Topological Query in Image Databases

Mihaela Scuturici, Jérémy Clech, and Djamel A. Zighed

Laboratoire ERIC – Université Lumière Lyon2
5, avenue Pierre Mendès-France – 69676 Bron cedex – France
{mscuturi, jclech}@eric.univ-lyon2.fr; zighed@univ-lyon2.fr

Abstract. In this paper we propose a topological[1] model for image database query using neighborhood graphs. A related neighborhood graph is built from automatically extracted low-level features, which represent images as points of $\mathbb{R}^p$ space. Graph exploration correspond to database browsing, the neighbors of a node represent similar images. In order to perform query by example, we define a topological query model. The query image is inserted in the graph by locally updating the neighborhood graph. The topology of an image database is more informative than a similarity measure usually applied in content based image retrieval, as proved by our experiments.

1 Introduction

The information retrieval in image databases is still a challenge due to the fact that frequently the users seek semantically similar images while an image database provides similarity only at low level, by using characteristics computed from pixel values. Visual information retrieval implies the use of an index. There are two approaches to image indexing [9]: visual content based and annotation based. Visual content indexing supposes that the visual information of each image (given by pixel values) is resumed to a feature vector containing low-level features (color histogram, textural features, form features). Consequently, the query process is reduced to neighbors research inside the representation space [1]. A similarity measure is defined to identify the neighborhood. In this context, the query will start with a sample image. Annotation based indexing suppose that each image is annotated by using a keyword, a label, more generically a text. Each image is described by keywords, expressing the image semantic. The user searching an image having a certain semantic can express his request as a list of keywords. A similarity measure can also be useful to identify a set of images expressing the query semantic.

In this paper we are interested in the concept of "neighbor – neighborhood" of an image in an image database. Most of the search algorithms in image databases pro-

[1] The word "topology" denotes here the relationships between elements linked together in a system. It is not used neither in terms of mathematical study of the geometric properties of figures that are independent of size or shape and are preserved through deformations, twisting and stretching, nor in terms of family of subsets (family of all open subsets of a mathematical set, including the set itself and the empty set, which is closed under set union and finite intersection) [12].

A. Sanfeliu and J. Ruiz-Shulcloper (Eds.): CIARP 2003, LNCS 2905, pp. 145–152, 2003.

pose to seek the k nearest neighbors (kNN) [7] of an image by using a similarity measure [12]. For instance, the QBIC system [3], in its implementation for the Hermitage Museum[2] always returns 12 nearest neighbors of the sketch presented by the user as query in the color search or layer search engine. In some situations, as illustrated in the section 2, the kNN algorithm produces surprising results compared to user expectations. In section 3, we introduce a more appropriate neighborhood representation model: the topological neighborhood. Section 4 describes the topological query model. We will discuss its advantages and its limitations. In section 5 we present experimental results on an image database where the two query methods (kNN and topological neighborhood) are compared on the basis of recall and precision indicators. Concluding remarks and future works are presented in section 6.

2 Anomalies and Assumptions of the kNN

In the context of navigation in an image database, the system is typically driven by the user. Then, it would be more convenient for him that the system follows the human cognitive model. Since the user expects to see together sets of similar images, the system must guarantee a stability of those sets and we assume that the symmetry is one of the required conditions. Unfortunately, in some situations the kNN model does not verify this condition. We will illustrate this by an example based on a 2NN algorithm. Given the 6 images in Figure 1, their distance matrix, based on L1 color feature (sum of image pixel values) is computed in Table 1.a.

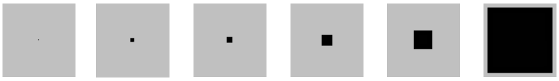

Fig. 1. Images list for 2NN example

Table 1. The distance matrix (a), the two nearest neighbors in the case of 2NN algorithm (b) and the geometrical neighbors (c). On each row, the black cells indicate the elements (columns) neighbors of the row element

Distance	A	B	C	D	E	F
A	-					
B	0.01	-				
C	0.03	0.02	-			
D	0.09	0.08	0.06	-		
E	0.20	0.19	0.16	0.11	-	
F	0.58	0.57	0.55	0.50	0.40	-

(a)

2 Nearest Neighbors	A	B	C	D	E	F
A		■	■			
B	■		■			
C	■	■				
D		■	■			
E			■	■		
F				■	■	

(b)

Topological Neighbors	A	B	C	D	E	F
A		■				
B	■		■			
C		■		■		
D			■		■	
E				■		■
F					■	

(c)

The user runs a query giving the image D as request. The system returns the images B and C (Table 1.b). Then, the user expects to find D at least when he runs the query with the image C as request. However, the system returns A and B. These query results are then surprising and even doubtful for the user. The property of non-symmetry of the kNN have been underlined several times, but seldom criticized as leading to cognitively erroneous results.

[2] http://www.hermitagemuseum.org/fcgi-bin/db2www/qbicSearch.mac/qbic?selLang=English

To avoid kNN to produce this adverse effect, the points have to be relatively uniformly distributed on the representation space. This assumption seems too strong for us and difficult to keep. We can manage this limitation by using the topological models which are symmetric (Table 1 c).

3 Topological Models

We consider a dataset Ω composed by *n* images. Each image is represented as a *p* dimensional numerical feature vector $X(i) = (X_1(i), X_2(i), ..., X_j(i), ..., X_p(i)) \in \mathbb{R}^p$. Therefore each image is a point in $\mathbb{R}^p$ space. A distance measure can be computed for each couple of images, for instance Euclidian or Cosine distance.

The *topology* of a dataset defines how data points are connected to one another. Topology can be represented by a graph, where data points X_j for $j \in [1, \ldots, p]$ represent nodes and the neighborhood relationships denote edges connecting nodes.

Each image is represented as a node in a neighborhood graph. The neighbors of a node represent similar images. Two points (images) are neighbors and connected by an edge if they verify a specific binary relationship. Many models may be used: Delaunay triangulation, relative neighborhood graph, Gabriel graph or minimum spanning tree. We choose the relative neighborhood graph representation for the image database for the reasons presented below. The binary relationship defined by each graph is symmetric. For more details about all the graph models and their properties see [8].

Relative Neighborhood Graph is a related graph where two points α and β are neighbors if they verify the following property: the lune, corresponding to the dashed area in Figure 2, must be empty. Two points α and β are connected by an edge if the following equation is verified:

$$d(\alpha, \beta) \le Max(d(\alpha, \gamma), d(\beta, \gamma)), \ \forall \gamma \in \Omega \setminus \gamma \ne \alpha, \beta \qquad (1)$$

where $d(a,b)$ is a distance between $a, b \in \Omega$ in $\mathbb{R}^p$.

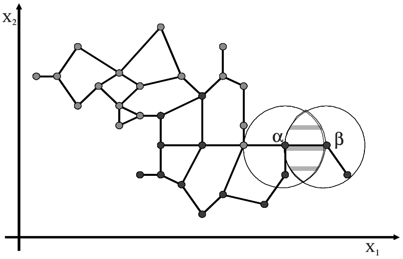

Fig. 2. Example of a Relative Neighborhood Graph in $\mathbb{R}^2$

3.1 Topological Neighborhood

Content based image retrieval is based on similarity measures. In this context, image retrieval requires an understanding of the notion of image *neighborhood*. The neighborhood is generally defined as the set of all points belonging to a given set whose distances from a given point are less than a given positive number. In order to decide which points are neighbors, the *geometrical neighborhood* or the *topological neighborhood* have to be examined. A geometrical neighborhood includes all the points within a certain distance from the sample point. kNN algorithm uses the geometrical neighborhood. A topological neighborhood contains all the points within a certain number of edges from the desired sample point. Two points are linked by an edge only if they satisfy the criteria presented before and does not necessarily involves a minimal distance.

3.2 Choice of the Topological Neighborhood

All neighborhood graphs model the similarity between images represented as points in $\mathbb{R}^p$ space. The relative neighborhood graph is a superset of minimal spanning tree and a subset of Delaunay triangulation [11]. We preferred the relative neighborhood graph because the definition of minimal spanning tree and for Delaunay triangulation is a global one. Therefore, each time we add a new image on the database, or we present a new image in order to perform similar image retrieval, the entire graph needs to be recalculated. Relative neighborhood graph and Gabriel graph are equivalent, they both have local definitions and we can easily insert new points in the graph, without redefining it. The Gabriel graph has more edges than relative neighborhood graph but in practice the computing time is equivalent. In our tests we used the relative neighborhood graph.

The related neighborhood graph is built in $\mathbb{R}^p$ space For p ≥3 the visual representation can be projected into the $\mathbb{R}^2$ space using, for example, the principal components analysis or phylogenic trees [5]. This representation is an alternative to Kohonen maps [6]. It can also be directly built from a given plane, like the first factorial plane if this one preserves the major part of information.

4 Topological Query

Given a query image *I*, and an image database, the user wants to find a set of similar images. When applying *k*NN algorithm, *k* images are returned, closest in distance with the query image *I*. Setting the value of *k* represents another drawback of the kNN approach, in addition to the non-symmetry. Each image has a variable number of neighbors, therefore we can distinguish two situations: either to *limit* the returned results at k items if the query image has more than k neighbors in the target database or to *force* the system to return k images if the query image has less than k neighbors in the target database.

Table 2 shows the distance matrix for eight images (presented in Figure 3) belonging to three semantic categories (“plants”, “fountain” and “mountain”). If we use a query image similar to A (“plants”), intuitively the system should return A, B and C. If we set the value of k to 2, we will obtain only two results, say A and B. Another query has to be performed to obtain the C image, even if it is very similar with the two others. On the other hand, if k value is set to 3 and the query image is similar to D (“fountain”), the system will return D, E and also another image whose distance is closest from the query image. The distance matrix in our example shows H as the second nearest neighbor of the D image, therefore the user will obtain a “mountain” picture, visually not similar with the query image representing a fountain.

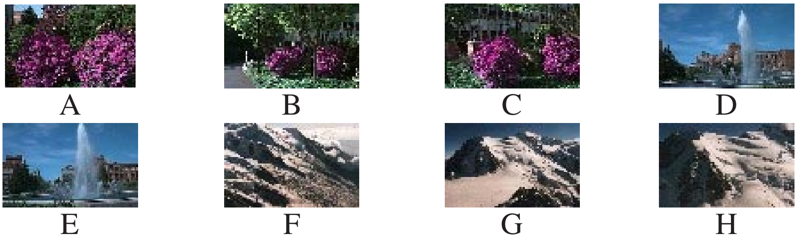

Fig. 3. Image list for three semantic categories: plants (A, B and C), fountain (D and E) and mountain (F, G and H)

Table 2. Distance matrix using the Euclidean distance and 15 color and textural features projected on the two first principal components

Distance	A	B	C	D	E	F	G	H
A	-							
B	2.35	-						
C	2.09	0.50	-					
D	15.44	13.91	14.42	-				
E	14.73	13.32	13.82	1.11	-			
F	8.54	9.11	9.39	11.95	10.88	-		
G	10.44	10.36	10.75	9.19	8.08	3.10	-	
H	10.59	10.19	10.62	7.80	6.71	4.25	1.41	-

The neighborhood graph we propose is build as in Figure 4-a by using color and textural features of images.

A visual exploration of the neighborhood graph shows that in this example the three clusters correspond to semantic classes. Distances between nodes in the neighborhood graph are a good representation of the visual similarity between images.

When a query image I is presented to the system, the following algorithm is applied:

1. Calculate the *p* low-level features corresponding to the representation space $\mathbb{R}^p$;
2. Calculate the distances[3] between the new point and all the existing points;
3. Insert the new point in the neighborhood graph by verifying the criteria specified in section 3.

The advantages of this method are: first, the neighborhood graph is locally updated, it is not entirely rebuild. Second, the user can browse the neighborhood graph and parse all the neighbors of the query image. The number of neighbors is not fixed as in the

[3] We considered the Cosine distance, but other distances may be used as well.

case of kNN algorithm where exactly *k* points are returned as query result. Third, the neighborhood relationship defined by RNG is symmetrical and more appropriated for the browsing process, as shown in section 2.

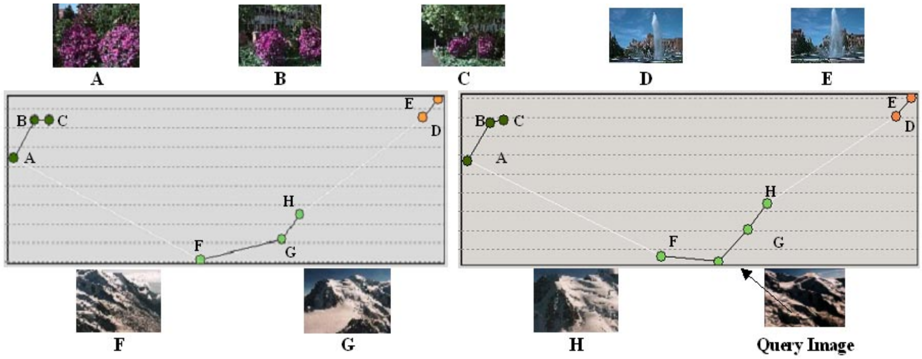

Fig. 4. The relative neighborhood graph. The nodes have different colors representing semantic categories. On left (4-a), without the Query Image ; included on right (4-b)

Figure 4-b presents an example of topological query. The new image is inserted in the neighborhood graph and the user can browse all of its neighbors. A new mountain image is used as query image (it does not belong to the database, but very similar with the other mountain images in the image database).

5 Experimentation

In our experiments we used a set of 259 images divided between six main categories, extracted from the Ground Truth Database[4] (University of Washington). We used predefined image categories as semantic information. We considered the following categories: “Arborgreens”, “Australia”, “Cherries”, “SwissMountains”, “Greenlake” and “SpringFlowers”. We used two categories of features automatically extracted from images: color features (normalized L1 and L2, predominant color) and textural features (the 14 features defined by Haralick in [4]). The numerical features may represent the whole image or objects inside the image. The features we used here are all global features. In a future work we will perform image segmentation and also use shape features on segmented regions.

5.1 Comparison Protocol

For a given representation space, we compare the coherence of a neighborhood in a related neighborhood graph (RNG) with the coherence of *k* nearest neighbors (kNN) of an image. To evaluate this coherence, RNG and kNN are compared in a classification context. Classification performances are usually measured in the term of the

[4] http://www.cs.washington.edu/research/imagedatabase

classic information retrieval notions of recall and precision [10]. Test images are spread over six semantic categories used to compute recall and precision. We specify that the relative neighborhood graph is built from low-level features only.

The recall and precision are defined as follows:

$$recall = \frac{number\ of\ categories\ found\ and\ correct}{total\ number\ of\ categories\ correct} \quad precision = \frac{number\ of\ categories\ found\ and\ correct}{total\ number\ of\ categories\ found}$$

We evaluate the capabilities of the graph used as a classifier, in order to find the category of the query image. To do that, the query image is inserted in the graph structure and the category is decided by its neighbor votes inversely weighted by the length of the edges. The length of the edges represent the distance between two linked nodes. Even if the nodes are neighbors in terms of distances, in RNG two nodes are linked by an edge if they verify the condition (1) presented in section 3. For the kNN classifiers, the category is decided by the votes of k nearest neighbors of the unknown image, inversely weighted by the distance. In our tests we have used the Cosine distance since it is scale invariant and then does not require to normalize the data. The number of neighbors (k) for kNN vary from 1 to 5 and classifier results are compared after a 10 folds cross validation.

5.2 Results and Discussion

The Figure 5 shows the results obtained from our experiments. We observe that RNG outperforms the different kNN models. For kNN we have performed tests up to 30 neighbors (k=30). Our results show that the success rate, the precision and recall in the case of RNG are superior of the best kNN results, obtained for 4NN. We can explain the better results for RNG by examining the category prediction process. In the case of kNN, exactly k neighbors will vote the query image category. In the case of RNG, the number of neighbors is adaptative according to the topology.

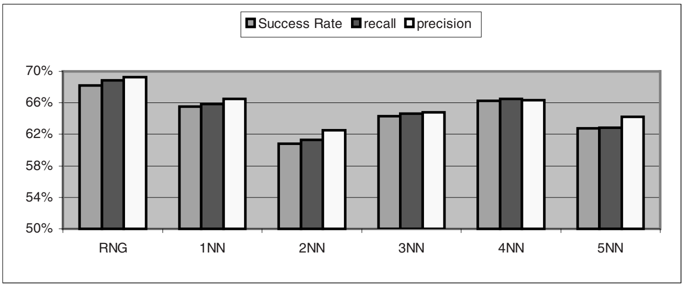

Fig. 5. Relative Neighbor Graph versus k-Nearest Neighbor (k varies from 1 to 5)

6 Conclusions and Future Work

Search algorithms in image databases usually return k nearest neighbors (kNN) of an image according to a similarity measure. This approach presents some anomalies and is based on assumptions that are not always satisfied. We have examined the causes of these anomalies and we have concluded that image query models have to exploit topological properties rather than the similarity degree. The knowledge inside an image database lies on the topological structure of a set of points (images) rather than on the distance between them. We have proposed a topological representation method based on neighborhood graphs built on automatically extracted image features. On the other side, MPEG 7 standard is set up gradually and query models could be built on MPEG 7 descriptors of multimedia data. Automatic extraction of semantic descriptors of audio-visual content still remains a problem, as well as their pertinent exploitation. The topological model proposed in this paper offers an exploratory analysis of MPEG 7 descriptors and also allow to use these descriptors in a query process. In a future work we will use MPEG 7 files as data source for our topological model. We also work on a faster retrieval algorithm using related neighborhood graphs.

References

1. P. Aigrain, H. Zhang, and D. Petkovic. "*Content-based representation and retrieval of visual media: A state-of-the-art review*". In Multimedia tools and applications, vol. 3, pp. 179–202. Kluwer Academic Publishers, 1996.
2. J.-P. Barthélemy & A. Guénoche, *Trees and proximity representations*, New York : John Wiley & Sons, 1991.
3. C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, and W. Equitz. "*Efficient and Effective Querying by Image Content*". Journal of Intelligent Information Systems, vol. 3, no. 3, pp. 231–262, 1994.
4. Haralick, R. M. Shanmugan, K., Dinstein, I. *Texture features for image classification*. IEEE Transactions Systems, Man and Cybernetics, Novembre 1973, vol. 3, pp.610–621.
5. D.M. Hillis, M.W. Allard and M.M. Miyamoto. *Analysis of DNA sequence data: phylogenetic inference*. Methods Enzymol 1993;224:456–87.
6. T. Kohonen, *Self-Organizing Maps*. Berlin/Heidelberg, Springer, 1995, vol. 30.
7. T.M. Mitchell, *Machine Learning*. Computer Science, New York, 1997.
8. F. Preparata, M. I. Shamos. "*Computational Geometry. An introduction*". Springer-Verlag, New-York, 1985.
9. Y. Rui, T.S. Huang and S-F. Chang. "*Image retrieval: current techniques, promising directions and open issues*". Journal of Visual Communication and Image Representation, Vol. 10, no. 4, pp. 39–62, April 1999.
10. J.R. Smith. "*Image retrieval evaluation*". In IEEE Workshop on Content-based Access of Image and Video Libraries (CBAIVL'98), pp. 112–113, June 1998.
11. G.T. Toussaint. "*The Relative Neighborhood Graph of a Finite Planar Set*". Pattern Recognition, 12(4), pp. 261–268, 1980.
12. R.C. Veltkamp and M. Tanase. "*Content-based image retrieval systems : A survey.*" Technical report UU-CS-2000-34, Department of Computing Science, Utrecht University, October 2000.
13. E. Weisstein. "*The CRC Concise Encyclopedia of Mathematics*". CRC Press, 1998.