Short Notes

Similarity Evaluation in Image Retrieval Using the Hough Transform

Eugenio Di Sciascio and Augusto Celentano[1]

Politecnico di Bari, Dipartimento di Elettrotecnica ed Elettronica, Bari, Italia

This work investigates the use of orientation features, computed using the Hough transform, as a criterion for image similarity evaluation in content based picture retrieval. The context of this work is the management of thematic catalogues, in which coherence in the meaning of the image contents can be relied upon to a certain degree. The vector space model, a well-assessed technique in textual information retrieval, is utilized for the retrieval model.

Keywords: image retrieval, image analysis, image databases, image transform, image similarity, image orientation.

1. Introduction

Current technology allows the acquisition, transmission, storing, and manipulation of large collections of images. Yet systems for their classification and retrieval still rely heavily on textual descriptions associated to images. Recently, a number of methodologies, techniques and tools have been studied for identification and comparison of image features in order to develop classification and retrieval systems based on (almost) automatic interpretation of image contents [1-4].

This work focuses on the use of orientation features as a criterion for similarity based retrieval; the Hough Transform [5] algorithm is used for features extraction. We do not claim that the analysis of the image directional features can fulfill all requirements of a content based image retrieval system, but we argue that this feature is well suited for evaluating the visual coherence of a set of non trivial, semantically homogeneous images.

2. Motivations and goals

The approach we consider in this paper is oriented to retrieving images from a thematic database, where the semantic content of the images is limited to a specific domain. Most image collections available in the public domain or through the commercial and professional distribution channels are organized in sub-collections (directories), each covering a separate theme.

The case study we discuss is a small database of eighty public domain images picturing planes.

[1]Current address: Università Ca' Foscari di Venezia, Dipartimento di Matematica Applicata e Informatica, Venezia, Italia

Work supported in part by the Italian MURST in the framework of the project "Basi di dati evolute: Modelli, Metodi e Strumenti".

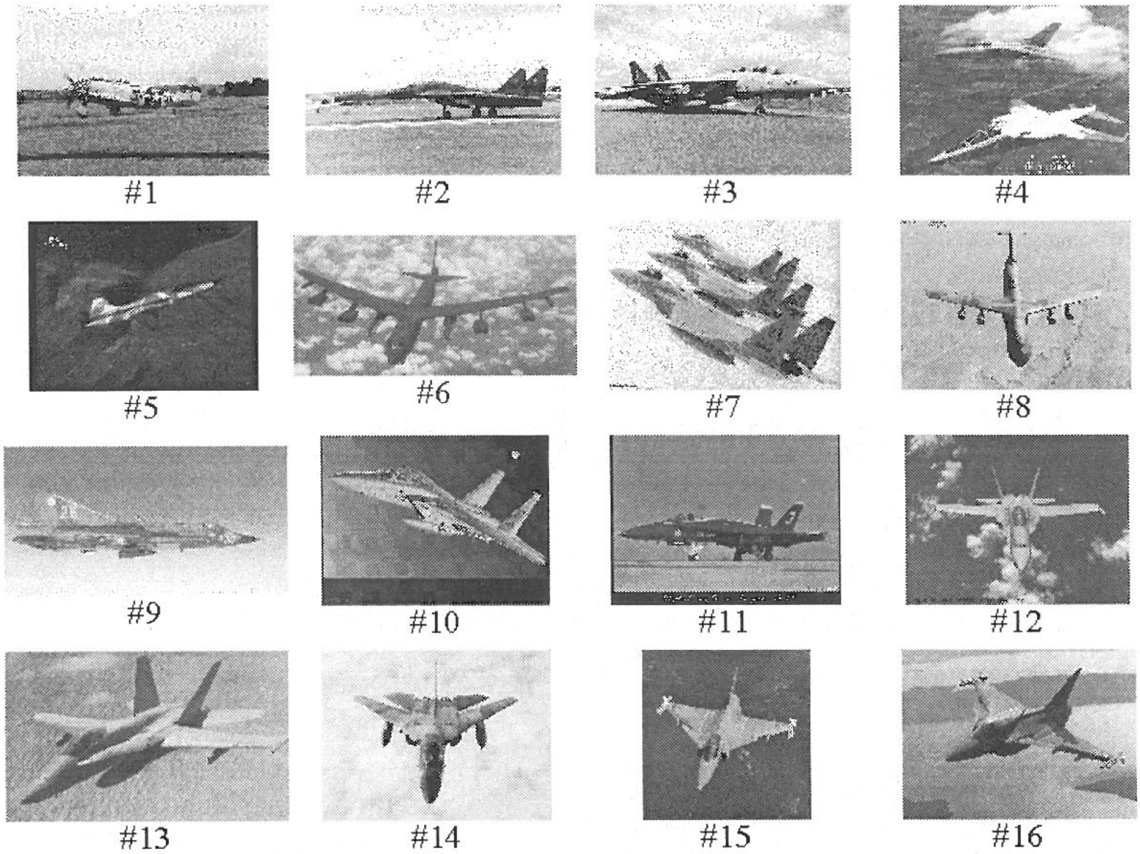

Fig. 1. A sample of the images used

Most of them are true photos, someone is a (good) painting. Size, resolution and number of colors are very different. Also, the image quality is not constant, and in some cases quite low. For space reasons, a set of only 16 images from the test set is used as an example, (Fig. 1). Though very small, the set is representative of the different images contained in the whole collection. Application of the retrieval model is illustrated through the similarity measures found among the images, and with sample sketches standing for user graphical queries.

3. Database population and image analysis

Various models have been proposed for similarity analysis in image retrieval systems. In this work we investigated the application of the vector space model [6], that has been widely used in conventional document retrieval systems. The basic model considers the same N elements term set to identify both stored data and queries, that can be represented as vectors in a N-dimensional space. As index terms, our approach assumes the weights that different directions have within an image. Directions are grouped in ranges of 10°. This value has been selected empirically. Narrower angles improve precision but, due to the richness of image details, they can result in very low recall, while wider angles result in higher recall and in less precision.

The procedure of computing the orientation feature of an image during the database population involves four steps.

First, the images are normalized in number of colors, and contours of the image objects are extracted following the approach of Torre and Poggio [7]. The edge lines are used as an image sketch in computing the Hough Transform, from

	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13	#14	#15
#2	0.968														
#3	0.990	0.973													
#4	0.596	0.550	0.541												
#5	0.323	0.323	0.272	0.899											
#6	0.175	0.167	0.186	0.447	0.546										
#7	0.383	0.409	0.426	0.496	0.497	0.681									
#8	0.558	0.510	0.579	0.560	0.489	0.780	0.706								
#9	0.954	0.867	0.938	0.675	0.395	0.247	0.392	0.618							
#10	0.378	0.394	0.421	0.468	0.512	0.609	0.907	0.748	0.378						
#11	0.975	0.924	0.975	0.590	0.327	0.252	0.461	0.628	0.968	0.444					
#12	0.596	0.544	0.588	0.488	0.421	0.678	0.484	0.903	0.617	0.572	0.609				
#13	0.284	0.286	0.256	0.790	0.873	0.548	0.650	0.563	0.322	0.681	0.313	0.433			
#14	0.572	0.506	0.566	0.515	0.421	0.683	0.702	0.918	0.610	0.742	0.620	0.895	0.505		
#15	0.421	0.366	0.430	0.488	0.465	0.763	0.697	0.903	0.505	0.691	0.523	0.851	0.551	0.884	
#16	0.380	0.384	0.383	0.806	0.846	0.637	0.748	0.698	0.443	0.772	0.431	0.533	0.932	0.596	0.654

Table 1. Similarity coefficients between image vectors

which the weight each direction (i.e., angle) has in building up the whole image is derived.

The procedure decomposes the edges in straight segments, for each edge segment and computes its length and the corresponding slope, within the 0° to 179° range. The computed data are grouped and sorted into an 18 elements array, each element representing the integral over a range of 10° (0°–9°, 10°–19°,...) of the weights corresponding to line directions.

The values are then normalized so that the integral of all the values adds up to 1.0, in order to compare, in subsequent retrieval, the contributions in each direction, regardless of image size.

It is worth noting that, while not trivial, the whole process is automatic and not driven by the user interpretation of the image meaning. It is however obvious, that the initial quality of

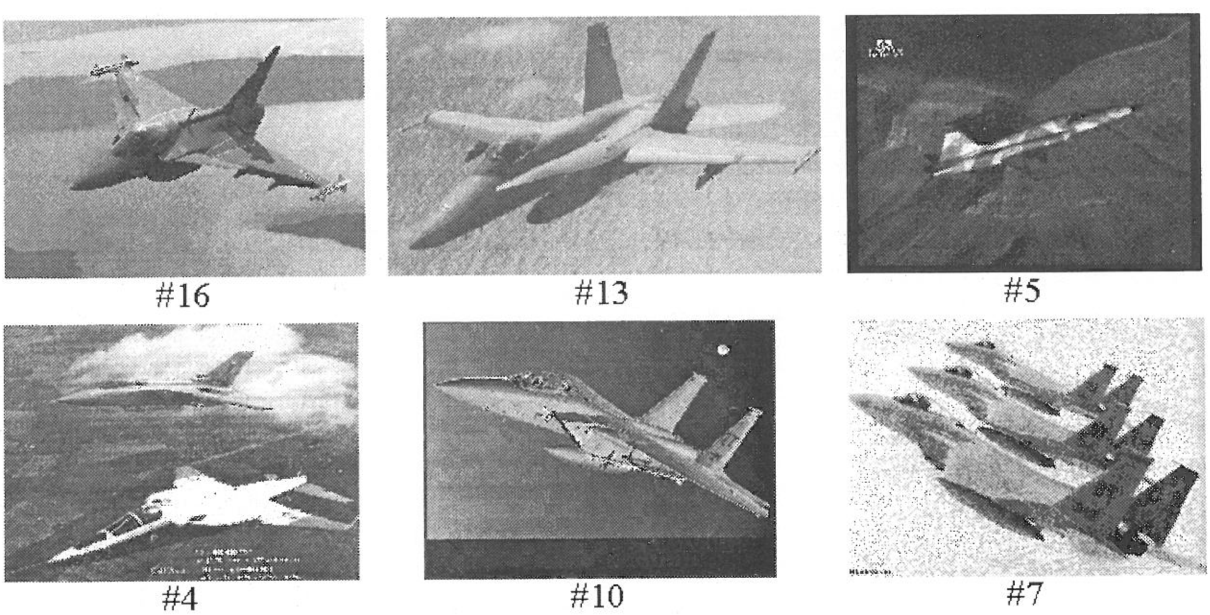

Fig. 2. Ranked results of query with image #16: best five retrieved images

#1	#2	#3	#4	#5	#6	#7	#8
#3 0.990	#3 0.973	#1 0.990	#5 0.899	#4 0.899	#8 0.780	#10 0.907	#14 0.918
#11 0.975	#1 0.968	#11 0.975	#16 0.806	#13 0.873	#15 0.763	#16 0.748	#12 0.903
#2 0.968	#11 0.924	#2 0.973	#13 0.790	#16 0.846	#14 0.683	#8 0.706	#15 0.903
#9 0.954	#9 0.867	#9 0.938	#9 0.675	#6 0.546	#7 0.681	#14 0.702	#6 0.780
#4 0.596	#4 0.550	#12 0.588	#1 0.596	#10 0.512	#12 0.678	#15 0.697	#10 0.748
#12 0.596	#12 0.544	#8 0.579	#11 0.590	#7 0.497	#16 0.637	#6 0.681	#7 0.706

#9	#10	#11	#12	#13	#14	#15	#16
#11 0.968	#7 0.907	#1 0.975	#8 0.903	#16 0.932	#8 0.918	#8 0.903	#13 0.932
#1 0.954	#16 0.772	#3 0.975	#14 0.895	#5 0.873	#12 0.895	#14 0.884	#5 0.846
#3 0.938	#8 0.748	#9 0.968	#15 0.851	#4 0.790	#15 0.884	#12 0.851	#4 0.806
#2 0.867	#14 0.742	#2 0.924	#6 0.678	#10 0.681	#10 0.742	#6 0.763	#10 0.772
#4 0.675	#15 0.691	#8 0.628	#9 0.617	#7 0.650	#7 0.702	#7 0.697	#7 0.748
#8 0.618	#13 0.681	#14 0.620	#11 0.609	#8 0.563	#6 0.683	#10 0.691	#8 0.698

Table 2. The highest six similarity scores of images of figure 1.

the image influences the final result, due to different edge identification. At the end of four steps each image of the database is endowed of a vector that represents "how much" the image (or more precisely the line that forms objects within it) is oriented in different directions.

4. Image similarity evaluation

Queries are formulated in two ways. The first assumes that the query is an image to which the database content is compared, the second uses as a query a drawing made by the user, sketching the distribution of the orientation of the lines of the requested image. In both cases the orientation vector associated to the image (taken from the database or computed as described above) is assumed as the set of query terms.

The similarity function $sim(D,Q)$ between a database image defined by the tuple $D = (d_0, d_1, \ldots, d_{17})$ and the query, also defined as $Q = (q_0, q_1, \ldots, q_{17})$ is computed using the *cosine similarity* coefficient

$$sim(D,Q) = \frac{\sum d_i q_i}{\sqrt{\sum d_i^2 \cdot \sum q_i^2}}.$$

We have also tested other coefficients, namely the Dice and Jaccard coefficients. They provide basically the same results, as far as higher ranking retrieved images are concerned, but with different absolute values.

Table 1 lists the cosine similarity coefficients measured between each pair of the sample images of Fig. 1. Table 2 summarizes the highest six similarity scores for each image. Fig. 2 shows the first 5 images retrieved using image #16 as a query.

In Fig. 2, the first three images retrieved exhibit a good visual coherence with image #16 (image #5 is actually oriented in the opposite direction, but this difference cannot be interpreted by a purely numerical evaluation), while the last two images, whose scores are under 0.8, are quite different.

In general, comparing Tables 1 and 2 with the actual image contents, it can be seen how scores from about 0.8 up constitute a good measure of image similarity, while lower scores correspond to visible differences, and with very low scores (i.e., less than 0.6) almost no visual similarity with the original image can be found. In order to better characterize the orientation comparison, we have introduced a heuristic modification in the ranking computation. This technique can be considered a sort of *a priori* relevance analysis: when an image in the database has a main orientation value (i.e., the orientation whose contribution is a maximum in the image associated vector) that differs from the one of the query by more than 20° in either direction, its similarity score is reduced to 50%. This correction puts such an image out of any meaningful

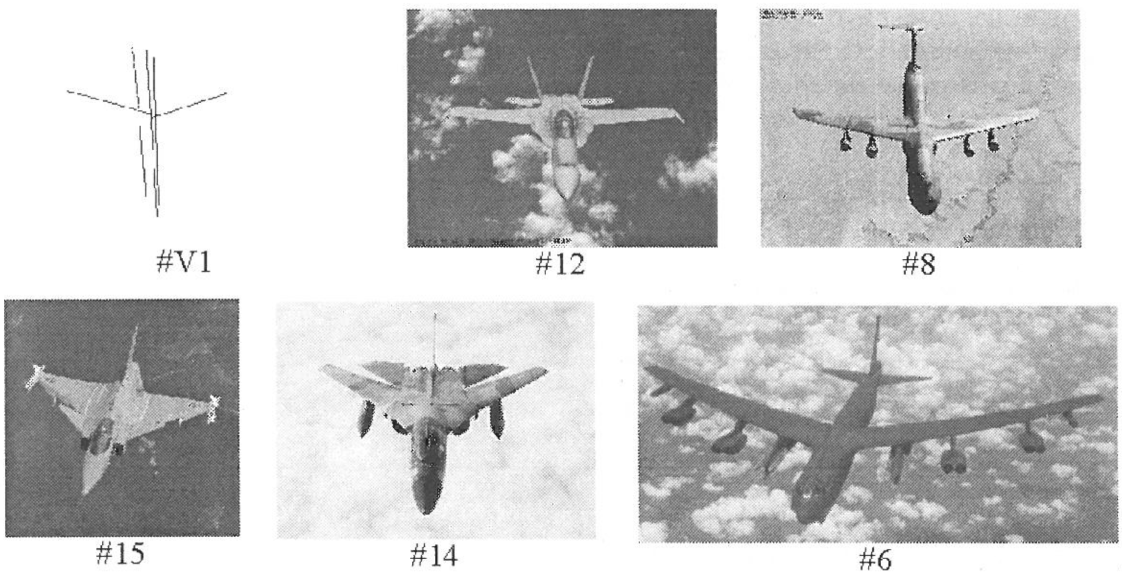

Fig. 3. Ranked results of query with sketch #V1: best five retrieved images

ranking evaluation. This is an admittedly *brute force* correction, but it prevents meaningless retrievals where high computed similarity scores come from matching only secondary details.

It can be noticed that this correction basically increases the weight of the main image orientation, and reduces the weight of the general distribution of lines within the image.

Retrieval through graphic query formulation can be approached by asking the user to draw a sketch of the desired images aspect. In this case, the sketch has only to indicate the distribution of line orientation in the images. In general, scores are lower than when comparing real images, due to the lack of several contributes in the direction vectors. Fig. 3 illustrates the first five ranked images similar to a simple sketch; Table 3 shows the corresponding scores.

Q3	
#12	0.77
#8	0.65
#15	0.62
#14	0.60
#6	0.59
#10	0.32

Table 3. The highest six similarity scores for sketch Q3

Generally, the scores allow good partitioning of images in classes exhibiting coherent visual properties and similar aspects. Local ranking within classes is biased by other images features, the most notable being the influence of contrast, resolution and foreground/background relationships. For the 16 images of Fig. 1 the following clusters can be easily identified by reading Table 2, and validated by visual inspection (but the clusters are not completely disjoint, e.g. image #16 can as well belong to cluster 2):

1. images #1, #2, #3, #9 and #11 (almost horizontal, side views);

2. images #6, #8, #12, #14 and #15 (extending in two orthogonal directions, almost top view);

3. images #4, #5, #13 and #16 (tilted towards left and down);

4. images #7 and #10 (tilted towards left and up).

6. Conclusion

In this work the orientation of images has been investigated as a similarity criterion in content based image retrieval. It has been shown that, although this feature alone cannot fulfill the requirements of a content based retrieval system, it works well in a semantically homogeneous framework. According to this pre-requisite, it can be assumed that no knowledge based interpretation of the contents meaning is required during database population, and image classification can be performed by an automatic process.

References

[1] ACM Multimedia Systems, special issue on Multimedia Databases, 3 (1995), 5/6.

[2] BARBER ET AL., Query by Content for Large On-line Image Collections, Multimedia Systems, an IEEE Tutorial, IEEE Press, 1994.

[3] IEEE Computer, special issue on Content Based Image Retrieval, 28 (1995), 9.

[4] NIBLAK ET AL., The QBIC project: Querying images by content using color, texture, and shape, Proceedings. Of IS&T/SPIE (1993), 1908.

[5] D. C. W. PAO, H. F. LI AND R. JAYAKUMAR, Shapes Recognition Using the Straight Line Hough Transform: Theory and generalization, IEEE Trans. on PAMI, 14 (1992), 11.

[6] G. SALTON, Automatic Text Processing, Addison Wesley, 1989.

[7] V. TORRE, T. POGGIO, On edge detection, IEEE Trans. on PAMI, 8 (1986), 147.

Received: September, 1996
Accepted: October, 1996

Contact address:
Eugenio Di Sciascio and Augusto Celentano
Politecnico di Bari
Dipartimento di Elettrotecnica ed Elettronica
Bari, Italia

EUGENIO DI SCIASCIO received the laura "cum laude" degree in Electronic Engineering from University of Bari in 1989 and the Ph.D. degree in 1994. In 1992 he joined the University of Lecce as an assistant professor. Since 1995 he is with Politecnico di Bari where he currently teaches "Fundamentals of Computer Science". His research interests include image processing and advanced information systems. He is member of IEEE and ACM.

AUGUSTO CELENTANO is Full Professor of Computer Science at University Ca' Foscari of Venice. He took a Master Degree in Electronic Engineering from Politecnico di Milano in 1973. His research interests cover information systems, multimedia systems, human engineering issues in collaborative systems, and the use of multimedia technology in distance education and professional training. He has published more the 40 papers on international Scientific Journals and Conference Proceedings. He is member of ACM and IEEE.