Content-Based Image Retrieval: on the Way to Object Features

Abstract

Content-based systems retrieve images based on low-level features (color,texture) while the user usually seeks some objects from real world. As segmentation is never accurate, such systems do not allow the use of powerful feature during retrieval (shape, structure). We propose a new system that relies on a hierarchy of segmentations, in order to handle some artifacts. Besides, it allows using some object-related features for indexing (shape, structure). We also present some result on a 600 images database from Corel.

1. Introduction

Indexing and retrieving images from repositories are still open issues today. Basically, content-based systems extract from each image a signature composed of low-level features. Then, they handle a request by comparing two signatures together.

Content-based systems face a crucial limitation today: they allow to retrieve images based on low-level features ("stuffs"), while users seek a more semantic-based similarity ("things"). For instance, they may want to formulate queries such as "images that depict cars". This raises two kinds of consequences. First, we need tools that allow us to bridge the gap between low-level features and semantics. Even if the so-called semantic gap prevents such a direct path, the use of structural description could level signatures to a more user-intuitive meaning. Second, the system should allow users to formulate their queries so as to be the closest to what they have in mind. Classical query-by-example paradigm allows them to retrieve images that are judged similar from a given one. However, this paradigm seems limited when handling queries related to objects, as the system is not able to generalize what users actually seek from the query image.

In this paper, we present a system that contributes to bridge the gap between low-level features and semantics, by handling a structural level of description. It relies on a multi-level segmentation step that prevents it from being too dependent of segmentation mismatch. Besides, it allows using several region-based features (shape, spatial layout) in order to retrieve multi-regions patterns from images. Finally, the system allows users to formulate model-based queries, which are more adapted when searching for objects.

2. Related work

The first content-based systems made use of global features, considering the whole image. Hence, color histograms can lead to quite good results, when the sought pattern has a unique color [9]. However their use are quite limited to stuff-based queries since objects are not extracted from background.

In order to deal with object-level information, a lot of work relies on a segmentation step: pixels are grouped according to several low-level criteria (color, texture) into regions. During comparison, several features are used for each region, such as color, texture and shape [1].

However, as segmentation alone is never fully accurate, extracted regions do not always match semantic objects. This leads to irrelevant extracted features. Another way is to segment images with user assistance [2]. Once regions have been manually segmented, the use of shape features in order to compare regions can be very effective [4] [5].

Such user dependence is too strong a limitation. That is why pixel-based methods have been extensively used in last years, so as not to do any segmentation at all. Hence, Schneidermann et al. [6] propose a Bayesian classifier, from a wavelet-based image description. Good quality results have been presented on queries such as "cars" or "face". However, statistical approaches are very time consuming. Besides, calibrating data are statically set during training and no further changeable. In this view, such methods can only leads to pre-defined queries ("find images of this kind of object"), which is too limitative in a content-based approach.

When regions have been extracted from image, they may be described by intrinsic features as previously described (color, texture, shape), but also considering their spatial layout (structure). It is obvious that such a description is able to strongly level features to a object-level query. In this

view, the system SaFe [8] stores spatial relationships between regions, each characterized by locations, size, and low-level features. In a narrower domain, Forsyth and Fleck [3] are able to find naked people in images, by recognizing some geometric-constrained structures from limbs. However, they integrate a lot of ad-hoc procedures that prevent from their generalization. Once again, segmentation is the limiting factor, as one has to extract relevant structures (related to a real object) and not accidental combinations of patches with no relation to the 3D world at all.

3. General framework

Even if robust segmentation is never available in unconstrained domain, it allows the use of powerful, object-level features such as shape or structural layout. That is why our framework does not rely on a single segmentation, but on a hierarchy of segmentations instead. This one is build by successive perceptual groupings on low-level primitives such as color regions, from strong details to rough description. During grouping, we use both low-level criteria (e.g. distance between colors) and geometric ones, such as edges smooth continuity between regions, or the creation of regions as compact as possible [11].

This multi-level image description allows us to handle several segmentation mismatches. Besides, we can make intensive use of structural information during retrieval: when comparing two objects, we use one-to-one correspondence between parts instead of comparing the whole object. This strongly increases the accuracy of comparison.

Finally, we propose a model-based querying system, which is able to help users formulating general, object-level queries.

3.1. Basic notations and definitions

We call *model* the query formulated by users (See figure 1(a)). It consists on one object, composed of several parts (object model sub-parts), denoted $M = M_1, \ldots, M_n$. Each sub-part M_i is characterized by several features, like shape and spatial relationships with the whole object.

(a) (b) (c)

Figure 1. Example of model (a), original Image (b), oversegmented image (c).

We call *region tree* the hierarchy of perceptual groupings. It is built from an oversegmented image [11]: when two regions R_i and R_j are merged together in a third region R_k during perceptual grouping, corresponding nodes R_i and R_j from region tree are set to be sons of node R_k. Figure 2 shows a simplified example of region tree obtained from oversegmentation of figure 1(c) (original image is shown on figure 1(b)). Note that regions from oversegmentation correspond to the leaves of region tree.

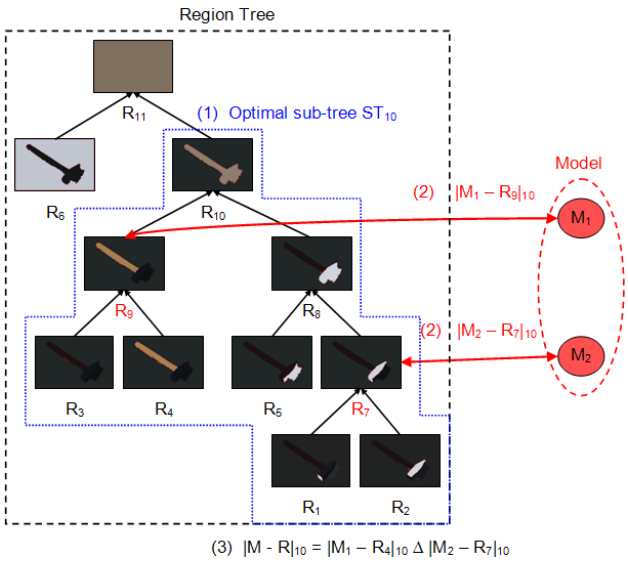

Figure 2. Example of region tree and model seeking

We call *semantic nodes* those from region tree that correspond to real object in the 3D world (R_7 and R_9 on figure 2). Note that, wherever a semantic node may be in the region tree, it can be recognized and associated to a model sub-part. It offers a great flexibility compared to standard segmentation, where the semantic node would have been extracted as a *result* of the segmentation. For instance, in figure 2, the head of hammer has been wrongly merged with the shadow from background (R_8). However, as the head appears deeper the tree (R_7), it could be matched to a model sub-part, though.

3.2. Matching a model with a region tree

Matching a model M with a region-tree R consists in computing the global distance $|M - R|$. We use a three-step process (figure 2). It first finds an *optimal sub-tree* ST_k (1), that contains all semantic nodes and the fewest other nodes as possible, i.e. the sub-tree which best represents the object extracted from the background. Since there is no robust method to extract such optimal a sub-tree, we consider all possible sub-trees ST_k.

Then the process matches each model sub-part M_i with

the region R_j from ST_k that best corresponds (2). To this end, several features (shape, spatial layout...) are used to compute, for each matching from each sub-tree ST_k a single distance $|M_i - R_j|_k$. Then, a global distance $|M - R|_k$ between the whole model and the sub-tree is computed (3), based on distances $|M_i - R_j|_k$. Evaluating $|M - R|_k$ consists in finding as many as possible one-to-one matches between object sub-parts M_i and regions R_j from sub-tree ST_k, while minimizing each distance between them: $|M_i - R_j|_k$.

The combination of distances $|M_i - R_j|_k$ in order to derive a global distance (denoted Δ in figure 2), is based on Dempster-Shafer theory of belief [7] which is especially well-suited for this. We do not detail this process here.

4. Feature space

We now describe the features used to compare regions R_j from region tree to each model sub-parts M_i ($|M_i - R_j|_k$). At this level of description (object), we consider that the most relevant features are related to shape. A lot of methods exists, which could be divided into two classes: those that describe a shape as a pixel-based spatial distribution (region-based) and those that relies on contour. We use both of them since there is broad agreement [10] that they are complementary. We also use structural features.

4.1. Shape features: ART and CSS

Angular Radial Transform (ART) is a region-based image descriptor. It is scale, rotation and translation invariant. It consists in a complex orthogonal unitary transform defined on a unit disk in polar coordinates [4].

Curvature Scale Space (CSS) [5] is a closed contour-based shape descriptor. Like ART, it is scale, rotation and translation invariant. The CSS representation of a closed contour is created by tracking the position of inflection points, while the contour is altered by low-pass Gaussian filters of variable widths. As the width of Gaussian filter increases, insignificant inflections are eliminated from the contour and the shape becomes smoother. The inflection points that remain present at the end are expected to be salient object characteristics.

4.2. Structural features

Content-Based systems usually considers objects on their wholes, without any structural information, even if it may represent additional information. For instance, if users want to retrieve some flags with special patterns, the structural layout of the components is far more important than the whole object, whose rectangular shape is not informative.

In this view, we introduce three structural features for each region: (1) relative position regarding the whole object; (2) relative size regarding the whole object; (3) relative orientation regarding the whole object. Note that all these features are *relative* to the whole object. Consequently, they depend on the sub-tree chosen, which stands for the whole object. Thus, we need a rough estimation of size and orientation of this area in the image. Therefore, we use the bounding ellipse E_k of the sub-tree ST_k under estimation as a reference.

When we want to match region R_j from a subtree ST_k to a model sub-part M_i, we perform a registration of sub-tree ST_k to whole model M, thanks to their bounding ellipse (see figure 3). We compute an affine transformation T that register bounding ellipse E_k of ST_k to the bounding ellipse E of the whole model M. Then, T is applied on bounding ellipses E_j of R_j and we compare $T(E_j)$ and E_i (bounding ellipse of M_i).

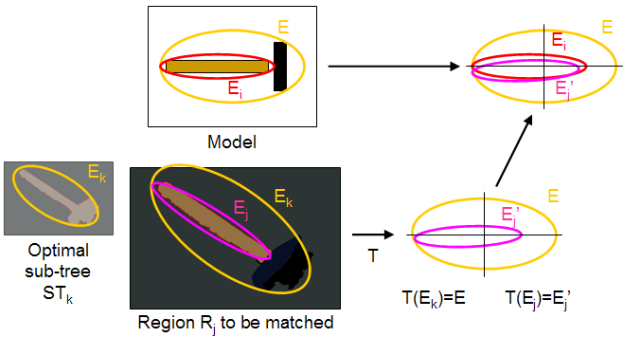

Figure 3. Structural Feature extraction

More precisely, we use three features: (1) Euclidean distance between centroids of bounding ellipses $T(E_j)$ and M_i; (2) area ratio of $T(E_j)$ and M_i; (3) orientation difference between bounding ellipse of $T(E_j)$ and of M_i.

5. Results

Tests have been run on 600 images from Corel database. Indexing step consists in oversegmenting each image, performing perceptual grouping and extracting features. It is conducted off-line, and has to be done only once. It roughly takes 1 second per image.

Model matching takes a linear time against database size. Here (600 images), it takes 5 seconds on a 1.7 GHz computer. The matching returns for each image an overall similarity measure regarding the model. It ranges from 0 (no similarity) to 1 (perfect similarity). Hence, it allows ranking of results. Figure 4 shows some results for query *hammer*. Several hammer-shaped objects are returned with good similarity score. Figure 5 shows some results for query *flag with three vertical patterns*. Results are quite efficient as the first

six results perfectly match the model. The canadian flag got a lower score due to its bigger white pattern on middle (not to the red leaf). Next three results get a significant lower score while last one has a 0 similarity, as expected.

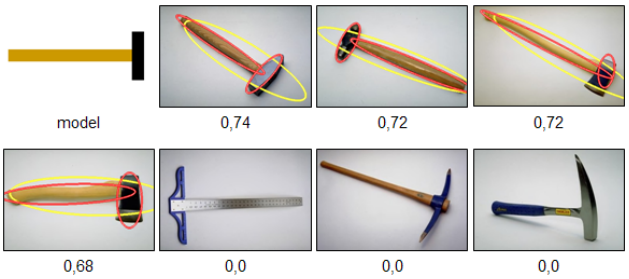

Figure 4. Example of results (query *hammer*)

Figure 5. Example of results (query *flag*)

Figure 6 shows recall-precision curves on 600 images. For query *hammer*, the system correctly retrieves 6 images over the 8 needed. For query *flag*, results are excellent, as the system retrieves all the 7 flags needed.

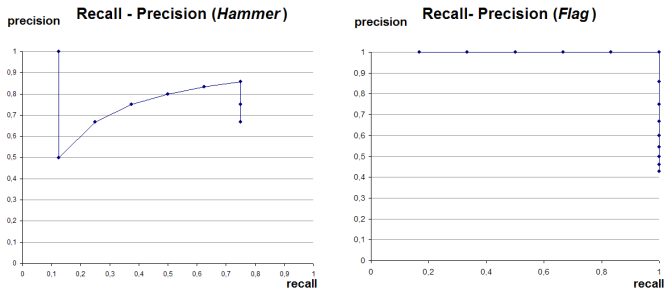

Figure 6. Recall Precision on 600 images.

6. Conclusion

We have presented a new content-based system, that relies on a hierarchy of segmentations. It allows to handle many segmentation mismatches and also to use structural features during matching. Besides, we use a model-based query paradigm. Further works will be directed on the relative weighting of features. As a matter of fact, increasing the relative weight of one feature modifies the kind of similarity sought, and could help to improve the quality of result. This could be done by designing new interface for query.

References

[1] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: image segmentation using expectation-maximization and its application to image querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1026–1038, 2002.

[2] M. Flickner, H. Sawney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, and D. Petkovic. Query by image and video content: the qbic system. *IEEE Special Issue on Content-Based Picture Retrieval System*, 28(9):23–32, 1995.

[3] D. Forsyth and M. Fleck. Automatic detection of human nudes. *International Journal of Computer Vision*, 32(1):63–77, 1999.

[4] W.-Y. Kim and Y.-S. Kim. A new region-based shape descriptor. In *Mpeg Meeting, TR 15-01*, Pisa, Dec. 1999.

[5] F. Mokhtarian and A. Mackworth. A theory of multiscale, curvature-based shape representation for planar curve. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14:789–805, 1992.

[6] H. Schneiderman and T. Kanade. A statistical method for 3d object detection applied to faces and cars. In *proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'00)*, 2000.

[7] G. Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, 1976.

[8] R. Smith and S. Chang. Integrated spatial and feature image query. *Multimedia Systems*, 7(2):129–140, 1999.

[9] M. Swain and D. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.

[10] T. Zaharia and F. Prêteux. Comparative study for 3d and 2d/3d shape descriptors. *Research Report ISO/IEC JTC1/SC29/WG11, MPEG04/10657*, 2004.

[11] N. Zlatoff, B. Tellez, and A. Baskurt. Region-based perceptual grouping: a cooperative approach based on dempster-shafer theory. In *proc. of IS&T / SPIE Electronic Imaging, Image Processing*, 2006, to appear.