150/2012

Raport Badawczy
Research Report

RB/39/2012

Classification problem in CBIR

T. Jaworska

Instytut Badań Systemowych
Polska Akademia Nauk

Systems Research Institute
Polish Academy of Sciences

POLSKA AKADEMIA NAUK

Instytut Badań Systemowych

ul. Newelska 6

01-447 Warszawa

tel.: (+48) (22) 3810100

fax: (+48) (22) 3810105

Kierownik Zakładu zgłaszający pracę:
Prof. dr hab. inż. Janusz Kacprzyk

Warszawa 2012

Classification problem in CBIR

Tatiana Jaworska
Polish Academy of Sciences, Systems Research Institute, Warsaw, Poland
Tatiana.Jaworska@ibspan.waw.pl

Abstract: *At present a great deal of research is being done in different aspects of Content-Based Image Retrieval (CBIR). Image classification is one of the most important tasks in image retrieval that must be dealt with. The primary issue we have addressed is: how can the fuzzy set theory be used to handle crisp image data. We propose fuzzy rule-based classification of image objects. To achieve this goal we have built fuzzy rule-based classifiers for crisp data. In this paper we present the results of fuzzy rule-based classification in our CBIR. Furthermore, these results are used to construct a search engine taking into account data mining.*

Keywords: *CBIR, spatial relationship, fuzzy systems, fuzzy rule-based classification, pattern recognition, image search engine.*

1. Introduction

In recent years, the availability of image resources and large image datasets has increased tremendously. This has created a demand for effective and flexible techniques for automatic image classification and retrieval. Although attempts to construct the Content-Based Image Retrieval (CBIR) in an efficient way have been made before, a major problem in this area, which is the extraction of semantically rich metadata from computationally accessible low-level features, still poses tremendous scientific challenges. Images and graphical data are complex in terms of visual and semantic contents. Depending on the application, images are modelled using their

- visual properties (or a set of relevant visual features),
- semantic properties,
- spatial or temporal relationships of graphical objects.

Over the last decade a number of approaches to CBIR have been proposed, e.g. Deb [6], Niblack et al. [16], Ogle and Stonebraker [18], Pons et al [19], Lee et al. [13], Berzal et al. [2]. Recently, Ali [1] has applied rough sets to image classification and retrieval.

Having analysed various CBIR system strengths and weaknesses, it seems necessary to introduce fuzzy information models into image retrieval, based on high-level semantic concepts that perceive an image as a complex whole. Zadeh's fuzzy set theory has allowed us to develop new programming tools, concerned with graphical applications and dealing with imperfect pictorial data [4]. Within the scope of semantic properties, as well as graphical object properties, the first successful attempt was made by Candan and Li [3], who constructed the Semantic and Cognition-based Image Retrieval (SEMCOG) query processor to search for images by predicting their semantic and spatial imperfection. Liu et al. [14] address the problem of narrowing down the 'semantic gap' that still exists in CBIR systems.

The classification problem is crucial for multimedia information retrieval in general, and for image retrieval in particular. There are a number of standard classification methods in use, some of which are briefly described below:

- A very simple classifier can be based on the *k*-nearest-neighbour approach. In this method, one simply finds in the *n*-dimensional feature space the closest objects from the training set to an object being classified. It is a type of instance-based learning, or lazy learning. The *k*-nearest neighbour algorithm is sensitive to the local data structure [5].
- A Support Vector Machine constructs a set of hyper-planes in a high-dimensional space which can be used for classification. Intuitively, good separation is achieved by the hyper-plane that has the largest distance (functional margin) to the nearest training data point of any class. If classes are linearly separable, a separating hyper-plane may be used to bisect the data. However, it is often so that the classes are linearly inseparable, then kernels are used to map non-linearly the input data to a high-dimensional space (feature space). The classes under this mapping may be then linearly separable [7].
- The Bayesian decision theory is the basis of statistical classification methods. It provides the fundamental probability model for well-known classification procedures such as the statistical discriminant analysis. A naive Bayes classifier assumes that the presence (or absence) of a particular feature of a class is unrelated to the presence (or absence) of any other feature, given the class variable. Depending on the precise nature of the probability model, naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In spite of their oversimplified assumptions, naive Bayes classifiers have worked quite well in many complex real-world situations [20].
- Neural network methods are widely known. The advantage of neural networks lies in the following theoretical aspects. First, neural networks are data driven self-adaptive methods. Second, they are universal functional approximators in that neural networks can approximate any function with arbitrary accuracy. Third, they are nonlinear models, which makes them flexible in modelling real world complex relationships [22].
- The decision tree methods, are widely used for some classification problems. The algorithms that are used for constructing these trees usually work top-down by choosing a variable at each step that is the (next) best variable to use in splitting the set of items. A tree can be trained by splitting the source set into subsets based on an attribute value test. This process is repeated on each derived subset in a recursive manner called recursive partitioning [7].

Having examined the above-mentioned methods, we have chosen a fuzzy rule-based classification to check the result of classification in troublesome cases as the most promising algorithm. The results we receive thanks to the adoption of this algorithm will support our pattern library with the intention of enabling the user to build their image query in as natural a way as possible. 'Natural' here means handling such objects as houses, trees, water instead of a red square, blue rectangle, etc.

In this paper we present a fuzzy rule-based classifier for object classification which takes into account object features, together with spatial location of segmented objects in the image. In order to improve the comparison of two images, we need to classify these objects in a semantic way. We present the concept of an image search engine which takes into account object feature vectors, together with spatial location of segmented objects in the image.

1.1. CBIR concept overview

In general, our system consists of four main blocks (see Figure 1):

1. the image preprocessing block (responsible for image segmentation), implemented in Matlab, (cf. [12]);
2. the database, which is implemented in Oracle Database (DB), stores information about whole images, their segments (here referred to as graphical objects), segment attributes, object location, pattern types and object identification, (cf. [11]);

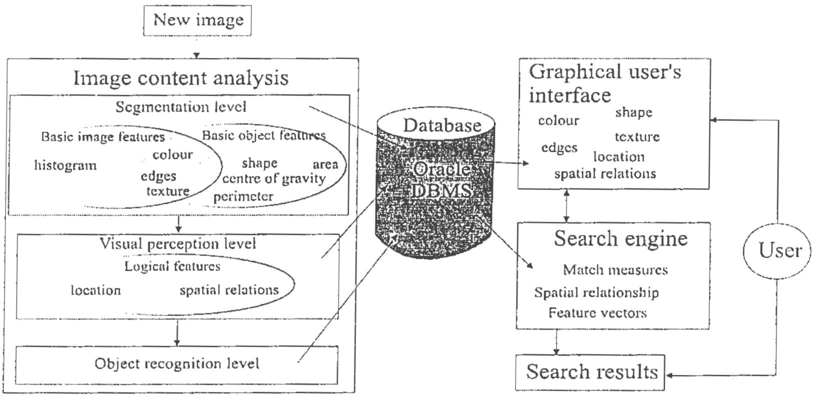

Figure 1. Block diagram of our content-based image retrieval system.

3. the search engine responsible for the searching procedure and retrieval process based on feature vectors of objects and spatial relationship of these objects in an image, implemented in Matlab;
4. the graphical user's interface (GUI) which allows users to compose their own image, consisting of separate graphical objects as a query. Classification helps in the transition from rough graphical objects to human semantic elements. We have had to create a user-friendly semantic system, also implemented in Matlab.

1.2. Representation of graphical data

In our system, a new image is segmented, yielding as a results a collection of objects. Both the image and the extracted objects are stored in the database. Each object, selected according to the algorithm presented in detail in [12], is described by some low-level features. The features describing each object include:

- average colour k_{av},
- texture parameters T_p,
- area A,
- convex area A_c,

- filled area A_f,
- centroid $\{x_c, y_c\}$,
- eccentricity e,
- orientation α,
- moments of inertia m_{11}, m_{12}, m_{21}, m_{22},
- bounding box $\{bb_1(x,y), ..., bb_4(x,y)\}$,
- major axis length m_{long},
- minor axis length m_{short},
- solidity s,
- Euler number E
- Zernike moments $Z_{00},\ldots,Z_{33}$.
- and some others.

Let Fo be a set of features where:

$$F_O = \{k_{av}, T_p, A, A_c, ..., E\} \tag{1}$$

Hence, for an object, we construct a feature vector

$$\mathbf{x} = [x_1, x_2, \ldots, x_n], \tag{2}$$

where n is the number of the above-mentioned features.

1.3. Classification problem in CBIR

The feature vector (2) is further used for object classification. Therefore, we propose to define a pattern for each class of objects at first in order to assign new images to a particular class. We define a representative feature vector, of the same length as all component feature vectors and name it a pattern P_k for each class. Patterns can be created in different ways. The simplest method is a calculation of the average value of each vector component.

We also assume weights $\mu_k(i)$ for all pattern features where: k is a number of classes, i is a number of feature, $1 \le i \le n$. Weights satisfy: $\mu_k(i) \in [0,1]$. These weights for each pattern feature should be assigned in terms of the best distinguishability of patterns and we assign them in a heuristic way. More sophisticated methods can also be used.

For all these data we create the pattern library (also stored in the DB) which contains information about pattern types and allowable parameter values for an object.

The above described procedure supports object classification which is crucial in the context of a CBIR and is used for several purposes, for example [21]:

1. to compare whole images. Specifically, an algorithm which describes a spatial object location needs classified objects.
2. to help the user form a query in GUI. The user forms a query choosing graphical objects semantically collected in groups.
3. to compare image objects coming from the same class as a stage in the image retrieval process. Details are presented in sec. 5.

2. Fuzzy classification

In spite of the existence of numerous classifiers, of which some were mentioned in sec. 1.1, in the case when ranges of feature values overlap the use of fuzzy classification seems to be justified.

According to Zadeh [22] a fuzzy set F in U is uniquely specified by its membership function $\mu_F: U \rightarrow [0,1]$. Thus, the fuzzy set is described as follows

$$F = \{(u, \mu_F(u)) | u \in U\} \tag{3}$$

For our purpose, we use a trapezoidal membership function μ_t which is defined by four parameters a, b, c, d:

$$\mu_t(u; a, b, c, d) = \begin{cases} 0, & u < a \\ (u-a)/(b-a), & a \le u \le b \\ 1, & b \le u \le c \\ (d-u)/(d-c), & c \le u \le d \\ 0, & d < u \end{cases} \tag{4}$$

Let F and G be two fuzzy sets in the universe U, we say that $F \subseteq G \Leftrightarrow \mu_F(u) \le \mu_G(u)$, $\forall\, u \in U$. The complement of F, denoted by F^c, is defined by $\mu_F^c(u) = 1 - \mu_F(u)$. Furthermore, the intersection $F \cap G$ and union $F \cup G$ are defined as $\mu_{F \cap G} = \min(\mu_F(u), \mu_G(u))$ and $\mu_{F \cup G} = \max(\mu_F(u), \mu_G(u))$, respectively.

2.1. Fuzzy rule-based classifiers

Let us consider an M-class classification problem in an n-dimensional normalized hypercube $[0, 1]^n$. For this problem, we use fuzzy rules of the following type [8]:

$$\text{Rule } R_q: \text{If } x_1 \text{ is } A_{q1} \text{ and ... and } x_n \text{ is } A_{qn} \text{ then Class } C_q \text{ with } CF_q, \tag{5}$$

where R_q is the label of the q^{th} fuzzy rule, $\mathbf{x} = (x_1, ..., x_n)$ is an n-dimensional feature vector (2), A_{qi} is an antecedent fuzzy set ($i = 1,...,n$), C_q is a class label, CF_q is a real number in the unit interval [0,1] which represents a rule weight. The rule weight can be specified by a heuristic manner or it can be adjusted, e.g. by a learning algorithm introduced by Ishibuchi et al. [17], [9]. We use the n-dimensional vector $\mathbf{A}_q = (A_{q1}, ..., A_{qn})$ to represent the antecedent part of the fuzzy rule R_q in (5) in a concise manner.

A set of fuzzy rules S of the type shown in (5) forms a fuzzy rule-based classifier. When an n-dimensional vector $\mathbf{x}_p = (x_{p1}, ..., x_{pn})$ is presented to S, first the *compatibility grade* of $\mathbf{x}_p$ with the antecedent part $\mathbf{A}_q$ of each fuzzy rule R_q in S is calculated by the product operator as

$$\mu_{\mathbf{A}_q}(\mathbf{x}_p) = \mu_{A_{q1}}(\mathbf{x}_{p1}) \times ... \times \mu_{A_{qn}}(\mathbf{x}_{pn}) \quad \text{for} \quad R_q \in S, \tag{6}$$

where $\mu_{A_{qi}}(.)$ is the membership function of A_{qi}. Then a single winner rule $R_{w(\mathbf{x}_p)}$ is identified for $\mathbf{x}_p$ as follows:

$$w(\mathbf{x}_p) = \arg\max_q \{CF_q \times \mu_{\mathbf{A}_q}(\mathbf{x}_p) \mid R_q \in S\}, \tag{7}$$

where $w(\mathbf{x}_p)$ denotes the rule index of the winner rule for $\mathbf{x}_p$.

The vector $\mathbf{x}_p$ is classified by the single winner rule $R_{w(\mathbf{x}_p)}$ belonging to the respective class. If there is no fuzzy rule with a positive *compatibility grade* of $\mathbf{x}_p$ (i.e., if $\mathbf{x}_p$ is not covered by any fuzzy rules in S), the classification of $\mathbf{x}_p$ is rejected. The classification of $\mathbf{x}_p$ is also rejected if multiple fuzzy rules with different consequent classes have the same maximum value on the right-hand side of (7). In this case, $\mathbf{x}_p$ is on the classification boundary between the different classes. We use the single winner-based fuzzy reasoning method in (7) for pattern classification.

An ideal theoretical example of a simple three-class, two-dimensional pattern classification problem with 20 patterns from each class is considered by Ishibuchi and Nojima [8] (Fig. 2 a)). There three linguistic values (*small*, *medium* and *large*) were used as antecedent fuzzy sets for each of the two attributes, and 3×3 fuzzy rules were generated. S_1 was the fuzzy rule-based classifier with the nine fuzzy rules shown below:

S_1: fuzzy rule-based classifier with nine fuzzy rules
R_1: If x_1 is *small* and x_2 is *small* then Class2 with 1.0,
R_2: If x_1 is *small* and x_2 is *medium* then Class2 with 1.0,
R_3: If x_1 is *small* and x_2 is *large* then Class1 with 1.0,
R_4: If x_1 is *medium* and x_2 is *small* then Class2 with 1.0,
R_5: If x_1 is *medium* and x_2 is *medium* then Class2 with 1.0,
R_6: If x_1 is *medium* and x_2 is *large* then Class1 with 1.0,
R_7: If x_1 is *large* and x_2 is *small* then Class3 with 1.0,
R_8: If x_1 is *large* and x_2 is *medium* then Class3 with 1.0,
R_9: If x_1 is *large* and x_2 is *large* then Class3 with 1.0.

For simplicity, the rule weight is 1.0 in S_1. The location of each rule is shown in Figure 2 b).

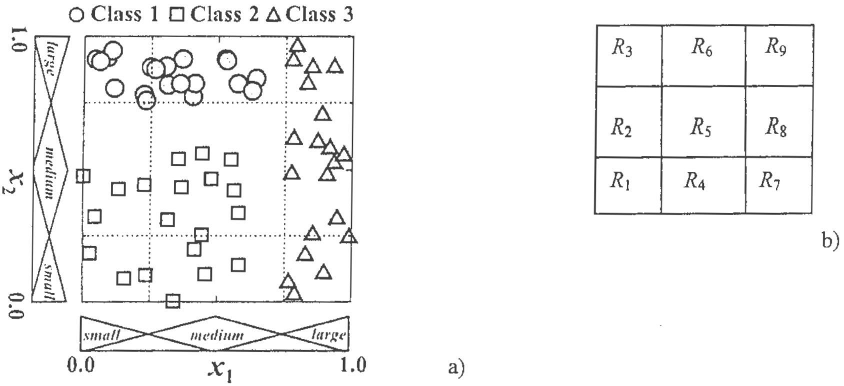

Figure 2. a) An ideal example of fuzzy rule-based classifier S_1 developed by Ishibuchi and Nojima [8]; b) Classification boundaries for fuzzy rule-based classifier S_1.

3. Classification results

Based on the data collected in our CBIR system (n = 32 features for each graphical object), we have analysed the most distinguished features to present our experimental results. As an example, we have chosen three classes from graphical objects in the training subset, namely: class1 - roof, class2 – window frame and class3 – window pane, presented respectively in Figure 3.

For our fuzzy rule-based classifier we have chosen a trapezoidal membership function (cf. (4)), as it is more convenient to represent the character of our data. We construct the fuzzy rule-based classifier S_{r1} based on data from a training subset. This classifier consists of nine fuzzy rules:

Figure 3. Examples of graphical objects used as class1 - roof a), class2 – window frame b) and class3 - window pane c) from the training subset.

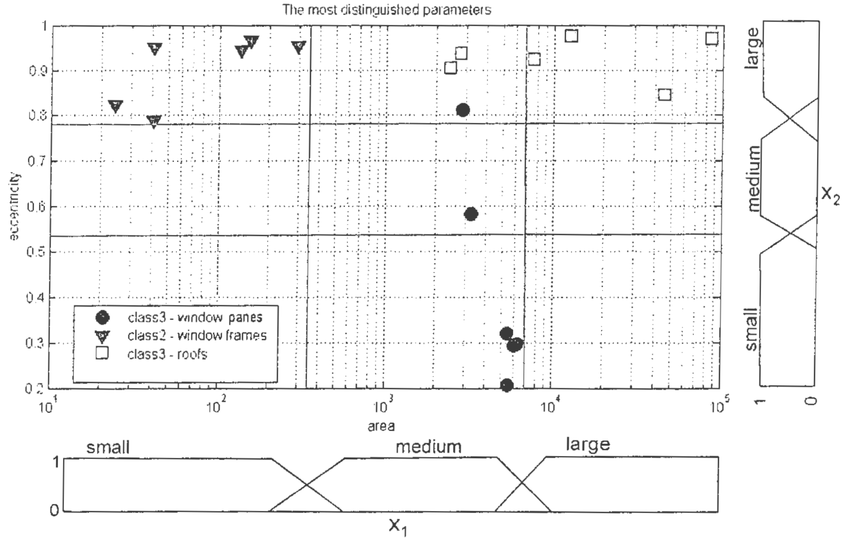

Figure 4. Three-class problem with two features: x_1 - area and x_2 – eccentricity characterising pattern.

S_{r1}: fuzzy rule-based classifier with nine fuzzy rules

R_1: If x_1 is *small* and x_2 is *small* then Class2 with 1.0,
R_2: If x_1 is *small* and x_2 is *medium* then Class2 with 1.0,
R_3: If x_1 is *small* and x_2 is *large* then Class2 with 1.0,
R_4: If x_1 is *medium* and x_2 is *small* then Class3 with 1.0,
R_5: If x_1 is *medium* and x_2 is *medium* then Class3 with 1.0,
R_6: If x_1 is *medium* and x_2 is *large* then Class3 with 1.0,
R_7: If x_1 is *large* and x_2 is *small* then Class1 with 1.0,
R_8: If x_1 is *large* and x_2 is *medium* then Class1 with 1.0,
R_9: If x_1 is *large* and x_2 is *large* then Class1 with 1.0.

As we mentioned earlier, in our experiment we used a three-class problem for two pairs of features: x_1 – area and x_2 – eccentricity (shown in Fig. 4), and x_1 – area and x_2 – solidity (shown in Fig. 5). These above-mentioned features describe the same classes of objects. For the latter pair we construct classifier S_{r3} analogically to S_{r1}.

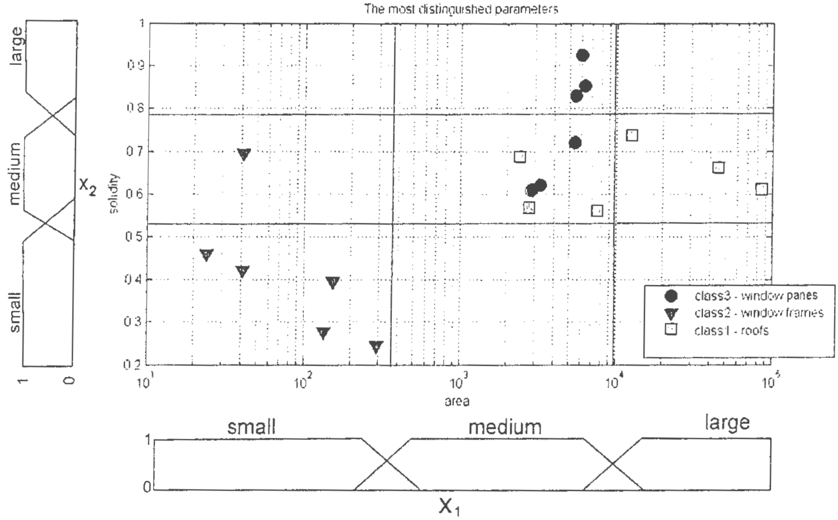

Figure 5. Three-class problem with two features: x_1 - area and x_2 – solidity characterising pattern.

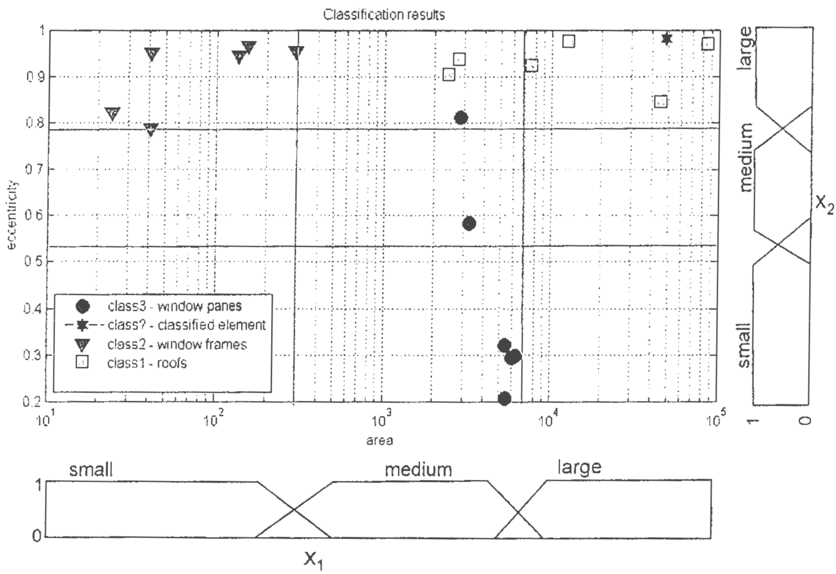

Figure 6. The black asterisk is a classified element for the fuzzy rule classifier S_{r1} with nine rules.

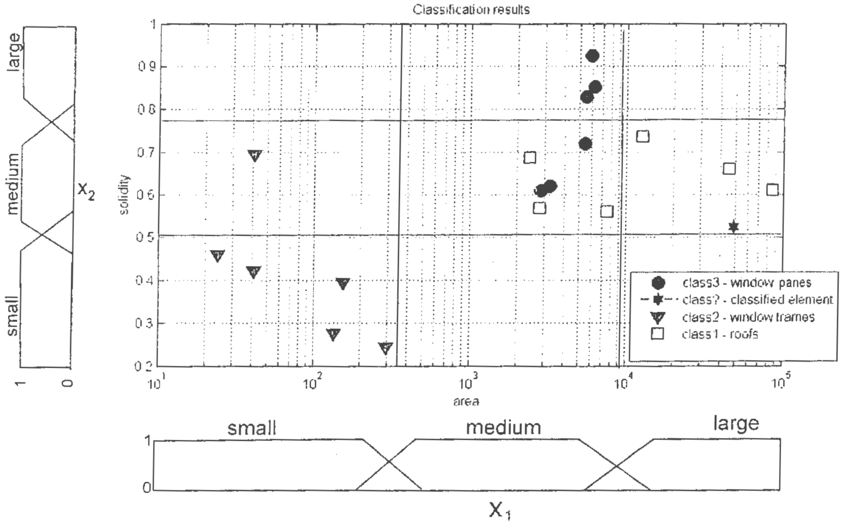

Figure 7. The black asterisk is a classified element for the fuzzy rule classifier S_{r3} with nine rules.

According to the fuzzy rule-based classifiers S_{r1} and S_{r3} we can classify a new object depicted as a black asterisk from unknown "class?" in Fig. 6 and 7 to class1, respectively. After a comparison with the real image object, we can conclude that the classified object, in fact, belongs to class1. This confirms that we can use the single winner-based fuzzy reasoning method for our pattern classification (see Fig. 6 and 7).

Let us try to simplify the fuzzy rule-based classifier S_1. We can see from Fig.2 a) that all training patterns with large values of x_1 are from class3. Accordingly, the last three fuzzy rules R_7, R_8 and R_9 in S_1 with the same antecedent condition "x_1 is large" have the same consequent class (i.e. class3). This observation suggests that we could combine that last three rules R_7, R_8 and R_9 of S_1 into a single fuzzy rule:

R_{789}: If x_1 is *large* then Class3 with 1.0.

Hence, if the negation of *large* (i.e., *not large*) is used in the antecedent part, the number of fuzzy rules in S_1 can be described as [8]:

S_2: fuzzy rule-based classifier with three fuzzy rules
R_{1245}: If x_1 is *not large* and x_2 is *not large* then Class2 with 1.0,
R_{36}: If x_1 is *not large* and x_2 is *large* then Class1 with 1.0,
R_{789}: If x_1 is *large* then Class3 with 1.0.

where the membership function of *not large* is defined as:

$$\mu_{not\ large}(x) = 1 - \mu_{large}(x) \quad \text{for} \quad 0 \le x \le 1 \tag{8}$$

Now, we show the use of a fuzzy rule-based classifier with three rules for our three-class problem, but, in our case, we use the negation of *small* (i.e., *not small*) in the antecedent part, where:

S_{r2}: fuzzy rule-based classifier with three fuzzy rules
R_{23}: If x_1 is *small* and x_2 is *not small* then Class3 with 1.0,
R_{5689}: If x_1 is *not small* and x_2 is *not small* then Class1 with 1.0,
R_{147}: If x_2 is *small* then Class2 with 1.0.

where the membership function of *not small* is defined as:

$$\mu_{not\,small}(x) = 1 - \mu_{small}(x) \quad \text{for} \quad 0 \le x \le 1 \tag{9}$$

For this purpose we have chosen the fuzzy rule-based classifier S_{r2}. As we have mentioned earlier, in our second experiment we used a three-class problem with two features: x_1 – minor axis length and x_2 – blue component of RGB colour. We use the same classes (class1 - roof, class2 – window frame and class3 – window pane). As it is shown in Fig. 8, the three rules are enough to separate the objects in our real data.

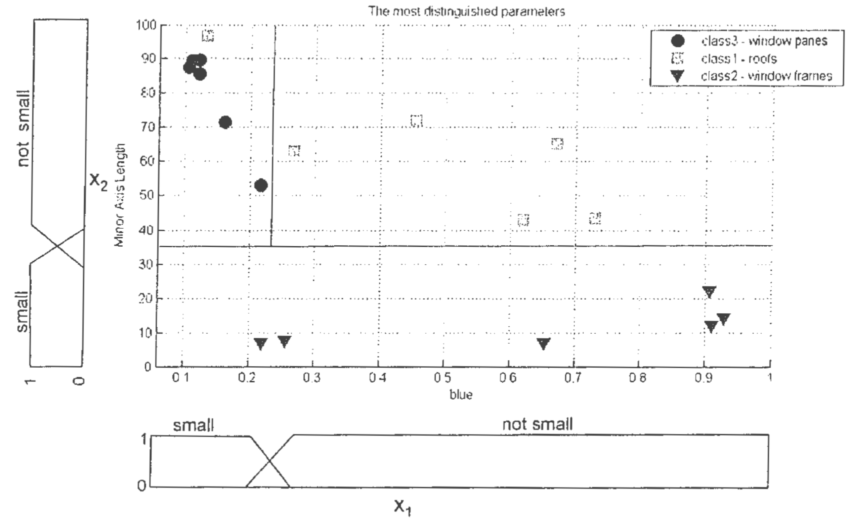

Figure 8. Classification with three fuzzy rules S_{r2}; x_1 – minor axis length and x_2 – blue.

According to the fuzzy rule-based classifier S_{r2}, we can classify a new object depicted as a magenta asterisk from unknown "class?" in Fig. 9 to class1. After a comparison with the real image object, we can conclude that the classified object in fact belongs to class1. This confirms that we can use a single winner-based fuzzy reasoning method for our pattern classification (see Fig. 9).

4. Use of classified objects in CBIR

Therefore, we have to classify objects, op. cit. Jaworska [10], in order to:

1. use particular patterns as classes. We store these data in DB to use them in CBIR algorithms;
2. specify a spatial object location to compare whole images in our system.

3. help the user ask a query in GUI. The user chooses for a query graphical objects semantically collected in groups;
4. compare image objects coming from the same class as a stage in the image retrieval process.

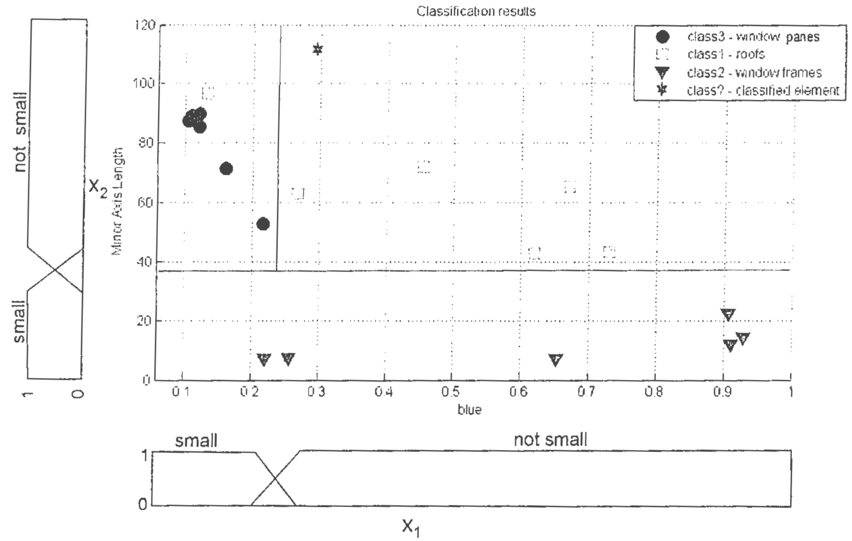

Figure 9. Classification results for a classifier with three fuzzy rules S_{r2}, where the magenta asterisk is a classified element.

5. Construction of the search engine

Now, we will describe how the similarity between two images is determined and used to answer a query. For the comparison of the spatial features of two images, an image I_i is interpreted as a set of n objects composing it:

$$I_q = \{o_{i1}, o_{i2}, \ldots, o_{in}\} \tag{10}$$

Let a query be an image I_q, such as $I_q = \{o_{q1}, o_{q2}, \ldots, o_{qn}\}$. An image in the database will be denoted as I_b, $I_b = \{o_{b1}, o_{b2}, \ldots, o_{bm}\}$. In order to answer the query, represented by I_q, we compare it with each image I_b in the database in the following way.

Let us assume that there are, in total, M classes of the objects recognized in the database, denoted as labels $L_1, L_2, \ldots, L_M$. Then, by the signature of an image I_i (cf. (10)) we mean the following vector:

$$\text{Signature}(I_i) = [\text{nobc}_{i1}, \text{nobc}_{i2}, \ldots, \text{nobc}_{iM}] \tag{11}$$

where: nobc_{ik} denotes the number of objects of class L_k present in the representation of an image I_i. First of all, we determine a similarity measure sim_{sgn} between query I_q and image I_b:

$$\mathrm{sim}_{\mathrm{sgn}}(I_q, I_b) = d(\mathrm{sgn}\,(I_q), \mathrm{sgn}(I_b)) \tag{12}$$

computing the distance between two vectors of their signatures.

If the similarity (12) is smaller than a threshold (a parameter of the query), then image I_b is rejected, i.e., not considered further in the process of answering query I_q. Otherwise, we proceed to the next step and we find the spatial similarity $\mathrm{sim}_{\mathrm{PCV}}$ of images I_q and I_b computing the Euclidean distance between their PCVs as:

$$\mathrm{sim}_{\mathrm{PCV}}(I_q, I_b) = 1 - \sqrt{\sum_{i=1}^{3} (PCV_{bi} - PCV_{qi})^2} \tag{13}$$

If the similarity (13) is smaller than the threshold (a parameter of the query), then image I_b is rejected, i.e., not considered further in the process of answering query I_q. Otherwise, we proceed to the final step, namely, we compare the similarity of the objects representing both images I_q and I_b. For each object o_{qi} present in the representation of the query I_q, we find the most similar object o_{bj} of the same class, i.e., $L_{qi} = L_{bj}$. If there is no object o_{bj} of the class L_{qi}, then $\mathrm{sim}_{\mathrm{ob}}\,(o_{qi}, o_b)$ is equal to 0. Otherwise, similarity $\mathrm{sim}_{\mathrm{ob}}\,(o_{qi}, o_b)$ between objects of the same class is computed as follows:

$$\mathrm{sim}_{\mathrm{ob}}(o_{qi}, o_{bj}) = 1 - \sqrt{\sum_{l} (Fo_{qil} - Fo_{bjl})^2} \tag{14}$$

where l indexes the set of features F_O used to represent an object, as described in (1). When we find highly similar objects (for instance, $\mathrm{sim}_{\mathrm{ob}} > 0.9$), we eliminate these two objects from the process of comparison described by Mucha and Sankowski [15]. This process is realized according to the Hungarian algorithm for the assignment problem implemented by Munkres. Thus, we obtain the vector of similarities between query I_q and image I_b.

$$\mathrm{sim}(I_q, I_b) = \begin{bmatrix} \mathrm{sim}_{\mathrm{ob}}(o_{q1}, o_{b1}) \\ \vdots \\ \mathrm{sim}_{\mathrm{ob}}(o_{qn}, o_{bn}) \end{bmatrix} \tag{15}$$

where n is the number of objects present in the representation of I_q. In order to compare images I_b with the query I_q, we compute the sum of $\mathrm{sim}_{\mathrm{ob}}\,(o_{qi}, o_{bi})$ and then use the natural order of the numbers. Thus, the image I_b is listed as the first in the answer to the query I_q, for which the sum of similarities is the highest.

6. Conclusions

In this paper, first we have determined the ability of fuzzy sets and fuzzy rule-based classifiers to classify graphical objects in our CBIR system. We have shown an example of classification based on nine and three fuzzy rules according to the data character. We have chosen the most distinguished coordinates from a feature vector in order to exemplify the proposed method that seems to be quite promising.

Intensive computational experiments are under way in order to draw some conclusions regarding the choice of parameters for the model. We are also verifying object classification and identification procedures that have been established. The GUI prototype which has been constructed is being put to test. However, the preliminary results we have obtained so far, using the simplest configuration, are quite hopeful.

As for the prospects for future work, the implementation of an optimised procedure should prove the feasibility of the approach. We expect a reasonable performance from the evaluation strategy outlined in the paper.

References

[1] Ali J.M. *Content-Based Image Classification and Retrieval: A Rule-Based System Using Rough Sets Framework.* Chap. IV, 68-82, Artificial Intelligence for Maximizing Content Based Image Retrieval, Zongmin Ma (ed.), New York, 2009.

[2] Berzal F., Cubero J.C., Kacprzyk J., Marin N., Vila M.A. and Zadrożny S. *A General Framework for Computing with Words in Object-Oriented Programming.* In: Bouchon-Meunier B. (ed.), International Journal of Uncertainty. Fuzziness and Knowledge-Based Systems, Vol. 15, (Suppl.), Feb., 111-131, World Scientific Publishing Company, Singapore, 2007.

[3] Candan K.S. and Li W-S. *On Similarity Measures for Multimedia Database Applications.* Knowledge and Information Systems, Vol. 3, 2001, 30-51.

[4] Cubero J.C., Marin N., Medina J.M., Pons O. and Vila, M.A. *Fuzzy Object Management in an Object-Relational Framework,* Proceedings of the 10th International Conference IPMU, Perugia, Italy, 2004, 1775-1782.

[5] Dasarathy B.V, (ed.) *Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques*, IEEE Computer Society Press, 1991.

[6] Deb S. (ed.) *Multimedia Systems and Content-Based Image Retrieval*, Chap. VII and XI, IDEA Group Publishing, Melbourne, 2004.

[7] Fei, B. and Liu, J. *Binary Tree of SVM: A New Fast Multiclass Training and Classification Algorithm.* IEEE Transaction on Neural Networks, Vol. 17(3), 2006, 696–704

[8] Ishibuchi H. and Nojima Y. *Toward Quantitative Definition of Explanation Ability of fuzzy rule-based classifiers*, IEEE International Conference on Fuzzy Systems, June 27-39, 2011,Taipai, Taiwan, 2011, 549-556.

[9] Ishibuchi H. and Yamamoto T. *Rule weight specification in fuzzy rule-based classification systems*, IEEE Transactions on Fuzzy Systems, Vol. 13(4), 2005, 428-435.

[10] Jaworska T. *A Search Engine Concept Based on Multi-feature Vectors and Spatial Relationship,* LNAI Vol. 7022, Springer, 2011, 137-148.

[11] Jaworska T. *Database as a Crucial Element for CBIR Systems.* Proceedings of the 2nd International Symposium on Test Automation and Instrumentation, Vol. 4, World Publishing Corporation, Beijing, China, 2008, 1983-1986.

[12] Jaworska T. *Object extraction as a basic process for content-based image retrieval (CBIR) system.* Opto-Electronics Review, Association of Polish Electrical Engineers (SEP), Vol. 15(4), Warsaw, 2007, 184-195.

[13] Lee J., Kuo J-Y. and Xue N-L. *A note on current approaches to extending fuzzy logic to object oriented modelling.* International Journal of Intelligent Systems, Vol. 16(7), 2001, 807-820.

[14] Liu Y., Zhang D., Lu G. and Ma W-Y. *A survey of content-based image retrieval with high-level semantics.* Pattern Recognition, Vol. 40, Elsevier, 2007, 262-282.

[15] Mucha M. and Sankowski P. *Maximum Matching via Gaussian Elimination,* Proceedings of the 45th Annual Symposium on Foundations of Computer Science (FOCS'04), 2004, 248-255.

[16] Niblack W., Berber R., Equitz W., Flickner M., Glasman E., Petkovic D. and Yanker P. *The QBIC Project: Querying Images by Content Using Colour, Texture and Shape*, SPIE, Vol. 1908, 1993, 173–187.

[17] Nozaki K., Ishibuchi H. and Tanaka H. *Adaptive fuzzy rule-based classification systems*, IEEE Transactions on Fuzzy Systems, Vol. 13(4), 1996, 238-250.

[18] Ogle V. and Stonebraker M. *CHABOT: Retrieval from a Relational Database of Images*. IEEE Computer, Vol. 28(9), 1995, 40-48.
[19] Pons O., Vila M.A. and Kacprzyk J. *Knowledge management in fuzzy databases,* Studies in Fuzziness and Soft Computing, Vol. 39, Physica–Verlag, Heidelberg and New York, 2000.
[20] Rish I. *An empirical study of the naive Bayes classifier*, Proceedings of IJCAI-2001 workshop on Empirical Methods in AI, 2001, 41-46.
[21] Zadeh L.A. *Fuzzy sets*, Information and Control, Vol. 8(3), 1965, 338-353.
[22] Zhang G.P. *Neural Networks for Classification: A Survey*, IEEE Transactions on Systems, Man and Cybernetics, part C: Applications and reviews, Vol. 30(4), 2000, 451-462.