G_1 \subseteq G_2$ and we also say that G_2 is a *supergraph* of G_1.

Given two graphs G_1 and G_2, a function f is an *isomorphism* between these graphs if $f : V_1 \to V_2$ is a bijective function, where:

- $\forall u \in V_1 : f(u) \in V_2 \wedge I_1(u) = I_2(f(u))$
- $\forall \{u, v\} \in E_1 : \{f(u), f(v)\} \in E_2 \wedge J_1(\{u, v\}) = J_2(\{f(u), f(v)\})$

If there is an isomorphism between G_1 and G_2, we say that G_1 and G_2 are *isomorphic*. If G_1 is isomorphic to G_3 and $G_3 \subseteq G_2$, then we say that there is a *sub-isomorphism* between G_1 and G_2, and we also say that G_1 is sub-isomorphic to G_2.

Let $D = \{G_1, \ldots, G_{|D|}\}$ be a collection of graphs and G be a labelled graph in L, the *support* value of G in D is defined as the fraction of graphs $G_i \in D$, such that there is a sub-isomorphism between G and G_i. This value of support is obtained through equation (1):

$$supp(G, D) = \frac{|\{G_i \in D : G \text{ is sub-isomorphic to } G_i\}|}{|D|} \tag{1}$$

Let Ω be the set of all possible labelled graphs in L, the *similarity* between two graphs $G_1, G_2 \in \Omega$ is defined as a function $sim : \Omega \times \Omega \to [0, 1]$. We say that the graphs are very

different if $sim(G_1, G_2) = 0$, the higher the value of $sim(G_1, G_2)$ the more similar the graphs are, and if $sim(G_1, G_2) = 1$ then there is an isomorphism between these graphs.

In Figure 1, an example of the similarity evaluation between two graphs is shown. For our example, it is supposed that the similarity function is based on the product of the similarities between vertices and between edges of two graphs with the same topology. Also, it is assumed that the vertices with labels "A" and "C" could substitute the vertices with labels "D" and "A" with similarity 0.667 and 0.2, respectively (represented by dashed lines in Figure 1). Moreover, it is assumed that edges with label "Y" could substitute edges with label "Z" with similarity 0.667 (represented by dotted lines in Figure 1). Finally, the vertices with label "B" and edges with labels "W" cannot substitute any other vertex or edge except themselves with a similarity of 1.0 (represented by continuous lines in Figure 1). Then, the graph G_1 is similar to G_2 with a similarity of 0.09.

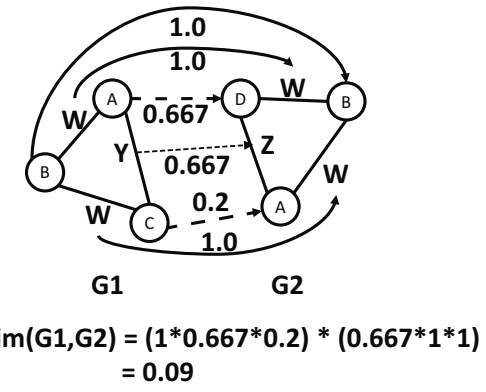

Figure 1: Example of similarity between two graphs.

Between two labelled graphs, there could be more than one inexact correspondence among its vertices and edges. In Figure 2, a different vertex correspondence between the graphs G_1 and G_2, of the previous example, is shown. This correspondence is achieved supposing that label "C" could substitute label "D" with a similarity 0.8. Then, using the same similarity function (the product of the similarities between vertices and between edges) a similarity value of 0.53 between G_1 and G_2 is obtained.

As we can see in Figures 1 and 2, there could be several correspondences between two graphs. Therefore, $sim_{max}(G_1, G_2)$ is defined as the highest similarity value among all the

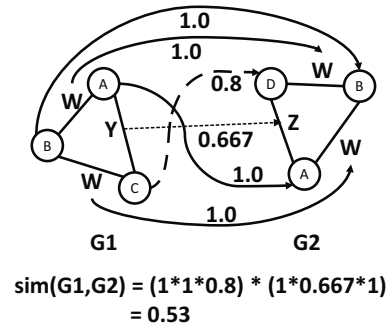

Figure 2: Example of an alternative similarity between two graphs.

possible correspondences between G_1 and G_2.

Using this definition of similarity between two graphs, a definition of support that allows inexact graph matching can be defined. Let $D = \{G_1, \ldots, G_{|D|}\}$ be a graph collection and G be a labelled graph in L, the support (denoted by *appSupp*) value of G in D, in terms of the similarity, is obtained through equation (2):

$$appSupp(G, D) = \frac{\sum_{G_i \in D} sim_{max}(G, G_i)}{|D|} \tag{2}$$

Using (2), when $appSupp(G, D) \geq \delta$, for a given support threshold δ, then G is a *frequent approximate subgraph* (FAS) in D. The value of the support threshold δ is in $[0, 1]$ because the similarity is defined in $[0, 1]$. *Frequent approximate subgraph mining* consists in finding all the FASs in a graph collection D, using a similarity function sim and an support threshold δ.

3. Related work

In the literature, only one work which aims to avoid using graph patterns that do not positively contribute to the classification task under the image classification approach based on FAS mining, has been proposed (Acosta-Mendoza et al., 2013). This approach combines FAS mining with attribute selection algorithms for reducing the attribute dimensionality in the image classification tasks. In this work, the authors represented an image collection based on an n-dimensional vector approach using graph patterns, where the attributes of

the vector are the patterns. Next, they apply an attribute selection algorithm for reducing the dimensionality of the vector; finally the classification task is performed over the reduced representation. In (Acosta-Mendoza et al., 2013) a good dimensionality reduction was achieved, and at the same time the classification results were also improved.

On the other hand, an important tool for performing the aforementioned image classification task is a FAS mining algorithm. Several algorithms for FAS mining in graph collections have been developed based on different similarity functions for graph matching (Holder et al., 1992; Song and Chen, 2006; Xiao et al., 2008; Zhang and Yang, 2008; Zou et al., 2009; Jia et al., 2011; Acosta-Mendoza et al., 2012a; Gago-Alonso et al., 2013), but only two of them are based on substitution probabilities keeping the graph topology (Jia et al., 2011; Acosta-Mendoza et al., 2012a). In these algorithms some vertex or edge labels can be replaced by other labels following some semantic substitution criteria according to the application context. In APGM, only vertex label variations are allowed, while VEAM is able to deal with variations on vertex and edge labels. Both algorithms have been successfully used for image classification.

In (Gago-Alonso et al., 2013), the usefulness of this approach using another kind of patterns, specifically geometric patterns, as attributes for classification is evaluated and an algorithm for computing this kind of patterns is introduced. In order to improve the efficiency of algorithms for mining FASs (as APGM and VEAM), two novel prunes were introduced in (Acosta-Mendoza et al., 2012b). Later, in (Acosta-Mendoza et al., 2012c), a way for automatically computing vertex and edge label substitutions based on image features is proposed. Finally, (Acosta-Mendoza et al., 2012c) was extended in (Morales-González et al., 2014), where a new criterion for selecting the similarity threshold for the mining step is also suggested. These works are key steps for the image classification approach based on FAS mining, since both vertex and edge label substitutions and the similarity threshold are parameters, which are difficult to be fixed by the user. However, an important aspect to explore is the study of the patterns that are really useful for classification; since many of the patterns could be redundant or useless for classification, when they are not representative of a specific class. Furthermore, the number of frequent approximate patterns may become

large, which negatively affects the performance of the classifiers.

In this paper, we propose using emerging patterns as attributes in order to reduce the set of FASs used as attributes for representing an image collection. To the best of our knowledge, this paper is the first that uses FAS mining together with emerging patterns for image classification.

4. Using emerging patterns as attributes

Given a set of images where each image is represented as a graph, the image classification approach based on FAS mining consists in finding, into the training set, a set of patterns to be used as attributes in a vectorial representation for describing all images in the training set. Using this vectorial representation, a traditional classifier can be used. In this way, for classifying a new image, it must be described as a vector in terms of the same attributes (patterns) used to describe the images in the training set. The number of patterns computed by a FAS miner could be very large; producing a high dimensionality in the image description, which could affect negatively the classification performance. For this reason, we propose to reduce the amount of patterns used for describing the images into this image classification approach, which allows improving both performance and classification results. This reduction is needed because there are several patterns that do not help to separate the classes. Into the whole set of patterns, an interesting subset is the one formed by the emerging patterns; since these patterns, by definition, are those capturing significant changes and differences between classes (Dong and Li, 1999). Thus, we propose the use of emerging patterns to reduce the representation of an image collection.

Let $D = \{G_1, \ldots, G_{|D|}\}$ be a collection of graphs, let $C = \{c_1, \ldots, c_{|C|}\}$ be a class collection where each class contains several graphs of D, and let G be a FAS of D, G is an *emerging pattern* for c_i if $appSupp(G, c_i) \geq \gamma$ and $appSupp(G, D - c_i) < \gamma$, for a given threshold $\gamma \in (0, 1)$ (Dong and Li, 1999; Garcia-Borroto, 2010; Kong et al., 2013).

Figure 3 shows a graphical example of emerging patterns for an object set categorized in three classes ($C1$, $C2$ and $C3$). The intersection of each coloured region (pattern) with each class, represents the percentage of occurrences of the pattern in the class. As we can see,

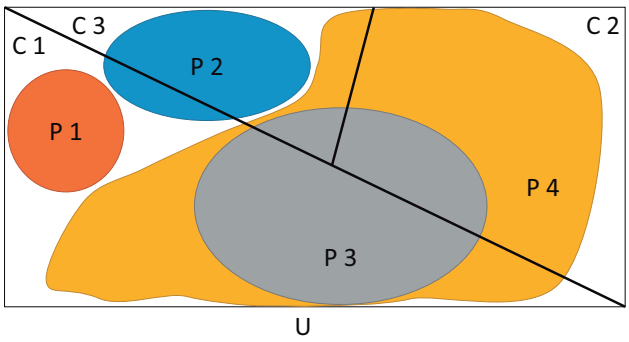

Figure 3: Example of emerging patterns for an object set categorized in three classes ($C1$, $C2$ and $C3$).

pattern "$P1$" appears exclusively into the class $C1$, therefore, it is an emerging pattern for this class. Patterns "P2" and "P3" are emerging patterns for classes $C3$ and $C1$, respectively, supposing that γ =0.4. That is, more than 40% of their occurrences are in a class and the rest of their occurrences, do not exceed 40% out of the class. On the other hand, pattern "P4" does not meet this definition since it appears in several classes with more than 40% of its occurrences.

Following to (Acosta-Mendoza et al., 2012a), there are several steps to achieve the final classification results starting by the representation of each image as a graph. In this paper, we propose adding a new step between the pattern extraction and the vector representation steps (see Figure 4), for selecting patterns useful for discriminating among classes (emerging patterns). The steps for our image classification approach based on FAS mining are as follows:

1. Representation:

   In this step, each image in a collection is represented as a graph. Several algorithms have been reported in the literature for representing images as graphs, such as: based on quad-trees (Acosta-Mendoza et al., 2012a), based on irregular pyramids (Morales-González and García-Reyes, 2013) and based on object skeletons (Pinilla-Buitrago et al., 2013), among others. Notice that, the time/space complexity needed for building the graph representation is very smaller than for mining patterns in the next step.

2. Pattern extraction:

   For extracting patterns from images represented as graphs, any algorithm for mining FASs from a graph collection can be used, for example: APGM (Jia et al., 2011), and VEAM (Acosta-Mendoza et al., 2012a), among others.

3. Pattern reduction:

   Unlike other works reported in the literature, for representing each class of the image collection, we propose to select, from the mined patterns, the emerging patterns. For selecting all emerging patterns, the support of each mined pattern is computed for each class and the definition of emerging pattern is verified.

4. Vector representation:

   Following the image classification approach based on FAS mining, for each graph (image) a vector representation is generated as in (Acosta-Mendoza et al., 2012a), but using only those FASs identified as emerging patterns.

5. Classification:

   Using the vector representation of the graphs (images), a conventional classifier is trained. In this way, for classifying a new image, first this image must be represented as a graph, then the emerging patterns, selected in the step 4, must be searched in the graph in order to build its vector representation. Finally, the new image (represented as a vector) is classified using the trained classifier. Notice that, in this step, the reduction of the number of patterns is very important, because it is not necessary to use the whole set of patterns for building the vector representation of the image, but only the emerging ones.

In summary, once we have the graph-based image representation for all the images in the collection, the FAS are computed using a FAS miner. Later, into the pattern reduction step, emerging FASs are selected. Then, a vector representation of the original images is built using only emerging FASs instead of using all the FASs. Thus, the dimension of the

vectors is determined by the number of emerging patterns in the training collection. Finally, these feature vectors are given to a classifier.

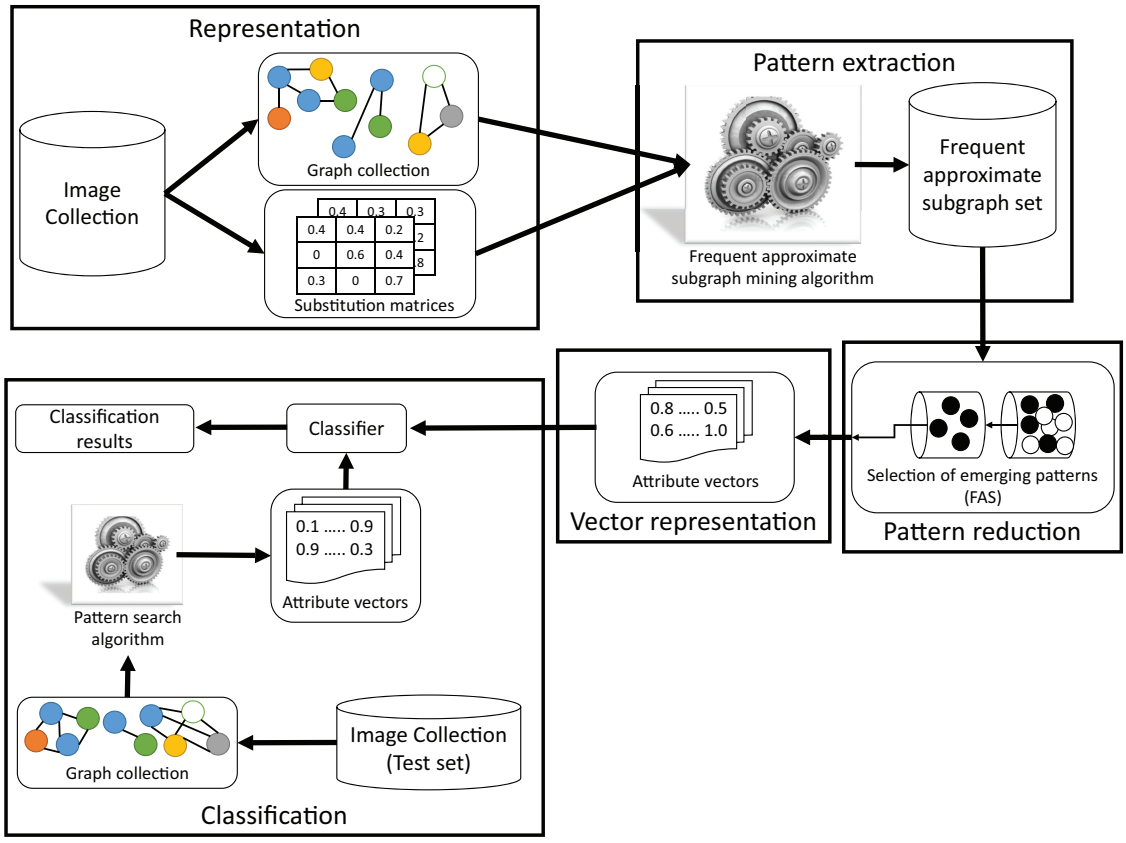

Figure 4: Work-flow of our proposal, where we introduce the pattern reduction step into a traditional graph-based image classification approach based on FAS mining.

It is important to highlight that in (Acosta-Mendoza et al., 2012a) the dimensionality of the attribute vectors is n (number of identified patterns), while in our proposal this dimension is k, where k is the number of emerging patterns, which is much less than n. Then, the computational cost of our proposal is lesser than the cost of the method reported in (Acosta-Mendoza et al., 2012a).

5. Experiments

In this Section, some experiments to show the efficacy of our proposal are presented. These experiments include a comparison between our proposal, which uses only emerging

patterns, and the one proposed in (Acosta-Mendoza et al., 2012a) that uses all the patterns computed by a FAS mining algorithm for representing images as attribute vectors. Besides, our approach is compared, in terms of efficacy, against attribute selection algorithms reported in the literature. In our experiments, the measures used for evaluating the classification results are precision, recall, f-measure and area under ROC, which have been commonly used in the literature (Huang, 2011; Jung and Kim, 2012; Mishra et al., 2012; Huang and Chen, 2013). Our experiments are performed using several synthetic image databases (see Section 5.1), an image skeleton database (see Section 5.2), and a real image collection (see Section 5.3). All our experiments were carried out using a personal computer with a 64 bits Intel(R) Core(TM)2 Quad CPU Q9300 2.50GHz and 8 Gb of RAM.

Based on the results reported in (Acosta-Mendoza et al., 2012a), the use of patterns taking into account semantic variations in the vertices and edges, i.e. those computed by VEAM (Acosta-Mendoza et al., 2012a), is better for image classification than using those patterns found by APGM (Jia et al., 2011) and using those patterns found by exact algorithms (Yan and Huan, 2002; Gago-Alonso et al., 2010). For this reason, in this paper we select the VEAM algorithm to be used in our experiments.

In order to show that the proposed method does not depend on a particular classifier, we use different classifiers in our experiments. These classifiers are of different nature: Support Vector Machine (SVM), decision trees (J48graft), and regression (ClassificationViaRegression). All these classifiers, were taken from Weka v3.6.6 (Hall et al., 2009) using their default parameters.

On the other hand, the value for the support threshold used in our experiments was selected experimentally by testing values from 20% to 60% with increments of 5 and selecting those values that allowed computing all patterns of an image collection in a reasonable time. This selection process was performed taking into account that when the value of δ is lower, the FAS mining process is computationally harder (in terms of time and memory). Beside, in our experiments, we used two other thresholds, which have an important role, the similarity threshold τ, and the emerging pattern threshold γ. The value for the similarity threshold is different for each image collection used in our experiments as it will be specified later. Also,

for choosing the best value for the emerging pattern threshold, we tested values from $\gamma = 0$ to $\gamma = 1$ with increments of 0.1. Notice that when the value of γ is lower, less emerging patterns are identified.

5.1. Synthetic databases

With the purpose of comparing the results obtained by our proposal against those reported in (Acosta-Mendoza et al., 2012a), we used the same image collection. This collection consists of 700 images obtained using the Random image generator of F. Coenen[1], which are split in two classes "landscape" and "seascape", represented as graphs using a quad-trees approach (Finkel and Bentley, 1974). In this image collection, each image is split into four quadrants with equal dimensions and recursively each quadrant is split until the maximal depth level threshold is achieved or the color of the quadrant is homogeneous. Once we have completed the quad-tree of an image, the leaves of this quad-tree are used as vertices for the graph that represents the image, where the predominant color of each leaf is used as its attribute (label), and each vertex is connected, by an edge, to its quadrant neighbors. The angle respect to the horizontal axis is used as attribute (label) for each edge. Figure 5 shows an example of this process. Detailed information about the representation of images following this approach can be seen in (Acosta-Mendoza et al., 2012a).

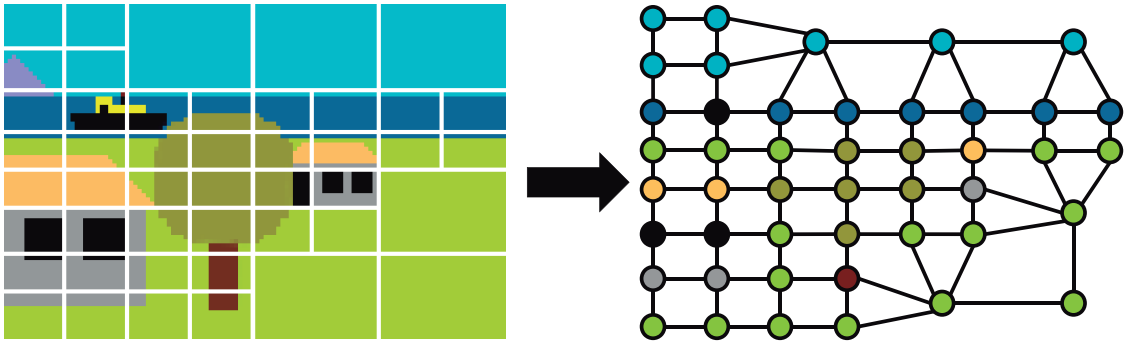

Figure 5: Example of an image represented as a graph using a quad-tree approach.

This synthetic collection was divided into six sub-collections with different sizes: from 200 to 700 images with an increment of 100 images. All graphs in these collections contain

[1]www.csc.liv.ac.uk/~frans/KDD/Software/ImageGenerator/imageGenerator.html

18 vertex labels, 24 edge labels and an average graph size ranging between 43 to 47 regarding the number of edges, and ranging between 24 to 27 regarding the number of vertices.

Once we have the images represented as graphs, the VEAM algorithm is used for extracting all patterns (FASs) from each collection. The VEAM algorithm needs the support threshold δ and the similarity threshold τ. For these collections, the similarity (0.4) and support (20%) threshold values were the same as in (Acosta-Mendoza et al., 2012a). This fact allows us to perform a fair comparison against the results reported in (Acosta-Mendoza et al., 2012a). Later, from the FASs, the emerging patterns are selected to represent each class in the collection. Finally, using these patterns (emerging patterns), obtained with $\gamma = 0.3$, 0.4 and 0.5, we build the attribute vectors which will be used by the classifiers in the classification step. In these experiments, several values of γ were tested (0.1, 0.2, 0.3, 0.4, 0.5, 0.6 and 0.7) but the best results were achieved with $\gamma = 0.3$, 0.4 and 0.5. Notice that for these experiments we used 10-fold cross-validation.

In Table 1 and henceforth, the method that uses all FASs as attributes for classification and the method that uses the emerging patterns with $\gamma = 0.3$, 0.4 y 0.5 will be represented as "All", "E(0.3)", "E(0.4)" and "E(0.5)", respectively. The first column of this table shows the collection identifiers. The other four consecutive columns show the number of patterns used as attributes for classification. The first of these four columns is the number of patterns computed by the FAS mining algorithm (all patterns), and the other three columns show the number of emerging patterns obtained using γ =0.3, γ =0.4 and γ =0.5.

Table 1: Average number of patterns used as attributes in the classification process using 10-fold cross-validation, for the synthetic databases with support threshold $\delta = 20\%$.

		Emerging patterns		
Collection	**All**	E(0.3)	E(0.4)	E(0.5)
Synthetic-200	340	256	133	74
Synthetic-300	374	299	139	76
Synthetic-400	433	292	150	78
Synthetic-500	453	357	140	70
Synthetic-600	498	321	149	69
Synthetic-700	864	326	143	65

An interesting characteristic of our proposal is the reduction in the amount of patterns, it uses as attributes (emerging patterns) to represent images in a collection, compared against using all the patterns found by VEAM. We can see from Table 1 that the dimensionality of the attribute vectors is drastically reduced if the images are represented using emerging patterns. Based on this fact, we can assure that regardless of the classifier we use this reduced-representation will improve the classifier's efficiency.

In our second experiment, we compare the efficacy of our proposal against using all the mined FAS as attributes for classification. Table 2 shows the average classification results (in terms of precision, recall, f-measure and area under ROC) using all FASs and the emerging patterns obtained with $\gamma = 0.3$, 0.4 and 0.5 as emerging pattern threshold. The first column of this table shows the metric used for evaluating the classification results (precision, recall, f-measure and area under ROC). The following groups of four consecutive columns show the results of all, E(0.3), E(0.4) and E(0.5); obtained by the classifier specified on the top of these columns. The best results appear boldfaced.

Table 2: Classification average results (%) achieved using $\delta = 20\%$ and different classifiers in synthetic graph (image) collections with different values of γ using emerging patterns and all patterns as attributes.

	SVM				**Regression**				**J48graft**			
Metric	All	E(0.3)	E(0.4)	E(0.5)	All	E(0.3)	E(0.4)	E(0.5)	All	E(0.3)	E(0.4)	E(0.5)
Precision	96.35	**96.81**	96.56	94.25	94.79	95.13	**95.66**	91.88	94.01	94.41	**94.46**	93.18
Recall	95.12	**95.80**	**95.80**	90.74	93.50	**94.10**	92.94	92.70	92.80	93.48	**93.83**	92.32
F-measure	95.73	**96.30**	96.13	92.46	94.14	**94.61**	94.28	92.29	93.04	93.94	**94.14**	92.75
ROC	96.41	96.81	**96.86**	94.51	98.60	98.43	**99.44**	98.79	95.39	95.89	96.36	**96.64**

As we can see in Table 2, the results achieved by our proposal E(0.3), E(0.4) and E(0.5) using different classifiers are better than those results obtained using all the FASs.

In addition, in Table 3 we present a statistical test for all pairwise comparisons between our proposal using emerging patterns (for different values of γ) and the option of using all patterns computed by VEAM. For this comparison, we use *Bergmann* (Bergmann and Hommel, 1988) as statistical test with $\alpha = 0.05$. In Table 3, the symbol "–" indicates that there is not a statistical significant difference between the results of the approaches under comparison; otherwise the winner approach is shown.

Table 3: Bergmann test for different classifiers in several graph (image) collections using emerging patterns and all patterns as attributes. Each cell of this table indicates which option was statistically better (among the methods compared in the column 1), "–" indicates no statistically significant difference.

Methods	SVM	Regression	J48graft
All vs. E(0.3)	–	–	–
All vs. E(0.4)	–	–	–
All vs. E(0.5)	All	E(0.5)	–
E(0.3) vs. E(0.4)	–	–	–
E(0.3) vs. E(0.5)	–	–	–
E(0.4) vs. E(0.5)	E(0.4)	–	–

As we can see from Table 3, the use of emerging patterns is not significant different to the use of all the FAS, but our proposal reduces the number of patterns (attributes) used to represent the images so that we need less than 50% of the total patterns computed by VEAM. Additionally, in 33.33% of the results E(0.4) is significantly better than E(0.5), while E(0.3) and E(0.5) were not significantly better than E(0.4) in any case. Thus, taking into account the balance between efficacy and dimensionality reduction, the best option for this collection is E(0.4).

The overall results show that our proposal, which uses the emerging patterns as attributes for representing the images in the collection, provides better outcomes than using all of the patterns computed by VEAM as attributes.

5.2. Image skeleton database

The database used in our second experiment was built from articulated shape skeletons (*ArSS*), including 36 image shapes divided into 9 classes: *elephant*, *fork*, *heart*, *horse*, *human*, *L-star*, *star*, *tortoise*, and *whale*, 4 images per class (see Figure 6). Due to the number of images per class, for this experiment we used 4-fold cross-validation, thus we take an image per class to create a test set with 9 image per fold. This process is repeated 4 times, each time taking a different image for each class.

In the representation module of our approach, the skeleton that will represent each image of the database was semi-automatically built. The program presented in (Bai and Latecki,

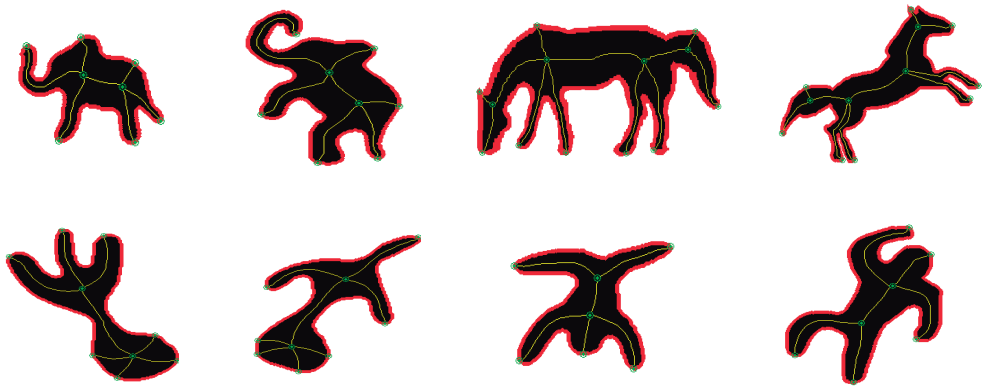

Figure 6: Example of image shapes in the ArSS database.

2007) and provided by Bai[2] was used to obtain an approximation of the medial axis of the figures based on the radios of the maximal disks inscribed inside of a figure. Then, manually, the branches which do not represent the structural form of the figure were pruned, and a few branches that were not built by the algorithm were added. Finally, simple skeletons that represent the structural information of the figures of the images were obtained from this semi-automatic process (see Figure 7).

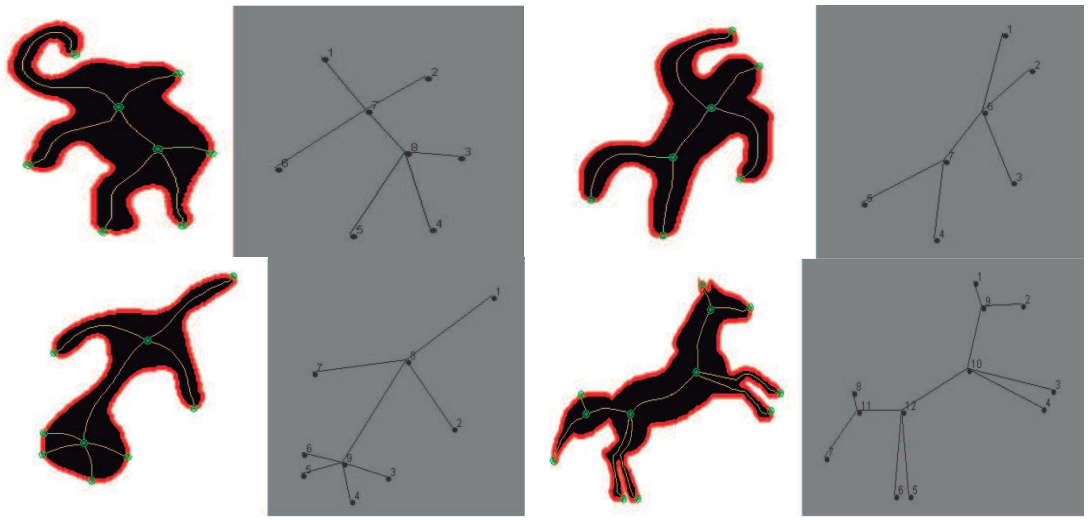

Figure 7: Example of the image skeletons for the real shapes collection, and the graphs obtained from them.

[2]https://sites.google.com/site/xiangbai/

Once we have the image skeletons, these are treated as graphs and the edges are labelled with the distance between vertices. The collection has 14 vertex labels (*trunk*, *ear*, *loin*, *tail*, *paw*, *head*, *body*, *tip*, *handle*, *butt*, *nose*, *arm*, *leg* and *fin*), 211 edge labels, with an average graph size of 6 in term of the number of edges, and an average graph size of 7 in term of the number of vertices. In this case, the vertex labels cannot be replaced by any other label, they can only be substituted by themselves. Since the edge labels are the distances between vertices they are connecting, there are possible substitutions among these labels. The substitution probabilities are computed following equation (3):

$$m_{i,j} = \frac{|d_i - d_j|}{|d_0 - d_n|} \tag{3}$$

Where d_i and d_j are the labels of the edges i and j, respectively, and d_0 and d_n are the smallest and the biggest labels (distance values) from the label set, respectively.

In the pattern extraction step, several values for γ, different from those used on the experiments of Section 5.1, were tested; because the threshold values used in the previous experiment did not identify any pattern over the current database. The values used for this database were $\gamma = 0.7$, 0.8 and 0.9 for several support threshold values δ. The similarity threshold (τ) was selected taking into account the edge label set used in the experiment; choosing 0.4 as similarity threshold value, which is the average similarity among the edge labels of the collection.

Table 4: Average of patterns used as attributes in the classification process using 4-fold cross-validation, for the ArSS database.

		Emerging patterns		
Support (δ)	**All patterns**	E(0.9)	E(0.8)	E(0.7)
25%	4873358	143660	101086	1408
30%	147113	19002	39	1
35%	381	71	0	0
40%	277	41	0	0

In Table 4, the first column shows the support threshold values used by VEAM. The other five consecutive columns show the number of patterns used as attributes for classification.

The first of these five columns represents the total number of patterns computed by VEAM, and the other four columns show the number of emerging patterns obtained using γ =0.9, 0.8 and 0.7 by the pattern reduction step of our proposal. In this experiment, the support threshold ranges from 25% to 40%, because the patterns obtained with thresholds smaller than 25% are not useful for classification since the low support jointly with the semantic variations in the vertices and edges allowed by VEAM introduce too much noise in the data representation, while the number of patterns identified with thresholds over 40% is too small, therefore a dimensionality reduction is not needed. Finally, from Table 4 we can see that the dimensionality is drastically reduced if images are represented using emerging patterns.

In the next experiment, using the classification step of our proposal, we compare the classification results (precision, recall, f-measure and area under ROC) reached by the reduced-representation of the images obtained by our proposal against the results obtained by the representation using all FASs as attributes for classification.

In Table 5, the classification results (in terms of precision, recall, f-measure and area unde ROC) for this experiment are summarized. The first and second column of this table show the support threshold values and the metric used for evaluating the classification results (precision, recall, f-measure and area unde ROC), respectively. In the following groups of four consecutive columns, the results of all, E(0.9), E(0.8) and E(0.7) are shown; obtained by the classifier specified on the top of these columns. The symbol "–" indicates that the classifier was not able to obtain a classification result because for those values of γ it was not possible to get emerging patterns (attributes) for classification. The symbol "*" indicates that the classifier was not able to obtain a classification result because there were too many attributes that make classification unfeasible. The best results appear boldfaced.

As we can see from Table 5, the results obtained by different classifiers using emerging patterns as attributes are as good as those obtained by using all the FASs. These results show that using emerging patterns as attributes for image classification allows obtaining good classification results, but using a much smaller set of attributes.

In addition, in Table 6 we present a statistical test for all pairwise comparisons between our proposal using emerging patterns (for different values of γ) and the option of using

Table 5: Classification average results (%) achieved using different classifiers in the ArSS collection with different values of δ and γ using emerging patterns and all patterns as attributes.

		SVM				**Regression**				**J48graft**			
Support	**Metric**	All	E(0.9)	E(0.8)	E(0.7)	All	E(0.9)	E(0.8)	E(0.7)	All	E(0.9)	E(0.8)	E(0.7)
	Precision	*	72.22	**91.67**	47.69	*	**83.33**	**83.33**	57.41	*	**75.93**	67.59	75.00
25%	Recall	*	88.89	**94.44**	58.33	*	80.56	**88.89**	66.67	*	80.56	77.78	**83.33**
	F-measure	*	85.19	**92.59**	50.93	*	75.00	**85.19**	60.19	*	77.31	70.83	**77.78**
	ROC	*	87.88	**97.23**	91.70	*	84.75	**95.48**	92.35	*	**89.78**	85.95	83.33
	Precision	**80.56**	72.22	15.09	11.76	**73.61**	73.15	15.09	5.42	64.35	**72.22**	15.56	15.14
30%	Recall	**86.11**	80.56	30.56	27.78	**80.56**	**80.56**	30.56	22.22	75.00	**80.56**	17.78	30.56
	F-measure	**82.41**	75.00	18.39	14.92	**75.46**	**75.46**	18.39	8.59	67.59	**75.00**	17.92	18.31
	ROC	90.30	**92.73**	79.01	75.01	89.95	**97.75**	77.08	74.30	**89.25**	84.23	70.85	74.48
	Precision	51.94	**91.67**	–	–	57.87	**58.10**	–	–	**61.11**	48.38	–	–
35%	Recall	63.89	**94.44**	–	–	**69.44**	**96.44**	–	–	**72.22**	61.11	–	–
	F-measure	55.09	**92.59**	–	–	61.11	**61.30**	–	–	**64.35**	52.04	–	–
	ROC	89.95	**93.58**	–	–	**89.60**	86.65	–	–	**87.50**	83.00	–	–
	Precision	38.43	**45.93**	–	–	57.18	**60.19**	–	–	59.03	**65.51**	–	–
40%	Recall	52.78	**58.33**	–	–	66.67	**72.22**	–	–	69.44	**75.00**	–	–
	F-measure	41.94	**49.07**	–	–	59.35	**63.89**	–	–	61.76	**68.24**	–	–
	ROC	88.73	**89.25**	–	–	89.25	**90.30**	–	–	**86.98**	85.78	–	–

all patterns computed by VEAM. For this comparison, we use *Bergmann* (Bergmann and Hommel, 1988) as statistical test with $\alpha = 0.05$. In Table 6, the symbol "–" indicates that there is not a statistical significant difference between the results of the approaches under comparison; otherwise the winner approach is shown.

Table 6: Bergmann test for different classifiers in the ArSS database with and without emerging pattern selection. Each cell of this table indicates which option was statistically better (among the classifiers compared in the column 1), "–" indicates no statistically significant difference.

Classifier	SVM	Regression	J48graft
All vs. E(0.7)	–	–	–
All vs. E(0.8)	–	–	–
All vs. E(0.9)	–	–	–
E(0.7) vs. E(0.8)	–	–	–
E(0.7) vs. E(0.9)	–	–	–
E(0.8) vs. E(0.9)	–	–	–

As we can see from Table 6, in general, classification results using emerging patterns are not significantly better than the results obtained by "All", but our proposal highly

reduces the number of patterns (attributes) used to represent images so that we need less attributes for classification in comparison with the total set of patterns computed by VEAM. Additionally, E(0.9) is better than E(0.8) and E(0.7) when $\delta < 30\%$, because E(0.9) is able to obtain classification results, while the other ones do not find out any emerging pattern (see the tables 4 and 5). However, the best balance between efficacy and dimensionality reduction for this collection is reached by E(0.8) with $\delta = 25\%$.

5.3. COIL database

The collection COIL (Nene et al., 2008) used in this experiment contains images of real objects taken from different viewpoints. In our experiment, we use the same 25 object classes used in (Morales-González and García-Reyes, 2013). COIL contains 1800 images divided into 198 (11%) images for training and 1602 (89%) images for testing. Each image of this collection is represented as a graph according to the representation proposed in (Morales-González and García-Reyes, 2013), where each image is hierarchically partitioned at different levels of resolution. Using these partitions, an irregular pyramid is built (Brun and Kropatsch, 2001; Kropatsch et al., 2005), where each level is represented as a region adjacency graph. In this kind of graphs, each region is represented as a vertex and there is an edge between two regions (vertices) if they are adjacent. Finally, the image is represented by the graph of the best level of the pyramid; according to the metric proposed in (Morales-González and García-Reyes, 2013). In Figure 8, an example of this representation is shown. Detailed information about the representation of images under this approach can be seen in (Morales-González and García-Reyes, 2013).

The graphs obtained from the images of COIL have an average size of 55 vertices and an average size of 135 edges, using 150 vertex labels and 27 edge labels.

In the pattern reduction step, several values of γ are tested. The values used for this database were $\gamma = 0.7$, 0.8 and 0.9 using different support threshold values δ (see Table 7). The similarity threshold τ used for the COIL collection is the value that allows to achieve the best classification results in (Acosta-Mendoza et al., 2012c) (i.e. $\tau = 0.66$).

In Table 7, the first column shows the support threshold values used by VEAM. The other

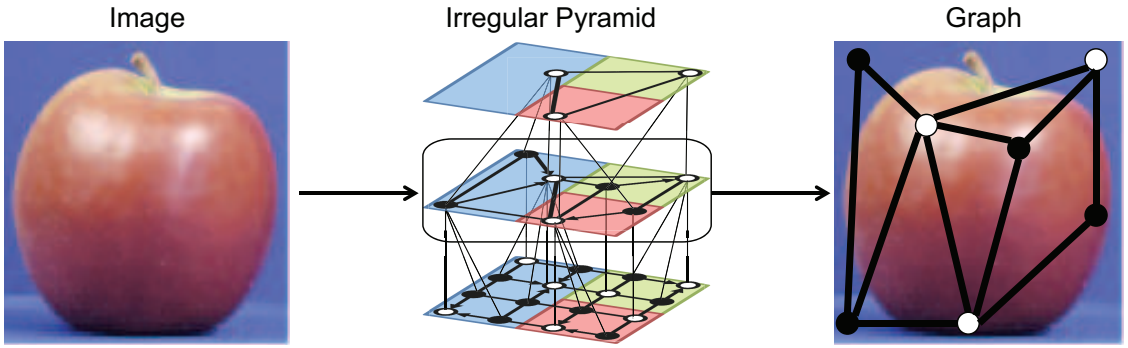

Figure 8: Example of an image represented as a graph using the irregular pyramid approach.

Table 7: Average of patterns used as attributes in the classification process, for the COIL database.

		Emerging patterns		
Support (δ)	**All patterns**	E(0.9)	E(0.8)	E(0.7)
30%	2668	664	208	3
40%	321	81	1	0
50%	91	15	0	0
60%	46	2	0	0

four consecutive columns show the number of patterns used as attributes for classification. The first of these four columns represents the total number of patterns computed by VEAM, and the other three columns show the number of emerging patterns obtained using $\gamma = 0.9$, 0.8 and 0.7 by the pattern reduction step of our method. As we can see in this table, a dimensionality reduction is not needed for $\delta \geq 50\%$ since there are only a few patterns, but for $\delta = 40\%$ and 30%, the dimensionality is drastically reduced when the images are represented using emerging patterns.

In Table 8, a comparison in terms of precision, recall, f-measure and area under ROC between our proposal and the method that uses all FASs as attributes for classification is shown. In this table, the first and second columns show the the support threshold values and the metric used for evaluating the classification results (precision, recall, f-measure and area unde ROC), respectively. In the next groups of four consecutive columns, the results obtained by the classifier specified on the top of these columns using all or emerging patterns as attributes for classification are shown. The emerging patterns are obtained with $\gamma = 0.9$

(E(0.9)), 0.8 (E(0.8)) and 0.7 (E(0.7)), respectively. The symbol "*" indicates that the classifier was not able to obtain a classification result because for those values of γ it was not possible to get emerging patterns (attributes) for classification. The best results appear boldfaced.

Table 8: Classification average results (%) achieved using different classifiers in the COIL collection with different values of δ and γ using emerging patterns and all patterns as attributes.

		SVM				**Regression**				**J48graft**			
δ	**Metric**	All	E(0.9)	E(0.8)	E(0.7)	All	E(0.9)	E(0.8)	E(0.7)	All	E(0.9)	E(0.8)	E(0.7)
30%	Precision	92.28	**94.06**	81.01	4.39	75.94	**80.55**	68.87	10.97	**90.89**	89.44	64.83	25.63
	Recall	92.08	**93.88**	77.26	13.06	74.40	**80.25**	67.18	17.22	**90.39**	88.47	59.14	22.61
	F-measure	92.06	**93.86**	77.59	6.00	72.54	**78.92**	66.14	12.14	**90.38**	88.38	59.38	21.04
	ROC	98.30	**99.00**	97.70	70.40	96.60	**96.80**	95.70	72.30	**94.60**	91.20	82.40	72.10
40%	Precision	**86.43**	76.04	5.02	*	72.05	**91.63**	4.79	*	77.11	**90.95**	4.79	*
	Recall	**84.59**	73.56	15.31	*	72.93	**91.30**	15.31	*	71.19	**90.14**	15.31	*
	F-measure	**84.71**	72.80	6.31	*	70.85	**91.26**	6.18	*	70.04	**90.20**	6.18	*
	ROC	**97.90**	97.20	78.20	*	97.60	**97.90**	81.50	*	86.70	**96.40**	81.10	*
50%	Precision	**80.05**	79.02	*	*	76.46	**77.39**	*	*	80.05	**84.86**	*	*
	Recall	73.31	**77.61**	*	*	73.02	**76.18**	*	*	73.31	**82.59**	*	*
	F-measure	71.89	**75.77**	*	*	71.51	**73.69**	*	*	71.89	**82.49**	*	*
	ROC	88.90	**93.20**	*	*	96.50	**96.60**	*	*	88.90	**95.36**	*	*
60%	Precision	**71.06**	33.69	*	*	**72.71**	9.08	*	*	**80.64**	13.04	*	*
	Recall	**67.00**	38.39	*	*	**72.47**	22.25	*	*	**76.87**	26.88	*	*
	F-measure	**64.05**	31.87	*	*	**70.96**	11.77	*	*	**75.77**	16.12	*	*
	ROC	**94.60**	83.40	*	*	**94.90**	83.20	*	*	**88.80**	85.50	*	*

As we can see from Table 8, the results obtained by different classifiers using emerging patterns as attributes are competitive regarding those obtained by using all the FASs. These results show that using emerging patterns as attributes for image classification allows obtaining good classification results, but using a smaller set of attributes.

In addition, in Table 9 we present a statistical test for all pairwise comparisons between our proposal using emerging patterns (for different values of γ) and the option of using all patterns computed by VEAM. For this comparison, we use *Bergmann* (Bergmann and Hommel, 1988) as statistical test with $\alpha = 0.05$. In Table 3, the symbol "–" indicates that there is not a statistical significant difference between the results of the approaches under comparison; otherwise, the winner approach is shown.

As we can see from the tables 8 and 9, in general, classification results using emerging patterns are not significantly better than the results obtained by "All", but our proposal

Table 9: Bergmann test for different classifiers in the COIL collection using emerging patterns and all patterns as attributes. Each cell of this table indicates which option was statistically better (among the classifiers compared in the column 1), "–" indicates no statistically significant difference.

Classifier	SVM	Regression	J48graft
All vs. E(0.7)	All	All	All
All vs. E(0.8)	All	All	All
All vs. E(0.9)	–	–	–
E(0.7) vs. E(0.8)	E(0.8)	E(0.8)	E(0.8)
E(0.7) vs. E(0.9)	E(0.9)	E(0.9)	E(0.9)
E(0.8) vs. E(0.9)	E(0.9)	E(0.9)	E(0.9)

reduces the number of patterns (attributes) used to represent images so that we need less attributes for classification in comparison with the total set of patterns computed by VEAM. Therefore, from the results of our experiments, in this database for $E(0.9)$ with $\delta = 30\%$ got the best results.

5.4. Emerging patterns vs. attribute selection algorithms

As we mentioned in Section 3, in (Acosta-Mendoza et al., 2013) the authors report a method for selecting FAS patterns by applying attribute selection algorithms, this work is the closest approach to our method. For this reason, in this section, a comparison between our proposal and the method reported in (Acosta-Mendoza et al., 2013) was performed over the three previously used collections (synthetic, skeleton and COIL).

In the following experiment we compare the efficacy of our proposal against attribute selection methods. For our comparison, we used the attribute selection algorithms chi-squared (CHI-Q), information gain (IG) and information gain ratio (GRAE), which obtained the best results in (Acosta-Mendoza et al., 2013). In the synthetic collections, we used the average number of patterns reported in (Acosta-Mendoza et al., 2013) (i.e. 33.67% for CHI-Q, 34.53% for GRAE and 38.81% for IG). For the other two collections (skeleton and COIL) we used the same number of patterns obtained by our proposal in the previous experiments (i.e. 2.07% for skeleton and 24.89% for COIL databases). Furthermore, for our method, we used the emerging pattern threshold γ and the support threshold δ that

produced the best dimensionality reduction jointly with good classification results in the previous experiments: $\delta = 20\%$ and E(0.4) for the synthetic collections; $\delta = 25\%$ and E(0.8) for the skeleton collection; and $\delta = 30\%$ and E(0.9) for the COIL collection.

The classification results (in terms of precision, recall, f-measure and area under ROC, respectively) for three classifiers (SVM, Regression and J48graft) using both, emerging patterns and attribute selection algorithms are summarized in Table 10. This table is split into three sub-tables; showing the results achieved over: (a) the synthetic collections, (b) skeleton (ArSS) and (c) COIL. The first column of these tables shows the metric used for evaluating the classification results. The next group of columns show the classification results achieved by using emerging patterns, CHI-Q, GRAE and IG, respectively, with the classifier specified on the top of each group of columns. The best results appear boldfaced.

Table 10: Classification average results (%) achieved using different classifiers in several graph (image) collections with emerging pattern selection and using CHI-Q, IG and GRAE.

(a) Using $\delta = 20\%$ and $\gamma = 0.4$ for the emerging pattern selection over the synthetic collections.

	SVM				**Regression**				**J48graft**			
Metric	Emerging	CHI-Q	GRAE	IG	Emerging	CHI-Q	GRAE	IG	Emerging	CHI-Q	GRAE	IG
Precision	**96.56**	94.34	94.88	94.33	**95.66**	95.35	94.00	94.45	**94.46**	94.20	94.03	93.76
Recall	**95.80**	93.28	93.19	93.57	**92.94**	90.71	91.64	92.28	93.83	94.61	94.76	**94.87**
F-measure	**96.13**	93.59	93.85	93.77	**94.28**	92.69	92.60	93.14	94.14	94.23	94.27	**94.31**
ROC	**96.86**	94.46	94.64	94.64	**99.44**	98.01	98.62	98.47	**96.36**	96.11	96.29	95.81

(b) Using $\delta = 25\%$ and $\gamma = 0.8$ for the emerging pattern selection over the ArSS collection.

	SVM				**Regression**				**J48graft**			
Metric	Emerging	CHI-Q	GRAE	IG	Emerging	CHI-Q	GRAE	IG	Emerging	CHI-Q	GRAE	IG
Precision	**91.67**	83.33	41.67	83.33	**83.33**	72.22	58.33	55.56	**67.59**	66.67	58.33	66.67
Recall	**94.44**	88.89	55.56	88.89	**88.89**	77.78	66.67	66.67	**77.78**	**77.78**	66.67	**77.78**
F-measure	**92.59**	85.19	45.19	85.19	**85.19**	74.07	60.00	59.26	**70.83**	70.37	60.00	60.00
ROC	**97.23**	94.40	89.60	95.80	95.48	**95.80**	91.70	91.00	85.95	**92.40**	91.70	91.70

(c) Using $\delta = 30\%$ and $\gamma = 0.9$ for the emerging pattern selection over the COIL collection.

	SVM				**Regression**				**J48graft**			
Metric	Emerging	CHI-Q	GRAE	IG	Emerging	CHI-Q	GRAE	IG	Emerging	CHI-Q	GRAE	IG
Precision	**94.06**	88.65	88.81	87.85	**80.55**	76.16	77.40	76.97	**89.44**	82.74	80.83	76.51
Recall	**93.88**	88.08	88.64	87.09	**80.25**	75.13	74.90	76.44	**88.47**	80.44	78.32	73.19
F-measure	**93.86**	87.99	88.33	86.99	**78.92**	73.87	74.55	75.06	**88.38**	80.01	78.20	72.25
ROC	**99.00**	**99.00**	**99.00**	98.90	**96.80**	96.50	96.10	96.40	**91.20**	89.80	88.70	87.20

In our experiments, the classification results achieved using emerging patterns are, in

most of the cases, better than those achieved by using attribute selection algorithms. It is important to comment that, although the best area under ROC results were obtained with the regression classifier, in general the best classification results were obtained with SVM. Additionally, it is important to remember that our proposal obtains the classification results in a more efficient way than the attribute selection algorithms. This is because, when we apply an attribute selection algorithm, this algorithm works with a vector representation using all the patterns, while by applying our proposal the emerging patterns are computed, and based on them, the vector representation is built. An example of this fact can be seen in the results of the skeleton collection using $\delta = 25\%$ and $\gamma = 0.8$, where, for the attribute selection algorithms, the vector representation with all attributes must be built before applying attribute selection, while using our proposal the attributes space is reduced in a 98% before building the vector representation.

Finally, we also compare our proposal against other methods that do not use patterns as attributes for classifying the image collections used in this section (Morales-González and García-Reyes, 2013; Pinilla-Buitrago et al., 2013). The results obtained using our method based on emerging patterns are better than those reported in (Morales-González and García-Reyes, 2013) over the COIL collection; using this method, the best F-measure value obtained is 92.19 while our method scored 93.86. For ArSS, the best accuracy reported in (Pinilla-Buitrago et al., 2013) is 88.80 while our proposal scored 94.44.

6. Conclusions

In this paper, we propose the use of frequent approximate subgraphs (FAS) together with emerging patterns for image classification. To the best of our knowledge, this is the first work where the impact of using emerging patterns in the FAS mining context is evaluated. In our experiments, we have contrasted our proposal against the use of attribute selection algorithms for reducing the dimensionality in the image representation based on FASs. From these experiments, we can conclude that both methods are competitive in classification quality and reduction, but our proposal obtains the classification results in a more efficient way than the attribute selection algorithms. This fact is because the attribute selection

algorithms must build the vector representation using all patterns, while our proposal only uses the emerging patterns. Even, as it was shown in our experiments, for some databases (for example ArSS database) the use of an attribute selector would be unfeasible due to the huge amount of patterns computed by FAS miners. Using emerging patterns as attributes, for representing the images in a collection, allows a drastic dimensionality reduction of the attribute vectors. Moreover, in most of our experiments the classification results were also improved with our proposal, compared against the results reached by using all the patterns computed by a FAS algorithm.

As future work, we will try to include the identification of emerging patterns inside the FAS mining process.

Acknowledgment

This work was partly supported by the National Council of Science and Technology of Mexico (CONACyT) through the scholarship grant 287045.

References

Acosta-Mendoza, N., Gago-Alonso, A., Carrasco-Ochoa, J., Martínez-Trinidad, J., Medina-Pagola, J., november 2013. Feature Space Reduction for Graph-Based Image Classification. In: Proceedings of the 18th Iberoamerican Congress on Pattern Recognition (CIARP'13). Vol. Part I, LNCS 8258. Springer-Verlag Berlin Heidelberg, Havana, Cuba, pp. 246–253.

Acosta-Mendoza, N., Gago-Alonso, A., Carrasco-Ochoa, J., Martínez-Trinidad, J., Medina-Pagola, J., 2015. A Nectar of Frequent Approximate Subgraph Mining for Image Classification. Cuban Journal of Informatics Sciences (RCCI) 9 (1).

Acosta-Mendoza, N., Gago-Alonso, A., Medina-Pagola, J., 2012a. Frequent approximate subgraphs as features for graph-based image classification. Knowledge-Based Systems 27, 381–392.

Acosta-Mendoza, N., Gago-Alonso, A., Medina-Pagola, J., 2012b. On speeding up frequent approximate subgraph mining. In: Proceedings of the 17th Iberoamerican Congress on Pattern Recognition (CIARP'12). Vol. LNCS 7441. Buenos Aires, Argentina, Springer-Verlag Berlin Heidelberg, pp. 316–323.

Acosta-Mendoza, N., Morales-González, A., Gago-Alonso, A., García-Reyes, E., Medina-Pagola, J., 2012c. Classification using frequent approximate subgraphs. In: Proceedings of the 17th Iberoamerican Congress

on Pattern Recognition (CIARP'12). Vol. LNCS 7441. Buenos Aires, Argentina, Springer-Verlag Berlin Heidelberg, pp. 292–299.

Anchuri, P., Zaki, M., Barkol, O., Golan, S., Shamy, M., 2013. Approximate graph mining with label costs. In: The 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2013. ACM, Chicago, IL, USA, August 11-14, pp. 518–526.

Aridhi, S., Lacomme, P., Ren, L., Vincent, B., 2015. A mapreduce-based approach for shortest path problem in large-scale networks. Engineering Applications of Artificial Intelligence 41, 151–165.

Bai, X., Latecki, L., 2007. Discrete Skeleton Evolution. In: Proceedings of the 6th International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR'07). Vol. LNCS 4679. Springer-Verlag, Berlin, pp. 362–374.

Bergmann, G., Hommel, G., 1988. Improvements of general multiple test procedures for redundant systems of hypotheses. In: P.Bauer, G. Hommel, and E. Sonnemann, editors, Multiple Hypotheses Testing. Springer, Berlin, pp. 100–115.

Brun, L., Kropatsch, W., 2001. Introduction to combinatorial pyramids. Digital and image geometry: advanced lectures, 108–128.

Dong, G., Li, J., 1999. Efficient mining of emerging patterns: Discovering trends and differences. In: Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Diego, Caliofornia, United States, pp. 43–52.

Finkel, R., Bentley, J., 1974. Quad Trees: A Data Structure for Retrieval on Composite Keys. Acta Informatica 4, 1–9.

Flores-Garrido, M., Carrasco-Ochoa, J., Martínez-Trinidad, J., 2014a. AGraP: an algorithm for mining frequent patterns in a single graph using inexact matching. Knowledge and Information Systems, 1–22.

Flores-Garrido, M., Carrasco-Ochoa, J., Martínez-Trinidad, J., 2014b. Mining maximal frequent patterns in a single graph using inexact matching. Knowledge-Based Systems 66, 166–177.

Gago-Alonso, A., Carrasco-Ochoa, J. A., Medina-Pagola, J. E., Martínez-Trinidad, J. F., August 2010. Full Duplicate Candidate Pruning for Frequent Connected Subgraph Mining. Integrated Computer-Aided Engineering 17, 211–225.

Gago-Alonso, A., Muñoz-Briseño, A., Acosta-Mendoza, N., 2013. A New proposal for Graph Classification using Frequent Geometric Subgraphs. Data and Knowledge Engineering (DKE) 87, 243–257.

Garcia-Borroto, M., 2010. Searching extended emerging patterns for supervised classification. In: Proceedings of the 10th international conference on Intelligent data engineering and automated learning. Computer Science Department, National Institute for Astrophysics Optics and Electronics, Puebla, Mexico.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I., 2009. The WEKA Data Mining Software: An Update. SIGKDD Explorations 11.

Holder, L., Cook, D., Bunke, H., 1992. Fuzzy substructure discovery. In: ML92: Proceedings of the ninth international workshop on Machine learning. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 218–223.

Huang, S., 2011. An advanced motion detection algorithm with video quality analysis for video surveillance systems. IEEE Transactions on Circuits and Systems for Video Technology 21 (1), 1–14.

Huang, S., Chen, B., 2013. Highly accurate moving object detection in variable bit rate video-based traffic monitoring systems. IEEE Transactions on Neural Networks and Learning Systems 24 (12), 1920–1931.

Jia, Y., Zhang, J., Huan, J., 2011. An efficient graph-mining method for complicated and noisy data with real-world applications. Knowledge and Information Systems 28 (2), 423–447.

Jin, N., Wang, W., April 2011. Lts: Discriminative subgraph mining by learning from search history. In: Proceedings of the 27th International Conference on Data Engineering (ICDE). Hannover, Germany, pp. 207–218.

Jung, C., Kim, C., 2012. A unified spectral-domain approach for saliency detection and its application to automathic object segmentation. IEEE Transactions on Image Processing 21 (3), 1272–1283.

Kong, X., Yu, P., Wang, X., Ragin, A., 2013. Discriminative feature selection for uncertain graph classification. In: Proceedings of Computing Research Repository. Vol. abs1301.6626. CoRR.

Kropatsch, W., Haxhimusa, Y., Pizlo, Z., Langs, G., 2005. Vision pyramids that do not grow too high. Pattern Recognition Letters 26, 319–337.

Mishra, A., Aloimonos, Y., Cheong, L., Kassim, A., 2012. Active visual segmentation. IEEE Transactions on Pattern Analysis and Machine Inteligence 34 (4), 639–653.

Moradi, P., Rostami, M., 2015. A graph theoretic approach for unsupervised feature selection. Engineering Applications of Artificial Intelligence 44, 33–45.

Morales-González, A., Acosta-Mendoza, N., Gago-Alonso, A., García-Reyes, E., Medina-Pagola, J., 2014. A new proposal for graph-based image classification using frequent approximate subgraphs. Pattern Recognition 47 (1), 169–177.

Morales-González, A., García-Reyes, E. B., 2013. Simple object recognition based on spatial relations and visual features represented using irregular pyramids. Multimedia tools and applications 63 (3), 875–897.

Nene, S., Nayar, S., Murase, H., 2008. Columbia object image library (coil-100). Structural, Syntactic, and Statistical Pattern Recognition, Joint IAPR International Workshop, SSPR and SPR 2008.

Pinilla-Buitrago, L., Martínez-Trinidad, J., Carrasco-Ochoa, J., 2013. New penalty scheme for optimal subsequence bijection. In: Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications CIARP 2013. Vol. Part I, LNCS 8258. Springer-Verlag Berlin Heidelberg, pp. 206–213.

Song, Y., Chen, S.-S., 2006. Item sets based graph mining algorithm and application in genetic regulatory networks. Data Mining, IEEE International Conference on Volume, Issue, 337–340.

Xiao, Y., Wu, W., Wang, W., He, Z., 2008. Efficient Algorithms for Node Disjoint Subgraph Homeomorphism Determination. In: Proceedings of the 13th international conference on Database systems for advanced applications. Springer-Verlag, Berlin, Heidelberg, New Delhi, India, pp. 452–460.

Yan, X., Huan, J., 2002. gSpan: Graph-Based Substructure Pattern Mining. In: International Conference on Data Mining. Maebashi, Japan.

Zhang, S., Yang, J., 2008. RAM: Randomized Approximate Graph Mining. In: The 20th International Conference on Scientific and Statistical Database Management. Hong Kong, China, pp. 187–203.

Zou, Z., Li, J., Gao, H., Zhang, S., 2009. Frequent subgraph pattern mining on uncertain graph data. In: CIKM'09: Proceeding of the 18th ACM conference on Information and knowledge management. ACM, New York, NY, USA, pp. 583–592.