Visual Image Retrieval by Elastic Matching of User Sketches

Alberto Del Bimbo, *Member, IEEE,* and Pietro Pala, *Member, IEEE*

Abstract—Effective image retrieval by content from database requires that visual image properties are used instead of textual labels to properly index and recover pictorial data. Retrieval by shape similarity, given a user-sketched template is particularly challenging, owing to the difficulty to derive a similarity measure that closely conforms to the common perception of similarity by humans. In this paper, we present a technique which is based on elastic matching of sketched templates over the shapes in the images to evaluate similarity ranks. The degree of matching achieved and the elastic deformation energy spent by the sketch to achieve such a match are used to derive a measure of similarity between the sketch and the images in the database and to rank images to be displayed. The elastic matching is integrated with arrangements to provide scale invariance and take into account spatial relationships between objects in multi-object queries. Examples from a prototype system are expounded with considerations about the effectiveness of the approach and comparative performance analysis.

Index Terms—Image database, image retrieval by sketch, shape similarity-based retrieval, elastic matching.

1 Introduction

The intrinsic visuality of the information contents associated with pictorial data advises against the use of indexing and retrieval based on textual keywords as traditionally used in text documents. Iconic indexes have been proposed in [25], to effectively support image retrieval by content. Iconic indexes may be in the form of symbolic descriptions of pictorial data or pictorial data relationships but may also include the actual values of object features, or be in the form of abstract images taking the salient characteristics of the original image. The use of iconic indexes naturally fits with the accomplishment of image retrieval according to visual querying by-example. In this approach, the user reproduces, on the screen, the approximate visual representation of the pictorial contents of images to be retrieved, and retrieval is reduced to the matching of the user visual representation against image representations in the database.

A number of techniques have appeared in the literature, which deal with content representation and visual retrieval of single images; differences between these approaches are related to the types of facets of pictorial data that are taken into account. Representation of image content in terms of spatial relationships has been expressed through symbolic strings that capture relative object positions either in the image [3] or in the scene space [7]. To reduce the complexity of the representation, objects are usually abstracted as simplified geometrical elements such as points or minimum enclosing rectangles. In this approach, further developed by other authors in [4], [6], [13], [16], [17], visual queries are parsed into symbolic strings [5]. Retrieval is thus reduced to check the match between two symbolic strings.

• *The authors are with the Dipartimento di Sistemi e Informatica, Università di Firenze, 50139 Firenze, Italy.*
E-mail: {delbimbo, pala}@aguirre.ing.unifi.it.

Manuscript received May 15, 1995; revised Apr. 8, 1996. Recommended for acceptance by L. Shapiro.
For information on obtaining reprints of this article, please send e-mail to: transpami@computer.org, and reference IEEECS Log Number P96120.

Querying by color or texture similarity has been proposed in several systems [2], [14], [15], [22], [23], [24]. Queries typically request images that contain colors or textures similar to those selected from a menu or a sample reference image. Matching is usually performed by comparing global measures such as histograms [24], or evaluating a distance in the original [10], [20] or a transformed feature space [14], [23].

Retrieval by content based on similarity between imaged object shapes and user drawn sketches has been addressed by a few authors [8], [10], [11], [12], [20]. In this case, the problem is complicated by the fact that a shape does not have a mathematical definition that exactly matches what the user feels as a shape. Well-known distance measures commonly used in mathematics are not suitable to represent shape similarity as perceived by humans in the reality. Human perception is not a mere interpretation of a retinal patch, but an active interaction between the retinal patch and a representation of our knowledge about objects.

Solutions proposed in the literature follow different approaches and emphasize different aspects of the problem. In the QVE system [11], retrieval by shape similarity is carried out by evaluating the correlation between a linear sketch and edge images in the database. High values of correlation require that the shape drawn by the user must be close to the shapes in the images, which is nearly impossible in practice. To model user imprecision, the correlation is evaluated with respect to a search area with limited horizontal and vertical shifts. In [12], shapes are represented as an ordered set of boundary features. Each boundary is coded as an ordered sequence of vertices of its polygonal approximation. Features are collections of a fixed number of vertices. This representation allows to roughly evaluate similarity as the distance between the boundary feature vector of the query and those associated with the target images. Boundary features of objects in database images are organized into a quite complex index tree structure. Improvements of this approach, with more effective query

0162-8828/97/$10.00 ©1997 IEEE

processing, have been proposed in [19]. Shape representation based on global features such as area, circularity, eccentricity, major axis orientation and moment invariants has been used in the QBIC system [10], [20]. A reliable characterization of shapes requires the extraction of a great number of features, for most of which, there is no warranty that our notion of perceptive closeness is mapped into the topological closeness in the feature space. In the QBIC system, shape similarity is evaluated as the weighted Euclidean distance in a low dimensional feature space.

In this paper, we propose a different measure of shape similarity, based on elastic deformation of user sketches to match image data. Similar techniques have been previously used for optimization problems [9] and trajectory fitting [21], [27]. Elastic matching promises to approximate human ways of perceiving similarity and to possess a remarkable robustness to shape distortion. In this approach, the sketch is deformed to adjust itself to the shapes of the objects in the images. The match between the deformed sketch and the imaged object, as well as the elastic deformation energy spent in the warping are used to evaluate the similarity between the sketch and the image. The elastic matching is integrated with arrangements to provide scale and partial rotation invariance, and with filtering mechanisms to prune the database.

In the following, in Section 2, the elastic approach to shape matching is introduced, expounding the model of shape similarity, the numerical solution and how similarity ranks of the matched images are obtained. In Section 3, it is expounded how spatial relationships between objects are represented and used to retrieve images. In Section 4, evidence about the effectiveness of the approach is provided with retrieval examples and a comparative performance analysis.

2 The Elastic Approach to Shape Matching

Suppose we have a one-dimensional sketched template, modeled by a second order spline $\vec{\tau} = (\tau_x, \tau_y) : \mathbf{R} \mapsto \mathbf{R}^2$ (that is a piecewise first degree polynomial function). We will always assume that the template is parameterized with respect to the arclength, and normalized so as to result of length one.

We have an image $I : \mathbf{R}^2 \mapsto [0, 1]$—we suppose the luminance at every point normalized in [0, 1]—that we search for a contour with a shape *similar* to that of $\vec{\tau}$. We have italicized the word similar to stress that, in general, the image will contain no contour *exactly* equal to the template. It is not just a matter of noisy images, which we can, to a limited extent, model and cope with. The image and the template can be different to begin with. This makes traditional template matching brittle.

To make a robust match even in the presence of deformations, we must allow the template to warp. If $\vec{\theta} = (\theta_x, \theta_y) : \mathbf{R} \mapsto \mathbf{R}^2$ is the deformation, then the deformed template $\vec{\phi}$ (also parameterized with respect to arclength) is given by:

$$\vec{\phi}(s) = \vec{\tau}(s) + \vec{\theta}(s).$$

The template must warp taking into account two opposite requirements. First, it must follow as closely as possible the *edges* of the image. The match between the deformed template and the edge image I_E can be measured as:

$$\mathcal{M} = \int_0^1 I_E(\vec{\phi}(s))ds.$$

If we normalize I_E so that $I_E \in [0, 1]$, then $\mathcal{M} \in [0, 1]$. A value $\mathcal{M} = 1$ means that the template lies entirely on image areas where the gradient is maximum (i.e., on image edges), while $\mathcal{M} = 0$ means that the template lies entirely in areas where the gradient is null.

The second requirement to be taken into account is the deformation of the template. We measure an approximation of the *elastic deformation energy* for the template given by:

$$\mathcal{E} = \mathcal{S} + \mathcal{B} = \alpha\int_0^1 \left[\left(\frac{d\theta_x}{ds}\right)^2 + \left(\frac{d\theta_y}{ds}\right)^2\right] ds + \beta\int_0^1 \left[\left(\frac{d^2\theta_x}{ds^2}\right)^2 + \left(\frac{d^2\theta_y}{ds^2}\right)^2\right] ds.$$

The quantity $\mathcal{S}$, depending on the first derivative, is a rough measure of how the template $\vec{\tau}$ has been *strained* by the deformation $\vec{\theta}$, while the quantity $\mathcal{B}$, depending on the second derivative, is an approximate measure of the energy spent to *bend* the template. Therefore, we assume $\mathcal{S}$ and $\mathcal{B}$ to be, respectively, measures of the *strain energy* and *bend energy* associated with the deformed template $\vec{\tau} + \vec{\theta}$ with respect to the original template $\vec{\tau}$.

Note that the elastic deformation energy depends only on the first and second derivatives of the deformation $\vec{\theta}$. This prevents penalizing discontinuities and sharp angles that are already present in the template $\vec{\tau}$, and penalizes only the degree of departure from those discontinuities. Also, since the elastic deformation energy depends only on the derivatives of $\vec{\theta}$, a pure translation of the template, for which $\vec{\theta}$ is constant, does not result in additive cost. This makes our scheme inherently translation invariant.

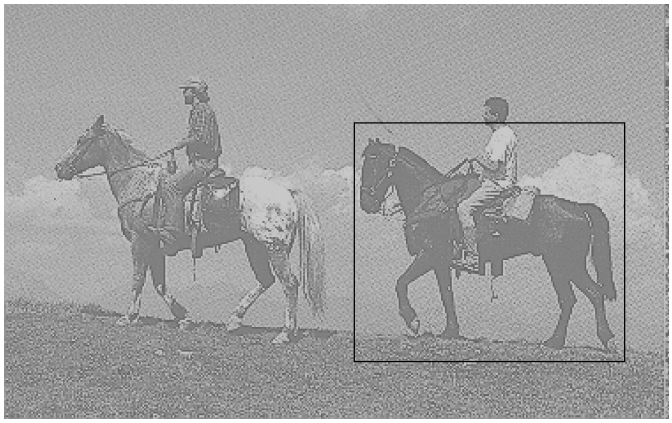
In order to discover the similarity between the original shape of the template and the shape of the edge areas on the image, we must set some constraints on deformation [26]. Hence, our goal is to maximize $\mathcal{M}$ while minimizing $\mathcal{E}$. This can be achieved by minimizing the compound functional:

$$\mathcal{F} = \int_0^1 \alpha\left[\left(\frac{d\theta_x}{ds}\right)^2 + \left(\frac{d\theta_y}{ds}\right)^2\right] + \beta\left[\left(\frac{d^2\theta_x}{ds^2}\right)^2 + \left(\frac{d^2\theta_y}{ds^2}\right)^2\right] - I_E(\vec{\phi}(s))ds. \quad (1)$$

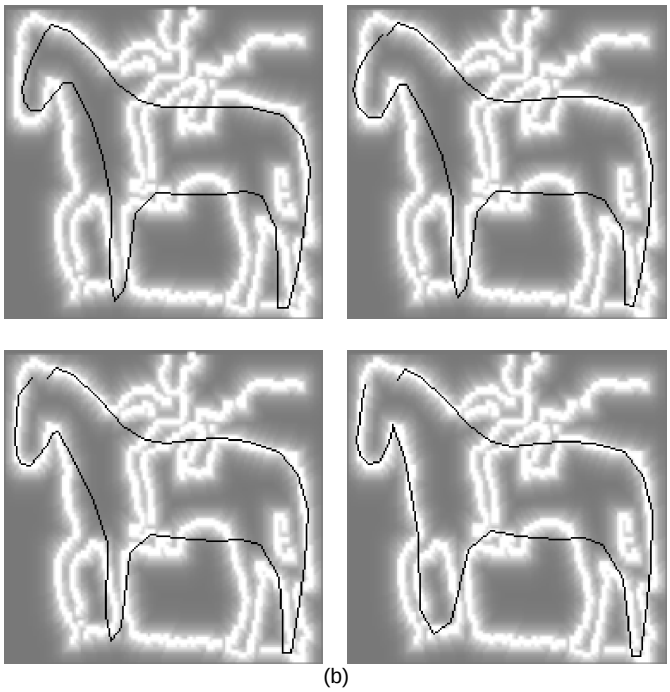
2.1 Numerical Solution

Since the compound functional (1) involves second order derivatives, a numerical solution using piecewise third-degree (fourth-order) splines, which guarantees the necessary degree of continuity, can be derived [1].

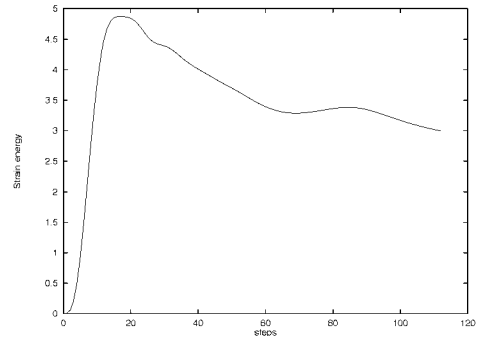
If $B_i(s)$ are the fourth order B-splines, the deformation $\vec{\theta}$ can be written as:



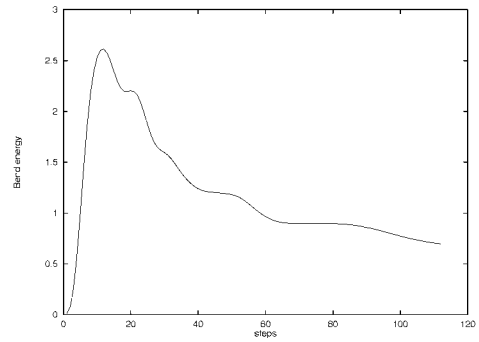
(a)



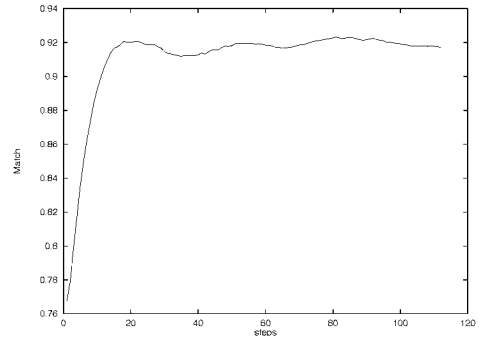
(b)



$\mathcal{S}$



$\mathcal{B}$



$\mathcal{M}$

(c)

Fig. 1. Elastic deformation of a horse-like template over an edge image with plots of $\mathcal{S}$, $\mathcal{B}$, and $\mathcal{M}$: (a) the original image; (b) different steps of the deformation process; (c) plots of $\mathcal{S}$, $\mathcal{B}$, and $\mathcal{M}$.

$$\vec{\theta}(\vec{C}, s) = \sum_{i=1}^{N} \vec{c}^{\,i} B_i(s), \tag{2}$$

where $\vec{C} = (\vec{c}^{\,1}, \ldots, \vec{c}^{\,N})$, with $\vec{c}^{\,i} = (c_x^i, c_y^i)$ and c_x^i, c_y^i real coefficients. The minimization of the functional $\mathcal{F}(\vec{C})$ obtained substituting (2) into (1), with respect to variables $(c_x^1, \ldots, c_x^N, c_y^1, \ldots, c_y^N)$, can be achieved through a gradient descent technique. Thus, variables are determined iteratively according to:

$$\vec{C}(k+1) = \vec{C}(k) - \epsilon \nabla \mathcal{F}(\vec{C}),$$

where:

$$\nabla \mathcal{F}(\vec{C}) = \left(\frac{\partial \mathcal{F}}{\partial c_x^1}, \ldots, \frac{\partial \mathcal{F}}{\partial c_x^N}, \frac{\partial \mathcal{F}}{\partial c_y^1}, \ldots, \frac{\partial \mathcal{F}}{\partial c_y^N} \right).$$

It can be derived that:

$$\frac{\partial \mathcal{F}}{\partial c_x^k} = \alpha \int_0^1 2 \left[\sum_{i=1}^{N} c_x^i B_i'(s) B_k'(s) \right] ds + \beta \int_0^1 2 \left[\sum_{i=1}^{N} c_x^i B_i''(s) B_k''(s) \right] ds - \int_0^1 \frac{\partial I_E(\vec{\phi}(s))}{\partial x} B_k(s) ds,$$

and

$$\frac{\partial \mathcal{F}}{\partial c_y^k} = \alpha \int_0^1 2\left[\sum_{i=1}^{N} c_y^i B_i'(s) B_k'(s)\right] ds + \beta \int_0^1 2\left[\sum_{i=1}^{N} c_y^i B_i''(s) B_k''(s)\right] ds - \int_0^1 \frac{\partial I_E(\vec{\phi}(s))}{\partial y} B_k(s) ds,$$

where:

$$B_k'(s) = \frac{dB_k(s)}{ds} \text{ and } B_k''(s) = \frac{d^2 B_k(s)}{ds^2}.$$

Taking the summations out of the integrals, we obtain:

$$\frac{\partial \mathcal{F}}{\partial c_x^k} = 2\alpha \sum_{i=1}^{N} c_x^i f(i,k) + 2\beta \sum_{i=1}^{N} c_x^i g(i,k) - \int_0^1 \frac{\partial I_E(\vec{\phi}(s))}{\partial x} B_k(s) ds, \quad (3)$$

$$\frac{\partial \mathcal{F}}{\partial c_y^k} = 2\alpha \sum_{i=1}^{N} c_y^i f(i,k) + 2\beta \sum_{i=1}^{N} c_y^i g(i,k) - \int_0^1 \frac{\partial I_E(\vec{\phi}(s))}{\partial y} B_k(s) ds, \quad (4)$$

with

$$f(i,k) = \int_0^1 B_i'(s) B_k'(s) ds \text{ and } g(i,k) = \int_0^1 B_i''(s) B_k''(s) ds.$$

The first two terms of (3) and (4) depend on α and β and model the elasticity of the template, that is the ability of every point of the template to move in the same direction as its neighbors. The higher α and β are, the less the template can warp. In fact, for high values of these two weights, even a minimal deformation of $\vec{\theta}$ causes a consistent contribution for the value of $\mathcal{F}$. In our approach, the values of α and β are initially low and are increased during the deformation process. In this way, at the initial steps of the deformation process, a precise adaptation is achieved for those parts of the contour which are similar in both the template and the image, and an approximate adaptation is determined for those which are not equally represented in the two. Subsequently, increased values of α and β essentially compel the template to regularize its deformation without loss of match.

The last term in (3) and (4) models a strength that moves a point of the template in the same direction as the edge image gradient ∇I_E evaluated at that point. Since this strength is null if I_E is constant, the template, to correctly warp, needs to locate where ∇I_E is not null. If we want to follow a correct gradient descent path, we must use nonbinarized edge images, that is we must deform the template over a blurred edge image.

In Fig. 1, they are shown an original raw image stored in the database and different steps of the deformation process of a sketched template roughly representing a horse, over one of the two horse shapes in the blurred edge image. Graphs are also shown reporting the values of strain energy, bend energy, and match in the deformation process. They can be noticed the effects of increasing the values of α and β during the deformation process: The template starts to warp in a somehow irregular manner, in order to adjust itself to the horse boundary. Deformations which should determine a too large expense of strain and bend energy, such as the adaptation to the rider contour or to the horse legs are not exploited. In the final steps, higher values of α and β impose the template to regularize its deformation on the horse shape and, as a consequence, the values of strain and bend energy decrease.

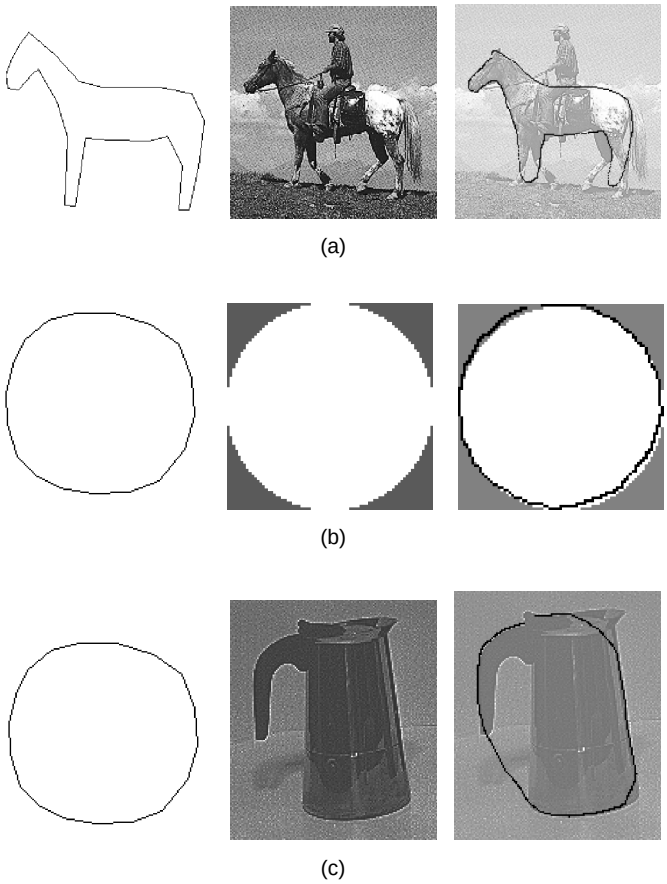
2.2 Template Matching

After a template reached convergence over an image shape, we need to measure how much the two are *similar*. Similarity is a fuzzy concept, and to measure it we need to take into account a number of things. A first thing to be taken into account is, of course, the degree of overlapping $\mathcal{M}$ between the deformed template and the gradient of the image. Another factor to be considered is how much the template had to warp to achieve that match in terms of strain energy $\mathcal{S}$ and bend energy $\mathcal{B}$.

Parameters $\mathcal{S}$, $\mathcal{B}$, and $\mathcal{M}$ alone are not enough to operate a good discrimination between different shapes. First of all, we have to consider that the values of $\mathcal{S}$ and $\mathcal{B}$ are somehow depending on the nature of the template shape. Fig. 2 shows three examples of template deformations; for each example, the template, the original image and the original image with the deformed template superimposed are shown. Table 1 reports the final values of $\mathcal{S}$, $\mathcal{B}$, and $\mathcal{M}$ for the three examples. It can be noticed that the deformation of the horse template over the horse shape image Fig. 2a is characterized by values of $\mathcal{S}$ and $\mathcal{B}$ which are fairly the same as those corresponding to the deformation of the circular template over the coffee-pot image Fig. 2c. While a good match of a complex shape can require high values for $\mathcal{S}$ and $\mathcal{B}$, a noncomplex shape can reach a good match with very low values of elastic deformation energy. A reliable solution is to consider a measure of the template *shape complexity*, in addition to the parameters of the deformation process. In our approach, the complexity of the template is measured as the number $\mathcal{N}$ of zeroes of the curvature function associated with its contour. When $\mathcal{N}$ is low, as in the case of the circular template, we expect to have low values of $\mathcal{S}$ and $\mathcal{B}$ for a correct deformation (Fig. 2b and Table 1b), while if $\mathcal{N}$ is high, as in the case of the horse template, we consider good values of deformation also values of $\mathcal{S}$ and $\mathcal{B}$ which, otherwise, should be discarded (Fig. 2a, and 2c and Table 1a and 1c).

Finally, we have to take into account that $\mathcal{S}$ and $\mathcal{B}$ give only a *quantitative* measure of the template deformation, while to estimate the similarity between the template and the image shape we must give also a *qualitative* measure of the deformation. This is imposed by considering the correlation C between the curvature function associated with the original template and that associated with the deformed one.

All these five parameters ($\mathcal{S}$, $\mathcal{B}$, $\mathcal{M}$, $\mathcal{N}$, C) are classified by a back-propagation neural network subject to appropriate training. For each input array, the neural classifier gives one output value ranging from 0 to 1, which represents the similarity between the shape in the image and the shape of the template.



(a) (b) (c)

Fig. 2. Examples of template elastic deformations.

TABLE 1
FINAL VALUES OF STRAIN, BEND, AND MATCH PARAMETERS FOR THE EXAMPLES SHOWN IN FIG. 2

Example	$\mathcal{S}$	$\mathcal{B}$	$\mathcal{M}$
(*a*)	3.72	1.00	0.84
(*b*)	0.14	0.27	1.00
(*c*)	3.70	0.95	0.85

2.3 Template Scaling and Rotation

When the user draws the sketch that will be used as a template, it is in an arbitrary scale and, in general, has an unknown relation with the scale of the objects it has to match. If we cover the image with a coordinate system (x, y), each interesting objects can be identified by its *minimum enclosing rectangles* (*MER*), with sides parallel to the coordinate axes, and lower left and upper right corners $\{(x_1, y_1), (x_2, y_2)\}$. We consider the *aspect ratio* of the rectangle:

$$\rho = \frac{y_2 - y_1}{x_2 - x_1}.$$

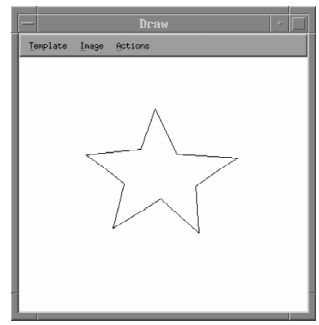
The sketch is similarly enclosed in its MER with extrema $\{(\tilde{x}_1, \tilde{y}_1), (\tilde{x}_2, \tilde{y}_2)\}$, which has an aspect ratio:

$$\tilde{\rho} = \frac{\tilde{y}_2 - \tilde{y}_1}{\tilde{x}_2 - \tilde{x}_1}.$$

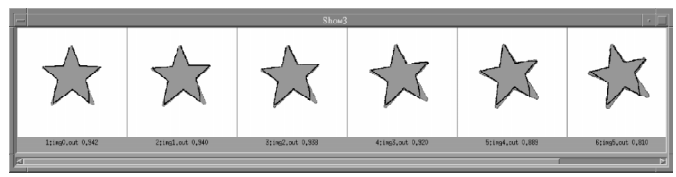
We can assume that the user, while making a query, draws an object approximately with the same aspect ratio of the object he wants to retrieve. For this reason, we can mark as nonmatched all those objects in the image whose aspect ratio is not such that:

$$\frac{1}{\kappa} \le \frac{\tilde{\rho}}{\rho} \le \kappa,$$

where κ is a fixed threshold. All the interesting rectangles that pass this sieve are candidates for matching. To speed up this checking, aspect ratios are organized into a binary tree index structure. Each node of the tree includes pointers to image rectangles with that aspect ratio. We have found, after tests discussed in Section 4, that the matching is improved if we normalize the sizes of both the template in the sketch and the shape in the image.



(a)



(b)

Fig. 3. (a) Sketched template of a five-tips star; (b) Matched images according to the elastic matching algorithm. The test set includes images representing a star rotated by three degree multiples with respect to the sketched template. Only matched images with a similarity rank greater than 0.7 are shown.

TABLE 2
VALUES OF $\mathcal{S}$, $\mathcal{B}$, $\mathcal{M}$, C AND SIMILARITY RANKS FOR THE STAR TEMPLATE OF FIG. 3 FOR TEST IMAGES ROTATED BY α_i DEGREES.

α_i	$\mathcal{S}$	$\mathcal{B}$	$\mathcal{M}$	C	Similarity rank
0	1.65	0.59	0.94	0.996	0.942
3	1.98	0.60	0.94	0.996	0.940
6	2.75	0.66	0.95	0.996	0.938
9	3.92	0.75	0.94	0.995	0.920
12	5.08	0.81	0.92	0.995	0.889
15	6.80	0.91	0.89	0.995	0.810
18	8.86	0.98	0.86	0.995	0.621
21	10.51	1.06	0.82	0.995	0.338
24	12.54	1.64	0.59	0.978	0.005
27	14.88	1.90	0.61	0.972	0.002

The elastic matching approach is not rotation invariant, since rotation contributes to the elastic deformation energy. However, small rotations (of the order of 12–15 degrees) usually can be coped with by the elastic matching algo-

rithm. As an example, in Fig. 3 the five-tip star template is made to warp over a set of I_i images, each one representing a star rotated by an angle α_i, of $i * 3$ degrees. From Table 2, it can be noticed that as the rotation angle increases, the deformation energies increase too, while the match decreases. For rotation angles greater than 18 degrees, the recognition ratio falls because of the high values of the elastic deformation energies and the low values of match.

3 Considering Spatial Relationships

We use spatial relationships between object MERs both as a mean to filter uninteresting database images and as a mean to make a more precise multi-object query. This is done by a slight modification of a method developed in [16].

Spatial relationships are defined by considering both relationships between projections of MER boundaries, and mutual objects orientations.

Projection of rectangles on the two coordinate axes determine begin (b_x, b_y) and end (e_x, e_y) boundaries of the object along x- and y-axis directions, respectively. Boundaries are sorted by introducing two precedence operators: "<" (*left-right*, *below-above*) and "=" (*same location as*). Therefore, all possible relationships between projections of two objects, o_i and o_j, are ranked in five categories C_{ij} (*disjoint*, *meet*, *contain*, *inside*, *partly_overlap*):

$$o_i \text{ disjoint } o_j \triangleq \left\{\begin{array}{l}\left[e_x(o_i) < b_x(o_j)\right] \vee \left[e_x(o_j) < b_x(o_i)\right] \vee \\ \left[e_y(o_i) < b_y(o_j)\right] \vee \left[e_y(o_j) < b_y(o_i)\right]\end{array}\right\}$$

$$o_i \text{ meet } o_j \triangleq \left\{\begin{array}{l}\left[e_x(o_i) = b_x(o_j)\right] \vee \left[b_x(o_i) = e_x(o_j)\right] \vee \\ \left[e_y(o_j) = b_y(o_i)\right] \vee \left[b_y(o_i) = e_y(o_j)\right]\end{array}\right\}$$
$$\wedge \neg\left(o_i \text{ disjoint } o_j\right)$$

$$o_i \text{ contain } o_j \triangleq \left\{\begin{array}{l}\left[e_x(o_j) \le e_x(o_i)\right] \wedge \left[b_x(o_i) \le b_x(o_j)\right] \wedge \\ \left[e_y(o_j) \le e_y(o_i)\right] \wedge \left[b_y(o_i) \le b_y(o_j)\right]\end{array}\right\}$$

$$o_i \text{ inside } o_j \triangleq \left\{\begin{array}{l}\left[e_x(o_i) \le e_x(o_j)\right] \wedge \left[b_x(o_j) \le b_x(o_i)\right] \wedge \\ \left[e_y(o_i) \le e_y(o_j)\right] \wedge \left[b_y(o_j) \le b_y(o_i)\right]\end{array}\right\}$$

$$o_i \text{ partly\_overlap } o_j \triangleq \neg\left(o_i \text{ disjoint } o_j\right) \wedge \neg\left(o_i \text{ meet } o_j\right) \wedge \neg\left(o_i \text{ contain } o_j\right) \wedge \neg\left(o_i \text{ inside } o_j\right).$$

Concerning orientations, we have to make image descriptions in terms of spatial relationships coherent with those operated by our visual perception. In practice, it is almost impossible for the user to reproduce object mutual orientations exactly as they are in the searched image. To cope with this inherent imprecision of the user query, given an object o_i its orientation with respect to o_j was evaluated by considering the position of the o_i centroid with respect to the o_j boundaries. If ($c_x(o_i)$, $c_y(o_i)$) are the projections of the centroid of o_i on the two axes, the orientation of o_i with respect to o_j is represented by the orientation vector, $\vec{O}_{ij} = \left(O_{ij}^1, O_{ij}^2, O_{ij}^3, O_{ij}^4\right)$, with:

$$O_{ij}^1 = \begin{cases}1 & if \quad c_x(o_i) \le b_x(o_j) \\ 0 & if \quad b_x(o_j) < c_x(o_i)\end{cases}$$

$$O_{ij}^2 = \begin{cases}1 & if \quad c_x(o_i) \le e_x(o_j) \\ 0 & if \quad e_x(o_j) < c_x(o_i)\end{cases}$$

$$O_{ij}^3 = \begin{cases}1 & if \quad c_y(o_i) \le b_y(o_j) \\ 0 & if \quad b_y(o_j) < c_y(o_i)\end{cases}$$

$$O_{ij}^4 = \begin{cases}1 & if \quad c_y(o_i) \le e_y(o_j) \\ 0 & if \quad e_y(o_j) < c_y(o_i).\end{cases}$$

Therefore, the spatial relationship $R_{ij} = R(o_i, o_j)$ of o_i with respect to o_j is represented by a symbolic 5-tuple:

$$R_{ij} = \left[C_{ij}, \vec{O}_{ij}\right].$$

3.1 Signature Files

If we have a picture I_1 (the image in the database) containing N_1 objects $o_1, \ldots, o_{N_1}$, its description in terms of spatial relationships is represented by the set of all the relationships R_{ij} such that $i, j \in \{1, \ldots, N_1\}$ and $i < j$. The cardinality of this set is $N_1(N_1 - 1)/2$. Given a second picture I_2 (the sketch) containing N_2 objects $o'_1, \ldots, o'_{N_2}$ (the templates), we have to filter out images that do not have N_2 objects in the same spatial relationships as the sketch, having no knowledge on the kind of objects represented in the two pictures. The problem is to find an injective function p such that:

$$p : \{1, \ldots, N_2\} \mapsto \{1, \ldots, N_1\},$$

and

$$R\left(o'_i, o'_j\right) = R\left(o_{p(i)}, o_{p(j)}\right) \quad \forall i, j \in \{1, \ldots, N_2\}. \tag{5}$$

As we have no a priori knowledge about the nature of the objects, the number of all of these p functions (5) is given by:

$$D_{N_1N_2} = \frac{N_1!}{(N_1 - N_2)!}.$$

To speed up the search, *binary codeword signature files* [18] are evaluated for the images in the database and for the user sketch. In this way, instead of testing all $D_{N_1N_2}$ combinations of spatial relationships, we test $D_{N_1N_2}$ signature files correspondences.

A signature file is composed of five fields (one field for each category C_{ij}), each field being itself composed of n_b bits. Each spatial relationship between two objects o_i, o_j is represented by one bit set in the signature file. The category

of the spatial relationship determines the field in the signature file, while the bit to be set in this field is determined with a hashing function $H = H\left(i, j, \vec{O}_{ij}\right) \mapsto \left\{0, \ldots, \left(n_b - 1\right)\right\}$, defined as follows:

$$H\left(i, j, \vec{O}_{ij}\right) = \left[n_0 * i + \left(j - i\right) + n_1 * u + n_2 * v\right] \bmod\left(n_b\right),$$

where n_0, n_1, and n_2 are integer parameters and u and v are defined as:

$$u = \begin{cases} 3 & if \quad \left(O_{ij}^1 = 1\right) \wedge \left(O_{ij}^2 = 1\right) \\ 2 & if \quad \left(O_{ij}^1 = 0\right) \wedge \left(O_{ij}^2 = 1\right) \\ 1 & if \quad \left(O_{ij}^1 = 0\right) \wedge \left(O_{ij}^2 = 0\right) \end{cases}$$

$$v = \begin{cases} 3 & if \quad \left(O_{ij}^3 = 1\right) \wedge \left(O_{ij}^4 = 1\right) \\ 2 & if \quad \left(O_{ij}^3 = 0\right) \wedge \left(O_{ij}^4 = 1\right) \\ 1 & if \quad \left(O_{ij}^3 = 0\right) \wedge \left(O_{ij}^4 = 0\right). \end{cases}$$

To improve the performance of the filter, the number of bits n_b and the values of the parameters n_0, n_1, and n_2 are chosen so as to reduce the probability of hash collisions. In the worst case condition, all the relationships between N objects in an image (which are $N(N-1)/2$) belong to the same category. Assuming N as the maximum number of objects for each database image, n_b can be chosen such that $n_b \geq N(N-1)/2$. Values of n_0, n_1 and n_2 are derived through statistical analysis to achieve uniform distribution of collisions and then minimize their probability of occurrence.

As an example, the spatial relationships between the objects in Fig. 4 are:

$$R_{12} = [\textit{disjoint}, 0, 0, 1, 1],$$
$$R_{13} = [\textit{disjoint}, 0, 0, 1, 1],$$
$$R_{23} = [\textit{disjoint}, 1, 1, 0, 0].$$

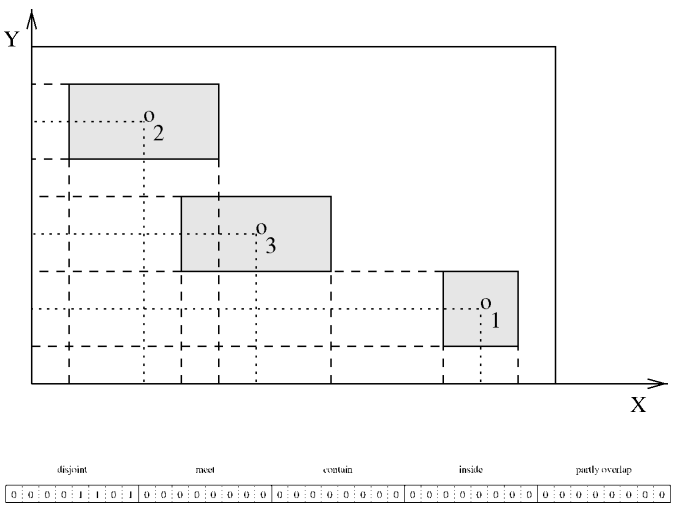


Fig. 4. Synthetic image with three sample objects and its signature file.

Assuming $n_b = 8$, $n_0 = 11$, $n_1 = 48$, and $n_2 = 16$, we have that $H(1, 2, 0, 0, 1, 1) = 4$, $H(1, 3, 0, 0, 1, 1) = 5$, and $H(2, 3, 1, 1, 0, 0) = 7$, respectively. Since the category of all these relationships is *disjoint*, the signature file of the sample image has the fourth, fifth, and seventh bit of the field *disjoint* set to one, as shown in Fig. 4.

Images whose signature files does not match the sketch signature file are filtered out. Matching of signature files does not ensure the exact correspondence between spatial relationships in the query and the image, since equal signatures may be due to hash collisions.

4 Experimental Results and Performance Analysis

Based on the techniques previously expounded, a prototype system has been developed for image retrieval by sketch, according to shape similarity. The following information structures are associated with each raw image of the database: One *image description file*, including a symbolic description of all spatial relationships between image objects; one *image signature file*, built according to the hashing discussed in Section 3.1 and used to filter out images with different spatial relationships; *edge images of rectangular areas* (one for each interesting shape) of 128×128 size, extracted through Canny edge detection.

In the very general case of sketches composed of multiple templates, a candidate image is retrieved if and only if:

1) it has two—or more—areas of interest in the same spatial relationships as the templates drawn on the screen;
2) the shapes contained in the areas of interest match the templates of the sketch within a certain degree.

The query sketch is represented as a polygonal and instantiated as a linear combination of B-spline functions with n_{knot} knots. Elastic matching is applied only to images that pass a composite filtering mechanism, based on spatial relationships matching (for multiple templates) and aspect ratio checking (for each template). For the experiments presented in this section, signature files were derived with $n_b = 8$ bits for each field, assuming the maximum number N of objects in the database images is four. Values of n_0, n_1, n_2 were chosen to make the probability of hash collisions equal to 0. A threshold $\kappa = 2$ has been used for the aspect ratio filtering. The average number of steps of the deformation process depends on how much the image and the sketch shapes are similar. After 20 steps, the match parameter $\mathcal{M}$ is compared with a fixed threshold. The deformation process is completed only for those shapes which pass this test.

The neural network that derives the similarity ratings, was a three layered 5–12–1 back propagation net. To determine similarity ratings on the basis of the shape complexity, the matching parameters and the amount of deformation, the net was trained using a set of sketched templates representing shapes with complexity ranging from zero to eight. Images of the training database included real objects shapes with different degrees of similarity with respect to the templates. Teaching outputs were adjusted manually, according to human similarity perception. Training and testing sets were both composed of about 350 examples.

Similarity ratings S_i evaluated for each of the N tem-

plates, are used to compute a compound similarity coefficient $R = \sum_{i=1}^{N} S_i$ for the whole image, which measures how much the user sketch globally matches the image. Retrieved images are sorted depending on the values of R and visualized on the computer screen.

4.1 Similarity Retrieval Effectiveness

Effectiveness of similarity retrieval according to the elastic matching approach was tested on a sample of users. In the test, we asked both the users and the system to rank a set of images with reference to their similarity to a few sketches, and we measured to what extent system answers conformed to those provided by the interviewed people.

We selected 22 sample images of bottles from a database of Morandi's paintings, and we drew three sketches roughly representing the shapes of three of these bottles. Images recall a familiar object to almost any people; nevertheless, bottle shapes are different though similar; in deciding their similarity ranking with reference to the sample sketches, the user must somehow capture both local and global properties of shape contours and make a form of adaptive matching, which generally cannot be transposed in algorithmic form.

We asked the people to assign for each sample image and each reference sketch, a value ranging from zero to one, representing the perceived similarity between the two. Each person had approximately 15 minutes to fill in the test forms. We collected answers from 42 people, all with university education. These were workers in the fine art field (15%), workers in other fields like engineering and literature education (10%), students in the fine art field (41%), students in engineering (34%). 75% of these people were male. Age was under 30 for the 83%, and under 40 for the 95%. Images used in the test set and sketches presented to the interviewed people are shown in Figs. 5a and 5b, respectively.

Answers confirmed that similarity is to a certain extent a subjective measure and that the range of the variability of human judgment must be considered different from shape to shape. For each of the 22 images, we derived three statistical functions $p_j(i)$, representing the ranking of the ith image with reference to the sketch j in the similarity list. For each function $p_j(i)$, a mean value $\bar{p}_j(i)$ and a standard deviation $\sigma_j(i)$ were derived, representing the average ranking of the ith image for a given sketch j in the similarity list, and a measure of the agreement about a ranking close to the $\bar{p}_j(i)$th rank, respectively. Finally, for each image i and rank k, a function $Q_j(i, k)$ with values in [0, 100] was considered, representing the percentage of people that ranked the ith image in the kth position with reference to the sketch j.

To measure the system performance and take into account the variability and shape dependency of the human judgment, we considered the percentage of people who ranks an image in the same position as the system, or in the very close neighbor. We considered for each bottle image i and reference sketch j, a window of width $\sigma_j(i)$ centered in the similarity rank $P_j(i)$ given by the elastic matching algorithm. The measure of the distance between the system and human similarity ranking for a reference sketch j and a test image i, was represented by the sum of the percentage of people who ranked the ith image in a position between $P_j(i) - \left\lceil \frac{\sigma_j(i)}{2} \right\rceil$ and $P_j(i) + \left\lceil \frac{\sigma_j(i)}{2} \right\rceil$. Therefore, we considered the function $S_j(i)$ defined as:

$$S_j(i) = \sum_{k=P_j(i)-\left\lceil \frac{\sigma_j(i)}{2} \right\rceil}^{k=P_j(i)+\left\lceil \frac{\sigma_j(i)}{2} \right\rceil} Q_j(i,k).$$

In Fig. 6, plots of $S_j(i)$ as a function of rankings $P_j(i)$ are presented. They show the agreement between the interviewed people and the system in ranking the ith image in the $P_j(i)$ position, for each of the three sample sketches. Only ranks from one to six are shown since they represent the agreement on the most similar bottles. As it can be noticed, there is a very large agreement between the interviewed people and the system in the assignment of the similarity ranks. With reference to sketch 3, since only a few thin bottles were present in the test set, after these have been correctly ranked in the highest positions, the similarity ranking agreement between humans and the system decays due to the practical impossibility to assign precise ranks for the other images. In no cases was the agreement below the 50%. In Figs. 7, 8, and 9, query results are shown for each of the three sketches.

An example of sketch-based retrieval of Italian sacred pictures of the 12*th*-13*th* century is shown in Fig. 10. For this test we used a database of 100 samples, including the 22 Morandi images, 10 sacred pictures, and sample pictures of diverse objects with dissimilar shapes. In the query, we drew a rough sketch of a crucified Christ with a special posture of head, arms and legs. Retrieved images with the highest similarity ranks are shown, where approximately the same postures are present.

In Fig. 11, a query with multiple templates is shown where the user asks for images with a crucified Christ hanging not too high up over a person standing at a close distance on his right side. Retrieved images are shown which include figures having mutual relationships and shapes like those in the sketch.

With reference to this example, 95% of database images were filtered out through signature file comparison.

4.2 Comparative Performance Analysis

In this section, we analyze the performance of the elastic template matching approach, ETM in the following, compared with QBIC [10], [20], and QVE [11] systems. We compared the three techniques with respect to computational complexity, similarity retrieval effectiveness, and robustness to shape variability.

4.2.1 Computational Complexity

Computational complexity is analyzed for queries with a single sketched template. Query processing steps and filtering methods adopted by the QBIC and QVE systems are briefly reviewed first.

In the QBIC system, database images are subject to edge extraction (automatic tools are provided for a restricted

class of images; manual identification of contours is otherwise required) and reduced to 64×64 binary edge images. Both for images in the database and the user sketch, a set of 22 features are computed, namely area, circularity, eccentricity, major axis orientation, and a set of its algebraic moment invariants. Moment invariants are derived as a function of central moments up to degree 8 [10]. A low dimensional feature space is obtained by applying the Karhunen-Loéve transform to the 22-dimensional feature vectors. For a single template, filtering of nonrelevant images is obtained through a search in an *R*–tree* index structure. Similarity is evaluated as a weighted Euclidean distance in the feature space.

In the QVE system, database images are subject to edge extraction, thinning and binarization, and are stored as 64×64 binary edge images. The user sketch is scaled up to a 64×64 binary mask and partitioned into 64 block of 8×8 size. No filtering mechanism is proposed. Matching is performed by evaluating a logical correlation between each block of the sketch and a corresponding search area in the image, with horizontal and vertical shifts (from one to four pixels) to model a limited warping of the sketch.

Table 3 compares the computational complexity of the three systems. Complexity of both QVE and QBIC systems depends on the size of the binary edge images. In QBIC, computation of features is performed only once for each query template and the query is resolved as a sublinear, very selective search through the index structure. QVE requires to apply the matching procedure to each database image. The complexity of ETM query processing for each image depends on the number of knots n_{knot} of the spline used to model the template deformation, and on the number of iterations n_{iter} needed to complete the deformation process. Experiments carried out have shown that 20 knots suffice to model the deformation of quite complex templates such as that reported in Fig. 1. Query processing requires a sublinear filtering of database images, and elastic matching for all the images that pass the filter.

TABLE 3
COMPLEXITY OF THE THREE SYSTEMS

System	Computational Complexity	Typical n. Operat.	$O(\cdot)$
ETM [(1)]	$[400 n_{knots} \cdot n_{iter}] \otimes$	240000	$O(n_{knots} \cdot n_{iter})$
QBIC [(2)]	$[200N^2+3000] \otimes$	822200	$O(N^2)$
QVE [(3)]	$[N^3+N^2(1-2\sqrt{N})] \oplus$	200704	$O(N^3)$

Only the most significant operations are considered.

[(1)] Average number of operations required for each image shape that passes the spatial and aspect ratio filters. Typical number of operations is evaluated for $n_{knots} = 20$ *and* $\overline{n}_{iter} = 30$ *(assuming that 20% of shapes require 70 deformation steps and remaining ones 20).* *[(2)] Number of operations required to compute the feature values for the sketch. Typical number of operations is evaluated for* $N = 64$*. Retrieval requires evaluation of a weighted Euclidean distance between the feature values of the sketch and those in the nodes of the R*–tree index.* *[(3)] Number of operations required for each image shape in the database. Typical number of operations is evaluated for* $N = 64$*.*

4.2.2 Similarity Retrieval Effectiveness

Comparison of similarity retrieval effectiveness was analyzed under the test discussed in Section 4. We used QBIC Version 1.1 under OS/2 WARP Connect Version 3; QVE algorithms were replicated according to the specifications given in [11]. Fig. 6 compares plots of the agreement between the human and system similarity rankings for the three sketches, by the three systems.

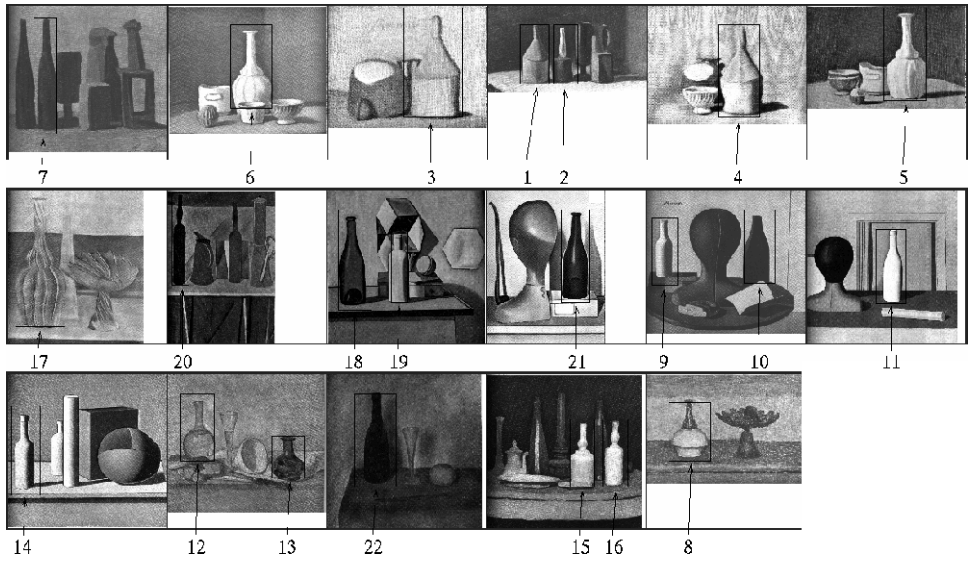
ETM shows a good ranking agreement for all the three sketches, as was discussed extensively in Section 4.1. Both QBIC and QVE systems exhibit substantially different ranking agreements depending on the silhouette of the template presented to the system: They manifest a considerable loss of performance in the similarity matching of the third template. Effective similarity retrieval is provided particularly according to global shape properties (for example roundness or squareness of the bottle body). But, they both provide only a very limited ability to evaluate similarity according to local properties. In the third query, although several bottles retrieved are long and narrow, bottles with a very different neck juncture from the sketch are ranked as more similar than those with the same one. Similarity rankings $P_j(i)$ derived by the three systems are presented in Table 4.

TABLE 4
RANKS OF MORANDI'S BOTTLE IMAGES AS THEY ARE DERIVED BY THE ELASTIC TEMPLATE MATCHING (ETM), QBIC, AND QVE SYSTEMS, FOR THE THREE SAMPLE SKETCHED TEMPLATES

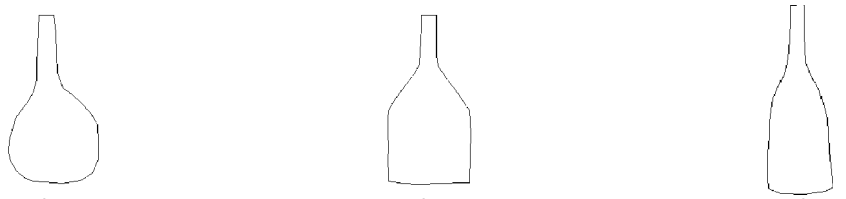
	ETM			*QBIC*			*QVE*		
bottle n.	T_1	T_2	T_3	T_1	T_2	T_3	T_1	T_2	T_3
1	7	2	15	5	1	19	9	3	1
2	10	8	10	8	10	6	5	13	11
3	5	3	13	6	2	16	10	1	3
4	4	1	14	2	4	14	13	2	2
5	1	4	16	7	6	10	3	8	7
6	2	5	18	3	3	12	6	7	8
7	22	22	12	20	20	15	19	15	14
8	6	10	22	4	7	21	1	14	12
9	17	18	5	18	15	11	17	22	19
10	19	9	9	11	12	2	22	19	22
11	15	17	6	21	21	13	16	5	9
12	3	6	20	1	5	20	4	6	13
13	8	13	21	12	9	22	7	18	15
14	16	16	4	17	16	7	21	17	18
15	11	7	11	9	8	8	8	4	16
16	18	15	8	16	19	9	20	12	20
17	9	20	19	10	11	1	2	10	4
18	12	11	3	13	13	4	15	16	5
19	21	19	7	19	18	17	14	21	21
20	20	21	17	22	22	18	18	20	17
21	13	12	2	14	14	5	12	11	10
22	14	14	1	15	17	3	11	9	6

4.2.3 Robustness to Shape Variability

Perceiving as similar objects that have undergone a great variation in shape, remaining somehow themselves, is a typical ability of humans. Retrieval robustness with respect to a sketched query was intended as the ability of the system to retrieve objects of the same type as the sketch, irrespective of shape variations of database instances. For this test, we used the three sample sketches as in Section 4 and the test database of 100 samples, including the 22 Morandi's images and sample pictures of diverse objects with dissimilar shapes.



(a)



(b)

Fig. 5. a. Test set of 22 bottle images from the Morandi's catalogue; b. sketched templates used in the test.

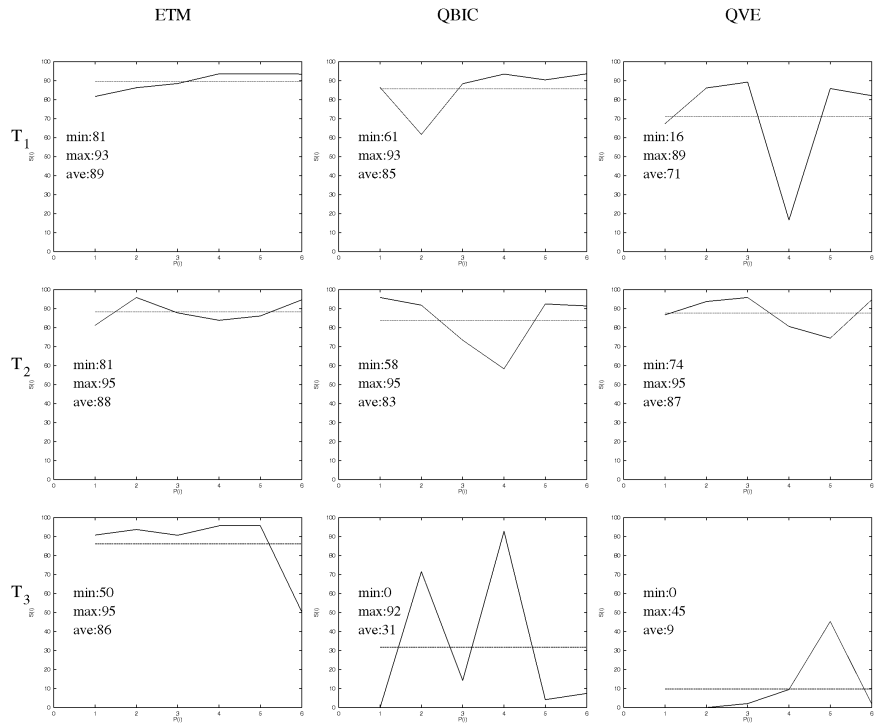


Fig. 6. Comparative results of the ETM, QBIC, and QVE retrieval effectiveness. Plots report the values of the agreement between system and human measure of similarity (continuous line), and values of the average agreement (dotted line), for the first six ranked bottles.`

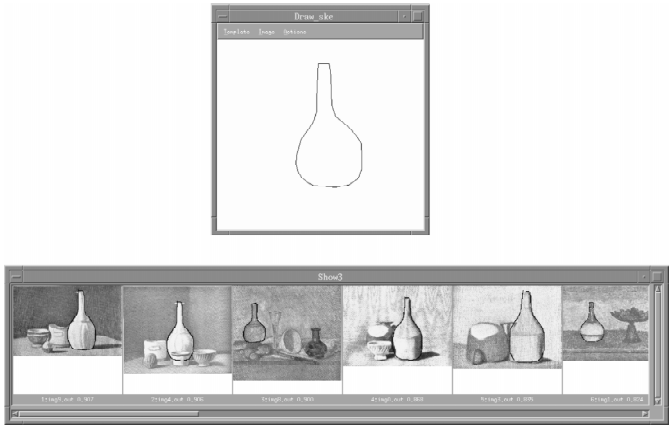


Fig. 7. Matched images with highest similarity ranks for the sketched template 1. Only the six highest ranked images are shown.

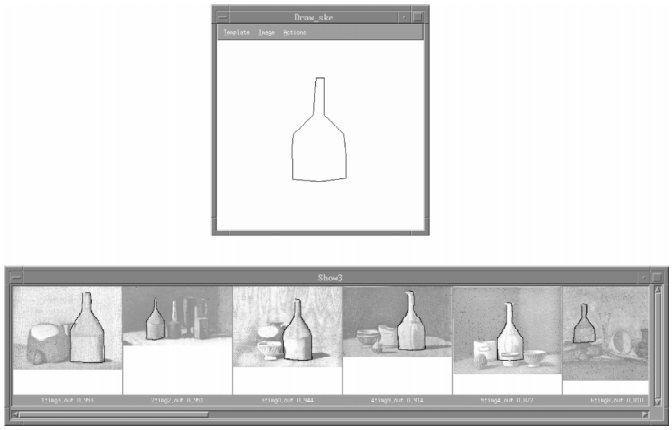


Fig. 8. Matched images with highest similarity ranks for the sketched template 2. Only the six highest ranked images are shown.

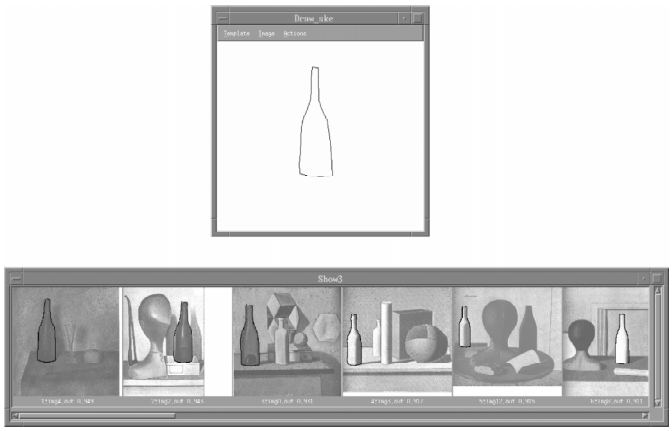


Fig. 9. Matched images with highest similarity ranks for the sketched template 3. Only the six highest ranked images are shown.

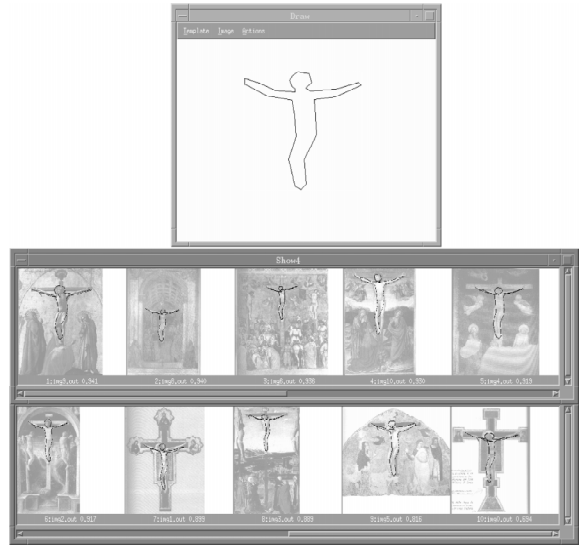


Fig. 10. Retrieval results for the sketch representing a crucified Christ.

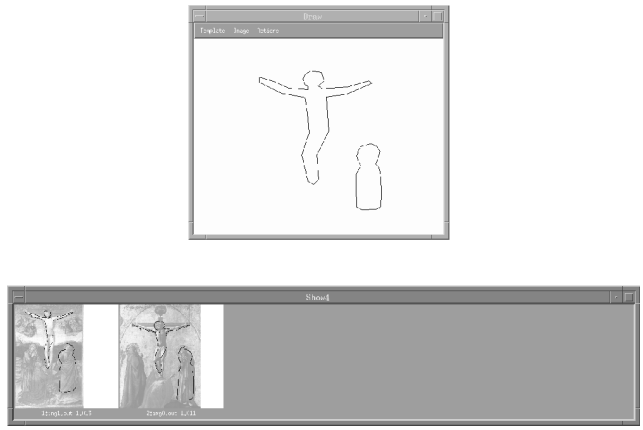


Fig. 11. Retrieval results for the sketch representing a crucified Christ with a close standing person.

TABLE 5
RETRIEVED ITEMS OF THE SAME TYPE AS THE SKETCH IN THE FIRST *n* POSITIONS FOR THE THREE SKETCHED TEMPLATES

	Ranking Interval System	1–5	1–10	1–22	1–30	1–40
T_1	*ETM*	5	10	21	22	22
	QBIC	4	7	11	12	14
	QVE	5	10	20	21	21

	Ranking Interval System	1–5	1–10	1–22	1–30	1–40
T_2	*ETM*	5	10	22	22	22
	QBIC	4	7	11	12	15
	QVE	5	10	20	21	22

	Ranking Interval System	1–5	1–10	1–22	1–30	1–40
T_3	*ETM*	5	10	22	22	22
	QBIC	4	9	13	16	19
	QVE	5	10	20	21	21

The total number of bottle-like shapes in the database is 22.

Table 5 reports results provided by the three systems. The number of retrieved items of the same type as the sketch are reported for different ranking intervals. Both ETM and QVE exhibit good tolerance to shape variations. They both retrieve in the first ranked positions all or most of the objects with the same structure as the sketch, although with different shapes, in all the three cases. QBIC exhibits a lower performance. In all the three cases, a lot of misspelled items are retrieved. The system fails to rank all the 22 bottles in the first 40 positions of the retrieval list. Several retrieved objects of different types have somehow similar global shape properties.

5 Conclusions

In this paper, we presented a technique for image retrieval by shape similarity which is based on elastic matching of sketched templates over the shapes in the images.

The sketch-based retrieval by-content system is intended as a special part of a multimedia system, especially oriented to support fine art specialists and researchers to discover shape similarities or, more generally, relationships between different paintings which are not explicitly expressed or known. Retrieval by shape similarity and relative positions supports the critic in the analysis of the artists' periods, as well as of the influences and commonalities between different paintings. We are presently working on the development of additional facilities to enable queries by shape similarity based only on parts of the object boundary and to support effective indexing of the pictures in the database according to common shape attributes.

References

[1] P.M. Anselone and P.J. Laurent,"A General Method for the Construction of Interpolating or Smoothing Spline-Functions," *Numerische Matematik*, vol. 12, pp. 66-82, 1968.

[2] E. Binaghi, I. Gagliardi, and R. Schettini, "Indexing and Fuzzy Logic-Based Retrieval of Color Images," *IFIP Trans. A-7, Visual Database Systems II*. Elsevier Pub., 1992.

[3] S.K. Chang, Q.Y. Shi, and C.W. Yan, "Iconic Indexing by 2D Strings," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 9, no. 3, Mar. 1987.

[4] S.K. Chang and E. Jungert, "Pictorial Data Management Based upon the Theory of Symbolic Projections," *Journal of Visual Languages and Computing*, vol. 2, no. 2, June 1991.

[5] S.K. Chang, C.W. Yan, D.C. Dimitroff, and T. Arndt, "An Intelligent Image Database System," *IEEE Trans. Software Eng.*, vol. 14, no. 5, May 1988.

[6] A. Del Bimbo, E. Vicario, and D. Zingoni, "A Spatial Logic for Symbolic Description of Image Contents," *J. Visual Languages and Computing*, vol. 5, no. 4, 1994.

[7] A. Del Bimbo, M. Campanai, and P. Nesi, "A Three-Dimensional Iconic Environment for Image Database Querying," *IEEE Trans. Software Eng.*, vol. 19, no. 10, Oct. 1993.

[8] A. Del Bimbo, P. Pala, and S. Santini, "Visual Image Retrieval by Elastic Deformation of Object Shapes," *Proc. IEEE VL'94, Int'l Symp. Visual Languages*, St. Louis, Mo, Oct. 1994.

[9] R. Durbin and D. Willshaw, "An Analogue Approach to the Traveling Salesman Problem Using an Elastic Net Method," *Nature*, vol. 326, Apr. 1987.

[10] C. Faloutsos, M. Flickner, W. Niblack, D. Petkovic, W. Equitz, and R. Barber, "Efficient and Effective Querying by Image Content," Research Report 9453, IBM Research Div., Almaden Research Center, Aug. 1993.

[11] K. Hirata and T. Kato, "Query by Visual Example, Content-Based Image Retrieval," *Advances in Database Technology—EDBT'92*, A.Pirotte, C.Delobel, and G.Gottlob, eds., Lecture Notes on Computer Science, vol. 580.

[12] W.I. Grosky, R. Mehrotra, "Index-Based Object Recognition in Pictorial Data Management," *Computer Vision Graphics and Image Processing* , vol. 52, no. 3, pp. 416-436, 1990.

[13] E. Jungert, "The Observer's Point of View, an Extension of Symbolic Projections," *Proc. Int'l Conf. Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, Pisa, Sept. 1992, *Lecture Notes in Computer Science*. Springer-Verlag, 1992.

[14] C. Jacobs, A. Finkelstein, and D.H. Salesin, "Fast Multiresolution Image Querying," *Proc. SIGGRAPH '95*, Los Angeles, Calif., Aug. 1995.

[15] A. Kankanhalli, H.J. Zhangh, and C.Y. Low. "Using Texture for Image Retrieval," *IEEE Proc. Int'l Conf. Automation Robotics and Computer Vision*, 1994.

[16] S.Y. Lee and F.J. Hsu, "Spatial Reasoning and Similarity Retrieval of Images Using 2D-C String Knowledge Representation," *Pattern Recognition*, vol. 25, no. 3, 1992.

[17] S. Lee, M.K. Shan, and W.P. Yang, "Similarity Retrieval of Iconic Image Database," *Pattern Recognition*, vol. 22, no. 6, 1989.

[18] S.Y. Lee, M.C. Yang, and J.W. Chen. "Signature File as a Spatial Filter for Iconic Image Database," *J. Visual Languages and Computing*, vol. 3, pp. 373–397, 1992.

[19] R. Mehrotra and J. Gary "Similar Shape Retrieval in Shape Data Management," *Computer*, vol. 28, no. 9, pp. 57-62, Sept. 1995.

[20] W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubino, "The QBIC Project, Querying Images by Content Using Color, Texture and Shape," *Storage and Retrieval for Images and Video Databases*, pp. 173-181, SPIE, Feb. 1993.

[21] M. Ohlssono, "Extensions and Explorations of the Elastic Arms Algorithm," Technical Report LU TP 92-28, Dept. of Theoretical Physics, Univ. of Lund, Sölvegatan 14A S-22362 Lund, Sweden, Dec. 1992.

[22] R.W. Picard, "The Society of Models for Video and Image Libraries," Technical Report 360, MIT Media Lab. Perceptual Computing Section, 1995.

[23] R.W. Picard and F. Liu, "A New World Ordering for Image Similarity," *Proc. IEEE Conf. Acoustic, Speech, and Signal Processing*, Adelaide, Australia, Apr. 1994.

[24] M.J. Swain and D.H. Ballard, "Color Indexing," *Int'l J. Computer Vision*, vol. 7, no. 1, 1991.

[25] S.L. Tanimoto, "An Iconic/Symbolic Data Structuring Scheme," *Pattern Recognition and Artificial Intelligence*, C.H. Chen, ed. New York: Academic, 1976.

[26] A.N. Tihonov, "Regularization +of Incorrectly Posed Problems," *Soviet Math. Doklady*, vol. 4, pp. 1,624–1,627, 1963.

[27] A.L. Youille, K. Honda, and C. Petersono, "Particle Tracking by Deformable Templates," *Proc. Int'l Joint Conf. Neural Networks*, 1991.

Alberto Del Bimbo received his doctoral degree in electronic engineering from the Università di Firenze, Italy, in 1977. He was with IBM Italia from 1978 to 1988. He is currently a full professor of computer systems at the Università di Firenze. Dr. Del Bimbo is a member of the IEEE and of the International Association for Pattern Recognition. He is a board member of the IAPR Technical Committee n.8 (Industrial Applications), and president of the IAPR Italian chapter. He presently serves as associate editor of *Pattern Recognition Journal* and the *Journal of Visual Languages and Computing*. His research interests and activities are in the fields of image analysis, image databases, and visual languages.

Pietro Pala received the MS degree in electronic engineering from the Università di Firenze, Italy in 1994. He is presently a PhD student at the Dipartimento di Sistemi e Informatica of the Università di Firenze. His current research interests include pattern recognition, image databases, neural networks, and related applications. He has been a member of the IEEE since 1993.