

# A Hybrid Approach for Image Retrieval with Ontological Content-Based Indexing

Oleg Starostenko, Alberto Chávez-Aragón, J. Alfredo Sánchez,  
and Yulia Ostróvskaya

Universidad de las Américas, Puebla,  
Computer Science Department, Cholula, Puebla, 72820, Mexico  
{oldwall, sp098974, alfredo, yulia}@mail.udlap.mx

**Abstract.** This paper presents a novel approach for image retrieval from digital collections. Specifically, we describe IRONS (Image Retrieval with Ontological Descriptions of Shapes), a system based on the application of several novel algorithms that combine low-level image analysis techniques with automatic shape extraction and indexing. In order to speed up preprocessing, we have proposed and implemented the convex regions algorithm and discrete curve evolution approach. The image indexing module of IRONS is addressed using two proposed algorithms: the tangent space and the two-segment turning function for shapes representation invariant to rotation and scale. Another goal of the proposed method is the integration of user-oriented descriptions, which leads to more complete retrieval by accelerating the convergence to the expected result. For the definition of image semantics, ontology annotation of sub-regions has been used.

## 1 Introduction

A typical approach to automatic indexing and classification of images is based on the analysis of low-level image characteristics, such as color, texture or shape [1], [2], but this type of systems does not provide the semantics associated with the content of each image. There are a number of well-known systems for visual information retrieval (VIR). The Query by Image Content system (QBIC) provides retrieval of images, graphics and video data from online collections using image features such as color, texture, and shape for computing the similarity between images [3]. AMORE (Advanced Multimedia Oriented Retrieval Engine) and SQUID systems provide image retrieval from the Web using queries formed by keywords specifying similar images, sketches, and SQL predicates [4]. Whereas the contributions of these systems have been important in the field, they do not provide ways to represent the meaning of objects in the images. In order to overcome this problem, our hypothesis is to apply the machine-understandable semantics for search, access, and retrieval of multimedia information using ontology [5]. The widely used Grubber's definition permits to describe semantics, establishes a common and shared understanding of a domain and facilitates the implementation of user-oriented vocabulary of terms and their relationship with objects in image the [6]. The potential applications of the proposed image retrieval facilities include systems for supporting digital image processing services,

high performance exchange of multimedia data in distributed collaborative and learning environments, digital libraries, etc.

## 2 Proposed Image Retrieval Method

The proposed method may be described as a combination of specific descriptors based on low-level image preprocessing for extraction of sub-regions (objects) invariant to scale, rotation, and illumination, and the application of ontology concepts for definition of machine-understandable semantics for retrieved images. The main procedures for image preprocessing, indexing, ontological description and retrieval are:

1. In order to divide the image into regions, the SUSAN corner detection method is used by applying the circular mask [7], [8]. The extracted principal corners present the points that define particular positions of objects in the image.

2. The spatial sampling of the original image is provided by computing the average values of color regions via the *111213* color model, applying slicing 8x8 pixels windows generating the main color descriptor of each region [9].

3. Comparing the proposed method with well-known prototypes, where the description is applied to the whole image, the textual annotations of sub-regions are preferred for the simple definition of their semantic characteristics. Subdivision of image into sub-regions is provided by Convex Regions Preprocessing Algorithm in Images (CORPAI) proposed by the authors [10]. Detected principal corners are used for convex hulls generation providing the vertical slabs algorithm and producing a sorted array that is used to determine the sub-region as a polygon.

4. The frequent problem of shape representation is a great number of necessary vertices for polygon transformation, which may be reduced by the proposed discrete curve evolution process. This technique reduces the set of vertices of a polygon to a subset of vertices containing relevant information about the original outline [10].

5. The next step is indexing of the simplified object (shape, polygon); here two different approaches have been proposed and implemented. One of them is based on an object transform to a tangent space, and the other represents the object as a two-segment turning function.

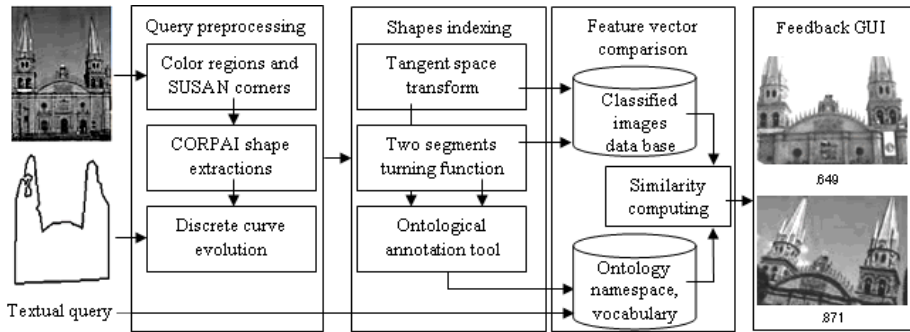
6. Finally, it is possible to establish the relationship between the object and its formal explicit definition. In such a way, the meaning of an image may be obtained in textual form as a set of annotations for each sub-region related to a particular ontology. The Resource Description Framework (RDF) language to support the ontology management is used in this approach that defines a syntactic convention and a simple data model to implement machine-readable semantics [11]. Using RDF it is possible to describe each web resource with relations to its object-attributes-value based on metadata standard developed by the World Wide Web Consortium [12].

## 3 Irons Image Retrieval System

The block diagram of the Image Retrieval by Ontological Description of Shapes (IRONS) system implementing the proposed method is shown in Fig. 1. The input for the system may be an image, its shape, or a keyword, which describes the object with

a certain degree of similarity. The retrieved images will be the ones with more similarity to the low-level features of the query (if the input is an image or its sub-region) and will have a high degree of matching with the ontology annotations defining the content of the image. Once the user draws a query, the system uses the shape indexing algorithm in order to generate the feature vector for comparison with the other ones in the image database [10]. Then the content-based recognition process is applied to shapes (based on the ontological textual description) in order to find similar ones in the ontology namespace.

The IRONS system consists of four principal modules: query preprocessing, indexing module, feature vector comparison and feedback GUI, and it operates according to the algorithm described in section 2. The query preprocessing module provides the selection of sub-regions containing relevant objects. Once the sub-region is extracted, the object within that sub-region is found by the CORPAI algorithm.



**Fig. 1.** Block diagram of the proposed IRONS system

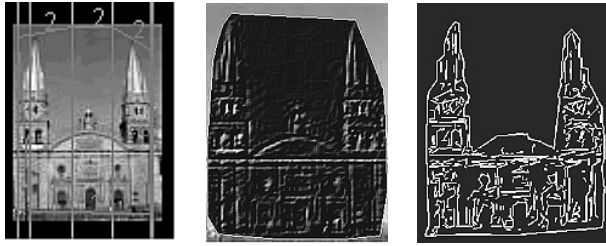
The discrete curve evolution process reduces the complexity of the extracted shape. If the query is a keyword, the preprocessing step is not applied. The indexing module generates a feature vector describing low-level image characteristics and content-based annotations. The preprocessed polygon is represented either by tangent space or by two-segment turning function because these techniques are invariant to scaling, rotation, and translation. The ontological annotation tool is used for searching matches in the ontology name space. The images with higher matching are retrieved and visualized on GUI with a certain degree of similarity.

### 3.1 Query Preprocessing Module

The algorithm for image preprocessing using color and principal corners is described.  
*Input:* A color image with luminance of pixels  $I_c$ ; *Output:* the region's feature vector

1.  $I_g \Leftarrow \text{ComputeLuminance}(\text{using } I_c)$  // it converts color into gray level image
2.  $\text{Principal corners} \Leftarrow \text{SUSAN operator}(I_g)$  // detection of object's corners
3.  $\text{Scs} \Leftarrow \text{SpatialSampling}(I_c)$  // reduction of image size to an 8x8 pixels window

4. *ColorDescriptor*  $\leftarrow$  *ComputeColorDescriptorIII2I3* (*Scs*) // descriptors based on III2I3 color system model
5. *FeaturesVector*  $\leftarrow$  *ComputeDescriptor* (*Principal Corners*, *ColorDescriptor*) // the sub-region descriptor includes a color vector and the principal corner's position.
6. *Subregion*  $\leftarrow$  *CORPAI*(*Ic*, *Sp*) // applying the CORPAI algorithm over regions  
*ConvexHulls*(*points*[]) // compute the convex hull  
{ if( *query\_sub-region*(*image*[[[]])) // apply boundary detection operator to sub-region (*operator*(*image*[[[]])) }
7. *Ic<sub>NEW</sub>*  $\leftarrow$  *TransformationFromSubregionToImage* (*Subregion*) // transformation of the irregular convex sub-region of the original image to a new normalized one
8. *FeaturesVector*  $\leftarrow$  *ComputeDescriptor* (*Principal Corners*, *ColorDescriptor*, *ConvexRegions*) // the convex region descriptor is obtained.
9. *FeaturesVector*  $\leftarrow$  *DiscreteCurveEvolution* (*Simplified Polygon*) // removal of the least important polygon vertexes.



**Fig. 2.** Results of applying the vertical slabs algorithm for extraction of a convex sub-region and the containing object

The last procedure of the preprocessing step is simplification of polygons representing the shape of objects detected by discrete curve evolution. This process removes the least important vertexes of a polygon, computing the relevance measure  $K$ , where  $\beta(s_1, s_2)$  is the turn angle of the common vertex of segments  $s_1, s_2$ , and  $l$  is the length function normalized with respect to the total length of the polygonal curve  $C$ .

$$K(s_1, s_2) = \frac{\beta(s_1, s_2)l(s_1)l(s_2)}{l(s_1) + l(s_2)} \quad (1)$$

The lower value of  $K(s_1, s_2)$  corresponds to the least contribution of this curve to a shape. This process of vertexes removal is repeated until we obtain the desired shape simplification using the designed interface presented in Fig. 3.

### 3.2 Indexing Module

The polygonal representation is not a convenient form for calculating similarity between two shapes, an alternative representation such as the Tangent Space Representation (TSR) is proposed for generation of the feature vector and quantitative comparison of simple shapes. Using TSR, a curve  $C$  is converted to a step function: the steps on the  $x$ -axis represent the arc length of each segment in  $C$ , and the  $y$ -axis

represents the turn angle between two consecutive segments in  $C$ . In Fig. 3 indexing module GUI of the TSR is shown where the results of applying the discrete curve evolution and the TSR for selected complexity of the shape are depicted.

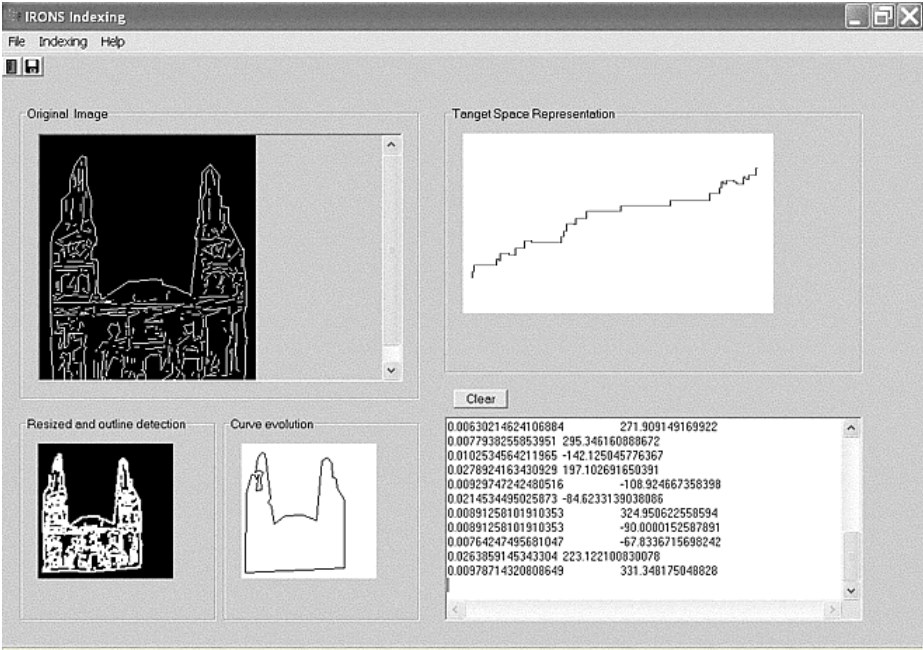


Fig. 3. Tangent space representation of shapes of the IRONS indexing module

Another possible way for shape indexing is to apply the cumulative angle function or turning function, which may speed up the computing the similarity between two shapes. In general, the turning function of a polygon  $A$  returns the angle between the counterclockwise tangent and the  $x$ -axis as a function of the arc length  $s$   $\Theta_A(s)$ . This function is invariant to translation and scale of a polygon, but it is not invariant to rotation. If a polygon is rotated by an angle  $\theta$ , the turning function value changes by the amount of  $\theta$ . To overcome this problem, we additionally propose an alternative way called two-segment turning function (2STF). With each iteration, the angle between two consecutive edges of a polygon is calculated. As a result, we may analyze the edges with rotation. Now we have the same representation of a shape even though that shape has been rotated. 2STF is calculated by traversing a digital curve in the counterclockwise direction and assigning a certain value of 2STF to each line segment. The  $x$ -value (the assigned value) defines the length of the line segment normalized with respect to the length of the complete curve. The  $y$ -value is the directional angle of the line segment with respect to its previous segment.

Using the GUI of the IRONS indexing module (Fig. 3), the 2STF may be computed and visualized in the same way as TSR. Once, the efficient way to represent a

shape is obtained via TSR or 2STF, the matching strategy to find the degree of similarity between two shapes is applied. Shape representation and matching are considered the most difficult aspects of content-based image retrieval [10]. In this work we use hybrid feature vector which defines such low-level image characteristics as semantic descriptions. This permits to speed up the matching process as well as reduce the number of iterations with non-sensical results. The similarity value between two shapes is based on proposed algorithm:

1. The polygon simplified by curve evolution is transposed into TSR or 2STF.
2. The resulting curves representing the polygon are scaled to the same length.
3. One of the curves is shifted vertically over the second one for a better fit.
4. The area between the two curves is computed.

Now the user may define a threshold value for the computed area as the acceptable degree of similarity between the reference and the analyzed patterns.

### 3.3 Ontological Annotation Tool

The ontology is described by a directed acyclic graph; each node has a feature vector that represents the concept associated with that node. Concept inclusion is represented by the IS-A inter-relationship. For instance, particular elements of buildings, churches, and so on form specific concepts of shapes defining these buildings, churches, etc. If the query describes an object using this ontology, the system would recover shapes that contain windows, columns, façades, etc. even though, those images have not been labeled as geometric figures for the retrieved object. The feature vectors of each node in the ontology name space consist of keywords linking the previously classified images to the characteristics of the new shape extracted by the TSR or 2STF. The indexing and the ontology annotation processes may be described now as:

1. *FeaturesVector*  $\leftarrow$  *Shape*(*Pentagon*,  $P_i, C_i$ ) //  $P_i$  is its TSR or 2STF representation and  $C_i$  is the compactness of the shape computed as a ratio: square of *Region-BorderLength* and *ShapeArea*.
2. *SaveRelationInOntology*( $I_c$ , *FeaturesVector* of  $I_{c_{NEW}}$ ,  $T_d$ ) // update the ontology namespace.

As has been mentioned, two kinds of vector comparison are used: matching the low-level image features and definition of similarity in ontological annotations. The computing of similarity is additionally provided by computing the Euclidean distance  $d_E$  to compare feature vectors according to the equation:

$$d_E(\mu, \sigma) = \sqrt{\sum (\mu - \sigma)^2}, \quad (2)$$

where  $\mu$  and  $\sigma$  denote two feature vectors.

The query interface of the IRONS system is shown in Fig. 4 where the images with high degree of matching are shown in downward order. The user may submit a visual example, a sketch, a keyword or a combination of the above.

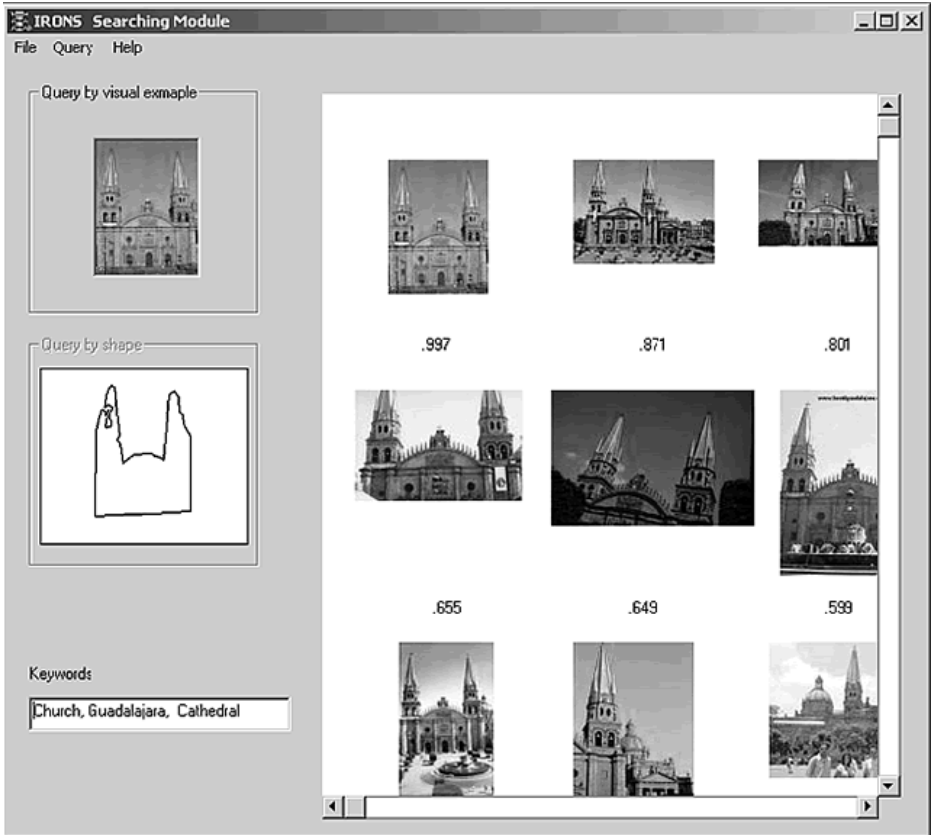


Fig. 4. Image retrieval GUI of the IRONS system

## 4 Evaluation, Contribution, and Conclusions

Evaluations of the proposed method and testing of the implemented system have been done comparing the results of image retrieval by the IRONS to several well-known systems, particularly, QBIC and AMORE systems. We performed a number of experiments to verify the role of the shape-based and ontology-based indexing in the retrieval process. We test the proposed method using the image collection CE-Shape-1. This database contains approximately 1400 images randomly taken from Internet and divided into 60 different categories with about 20 images per category.

The system performance is better when the image is processed in sub-regions; excessive subdivision does not produce good results. Satisfactory retrieval of expected images is achieved faster through the use of ontological descriptions due to the lower number of iterations in the search process. The analysis of the indexing approaches shows that 2STF is twice as fast as TSR. This occurs because the typical data structures used in indexing tools are hashing tables, which are manipulated with specific keys or signatures representing a shape. The disadvantages of the system are errors in

spatial sampling during generation of the image feature vector as well as the required amount of system memory. Factors like tolerance to occlusion and deformation, robustness against noise, and feasibility of indexing are also considered in our approach.

The most important contribution of this research is the proposed hybrid method combining the advantages of low-level image characteristics extraction with textual description of image semantics. The use of ontological annotations allows simple and fast estimation of the meaning of a sub-region and of the whole image. The proposed image retrieval method is robust to partial occlusion and to small changes in the position of the objects. From the obtained experimental results, we can conclude that the method could be considered as an alternative way for the development of visual information retrieval facilities.

## References

1. T. Gevers, A.W. Smeulders.: PicToSeek: Combining color and shape invariant features for image retrieval, *IEEE Trans. on Image Processing*, Vol. 9(1) (2000) 102-119.
2. O. Starostenko, J. Chávez.: Motion estimation algorithms of image processing services for wide community, *Proc. of Knowledge/Based Intelligent Information Engineering Systems Conference KES'2001, Japan* (2001) 758-763.
3. QBIC (TM). IBM's Query by image content, <http://www.qbic.almaden.ibm.com/>.
4. The Amore. Advance multimedia oriented retrieval engine, <http://www.ariadne.ac.uk/issue9/web-focus/>
5. D. Fensel.: *Ontologies: a silver bullet for knowledge management and electronic commerce*, USA: Springer (2001).
6. T.R. Gruber, A translation approach to portable ontology specifications, *Knowledge Acquisition* (1993) 199-220.
7. S.M. Smith, J.M. Brady.: A new approach to low-level image processing. *Journal of Computer Vision*, 23(1), 1997, 45-78.
8. O. Starostenko, J. Neme.: Novel advanced complex pattern recognition and motion characteristics estimation algorithms, *Proc. VI Iber - American Symposium on Pattern recognition, Brazil* (2001) 7-13.
9. M.S. Lew.: *Principles of visual information retrieval*, *Advances in pattern recognition USA: Springer-Verlag* (2001).
10. J.A. Chávez-Aragón, O. Starostenko, M. Medina.: Convex regions preprocessing algorithm in images, *Proc. of III International Symposium in Intelligent Technologies, Mexico* (2002) 41-45.
11. D. Fensel.: The semantic web and its languages, *IEEE Computer Society Vol. 15(6)* (2000) 67-73.
12. D. Beckett.: The design and implementation of the redland RDF application framework, *Proc. of the 10th International World Wide Web Conference, WWW* (2001) 120-125.