Size Functions for Image Retrieval: A Demonstrator on Randomly Generated Curves

A. Brucale, M. d'Amico, M. Ferri, L. Gualandri, A. Lovato

Dip. di Matematica, Università,
Piazza di Porta S. Donato, 5
I-40126 Bologna Italy
`ferri@dm.unibo.it`

Abstract. Size functions, a class of topological-geometrical shape descriptors, are applied to the search of image datasets by a hand-drawn input: This is the core of a demonstrator accessible through the Internet. Two datasets are provided, of about 700 curves each; both are unstructured sets of randomly generated curves. One consists of piecewise smooth curves, the other of polygonals. The curve generator has been explicitly designed so that the curves have no semantic content; they don't have any kind of indexing or textual caption either. The characteristics of the demonstrator and some experimental results are presented.

1 Introduction

We present a demonstrator of an image retrieval engine, based on size functions, working on two unstructured datasets of randomly generated curves. The datasets consist of 700 curves each; curves are closed unions of smooth (but not rectilinear) arcs in one case, closed polygonals in the other. The query is performed by drawing an example by hand. The demonstrator is reachable through the Internet (http://vis.dm.unibo.it/EVAM/).

Size functions are a relatively new class of shape descriptors, which can be modularly adapted to specific recognition or comparison problems. From the mathematical viewpoint, they are based on geometric–topological theory of critical points, as will be specified in the next Section. So long, they have been applied to several classification problems, mostly on images of natural origin.

In the terminology of [14], our system belongs to the categories of "content based", "search by association", "narrow domain", "approximate query by image example", but the difference with the systems reported in that survey are: Semantic interpretation (as, e.g., in [12]) is impossible or at least highly subjective (Rorschach-like); no colors or textures are present to direct selection (as, e.g., in [5]); also salient geometric features (as used, e.g., in [13]) would be of little of no use while comparing a hand-drawn sketch with a piecewise smooth curve or a polygonal. No indexing or other structure (as, e.g., in [1]) is present in the image domain (it is a dataset, not a database).

The demonstrator is far from being a general purpose engine: Image processing problems have been by-passed, so that the user can concentrate on our main issue, i.e. the value of size functions in finding similarities and differences of "pure" shapes. We are well aware that a commercial (or at least a useful) product should integrate size functions with other techniques, but this is far from our present scope.

2 The definition of Size Function

Size functions are a simple, but effective tool for shape comparison. Their application is particularly useful when no standard, geometric templates are available. Examples of applications are tree-leaves, hand-drawn sketches, monograms, hand-written characters, white blood cells and the sign alphabet.

Let us recall the definition of a size function. Consider a continuous real-valued function $\varphi : M \to \mathbb{R}$, defined on a subset M of a Euclidean space. The *size function* of the pair (M, φ) is a function $\ell_{(M,\varphi)} : \mathbb{R}^2 \to \mathbb{N} \cup \{\infty\}$. For each pair $(x, y) \in \mathbb{R}^2$, consider the set $M_x = \{P \in M : \varphi(P) \leq x\}$. Two points in M_y are then considered to be equivalent if they either coincide or can be connected by a continuous path in M_y. The value $\ell_{(M,\varphi)}(x, y)$ is defined to be the number of the equivalence classes obtained by quotienting M_x with respect to the previous equivalence relation in M_y.

Figure 1 shows a simple example of size function. In this case the topological space M is a curve, while the measuring function φ is the distance from the centre of mass C.

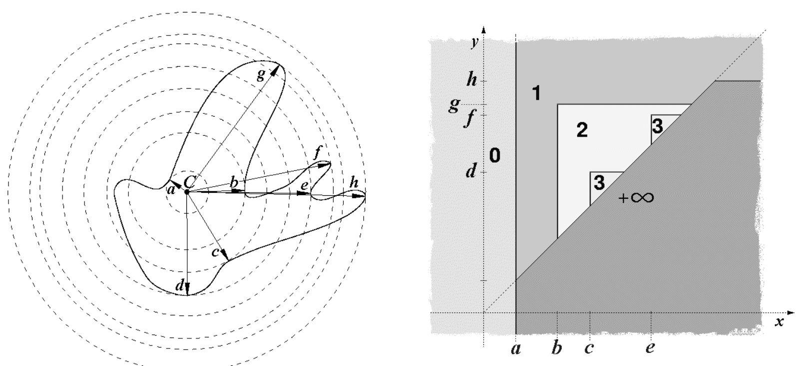

Fig. 1. A curve and its size function with respect to the distance from the centre of mass.

The use of size functions in shape description and comparison is motivated by the following properties:

Invariance. Size functions inherit the same invariance of the corresponding measuring functions.

Modularity. Changing the invariance group simply requires changing the measuring functions.

Multilevel analysis. Details are described close to the diagonal $\Delta = \{(x, y) \mid x = y\}$, while more general aspects of shape appear far from Δ.

Resistance to noise. Information is distributed all over the real plane, so that size functions can be used in presence of noise and occlusions.

Fast computation. Size functions are easily computable.

Standardization. The problem of comparing shapes is changed into a simple comparison of functions.

3 Representation and Comparison of Size Functions

As Figure 1 shows, size functions have a typical structure. In the halfplane $\{x < y\}$ they are linear combination of characteristic functions of triangular regions. In other words, if ℓ is a size function we have that $\ell(x, y) = \sum_{i \in I} \chi_i(x, y)$ (for $x < y$), where I is countable set and each χ_i is the characteristic function of a triangle T_i, having vertices $(a_i, a_i), (a_i, b_i), (b_i, b_i)$ (i.e., $\chi_i(x, y) = 1$ if $a_i \leq x \leq y \leq b_i$ and 0 otherwise). That means that each size function can be described by a formal linear combination $\sum_{h \in I} m_h C_h$ of points $C_h = (a_h, b_h)$ (the right angle vertex of T_h) with integer multiplicities m_h. If $b_h < \infty$ we call C_h a *cornerpoint*, otherwise we say that C_h represents a *cornerline* (in this case the corresponding triangle T_h degenerates to an "infinite" triangle). Each distance between formal series naturally produces a distance between size functions. A formal and detailed treatment of this subject can be found in [6].

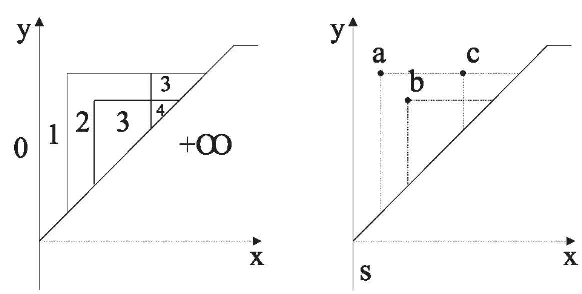

Fig. 2. A size function on the left and the corresponding cornerpoints a, b, c and cornerline s on the right. The resulting formal series is $s + a + b + c$

Of the many available distances between formal series, the one we use in this paper is the matching distance (similar to the "bottleneck matching" of [15], but performed on the size functions, instead of the images). In plain words, if we wish to compute the distance between two formal series $\sum_{i\in I} m_i C_i$, $\sum_{j\in J} m'_j C'_j$ we consider any possible matching between the C_i's and the C'_j's, considering the multiplicities. Each of these matchings is weighted by a deformation cost (in this work: Euclidean displacement). The wanted distance is the infimum in the set of all possible costs of matchings. We also allow the "destruction" of cornerpoints by taking them onto the diagonal $\Delta = \{(x, y) | x = y\}$. The inverse process of cornerpoint creation is allowed, too.

The computation of the matching distance is performed by means of a Max Flow algorithm. Only the first ten cornerpoints are considered, with respect to the decreasing distance from the diagonal Δ. A similar version of the matching distance (more along the lines of "elastic matching" [2]) can be found in [7].

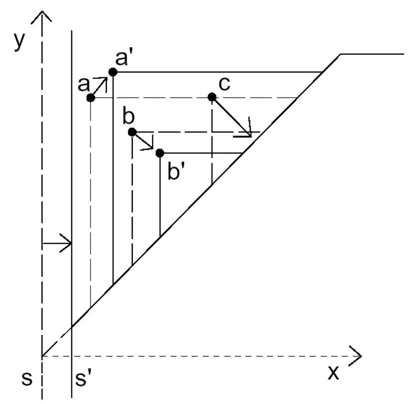

Fig. 3. A possible way to deform a size function, represented by dashed lines, into another one, represented by continuous lines. The cornerpoint c is destroyed.

4 Curve Generation

Two sets of 700 curves have been built for the demonstrator. The curves of the first set have a finite (possibly empty) set of singular points, out of which they are smooth and non-rectilinear (see Figure 4 for the first 70 of them). The second set consists of closed polygonals (see Figure 5 for the first 70 of them).

The piecewise smooth curves are all generated by random parameter variations of the formula

$$\begin{cases} x(t) = \lambda\rho(t)\cos\theta(t) \\ y(t) = \frac{1}{\lambda}\rho(t)\sin\theta(t) \end{cases}$$

with

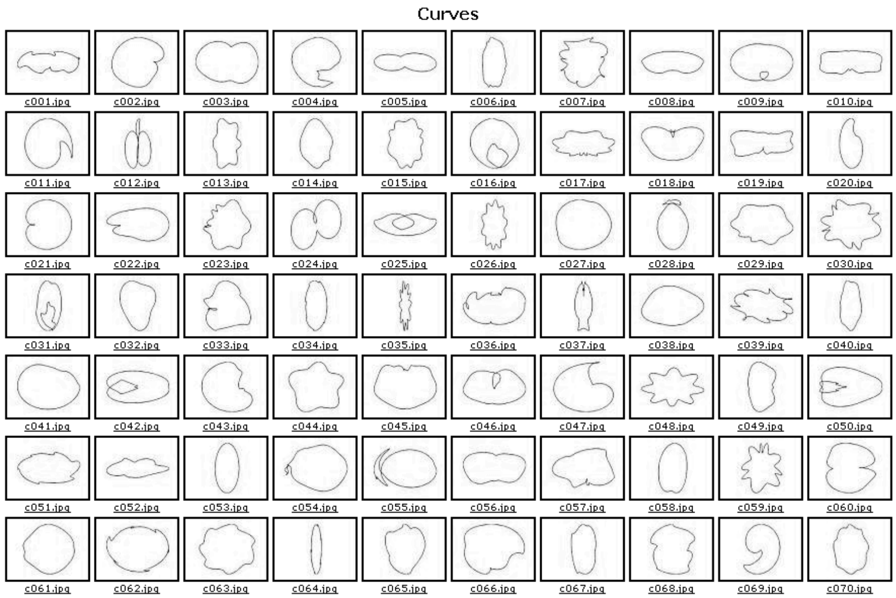

Fig. 4. A sample of our dataset of randomly generated smooth curves.

$$\begin{aligned}
\theta(t) &= s\,(2\pi t + \alpha_1 \sin(\ell_1 2\pi t) + \beta_1 \cos(\ell_2 2\pi t)) + \frac{k\pi}{2} \\
\rho(t) &= r_0 + \alpha_2 \sin(h_1 2\pi t) + \beta_2 \cos(h_2 2\pi t)
\end{aligned}$$

where λ is randomised with normal distribution, with mean $m = 1$ and variance $\sigma^2 = 0.3$, and r_0 is also randomised with normal distribution, $m = 3$, $\sigma^2 = 1$. The values of the other parameters are uniformly distributed in their sets:

$$\begin{aligned}
s &\in \{-1, 1\} \\
k &\in \{0, 1, 2, 3\} \\
\alpha_1 &\in [0, 2] \\
\ell_1 &\in \{0, 1\} \\
\beta_1 &\in [0, 0.3] \\
\ell_2 &\in \{0, 1, 2, 3, 4, 5, 6, 7\} \\
\alpha_2 &\in [0, 3] \\
h_1 &\in \{1, 2, 3\} \\
\beta_2 &\in [0, 0.6] \\
h_2 &\in \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}.
\end{aligned}$$

Polygonals are generated by connecting random points in the prescribed frame with segments.

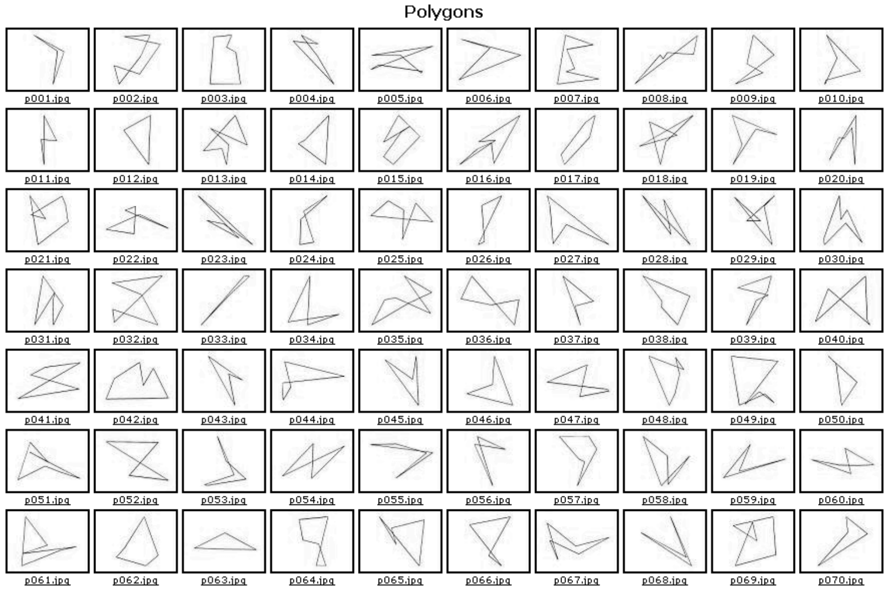

Fig. 5. A sample of our dataset of randomly generated polygonal curves.

5 Experimental Results

The measuring functions used in the demonstrator are the distances from 16 fixed points forming a regular grid around the center of mass of the image; they are disposed in four horizontal rows, each of length equal to 2.8 times the average distance of the image pixels from the center of mass. The invariance granted by this type of measuring functions is under translations and scale changes. We remind that invariance under other transformation groups (rotations, affine transformations etc.) could be easily imposed by a different choice of measuring functions.

Two types of queries were performed: by submitting an image extracted from the dataset, and by submitting a hand–drawn curve. The latter is of course the more interesting experiment, and it is the one on which we are reporting. This was done separately for the smooth curves and for the polygonals.

It should be noted that it is not possible to establish a hit/miss ratio or an analogous mark, since shape similarity is subjective. Moreover, the curves of our dataset do not represent any concrete object, so they cannot be classified even according to a semantic criterion. Next, we present some of the queries (Figures 6 and 7), with the first five output curves.

Some remarks about the depicted results:

- Some queries were aimed to specific curves present in the datasets, and some were totally generic.

- Given an input drawing, it is unlikely that there are several resembling shapes in the dataset, so even among the first five output curves there are some scarcely related.
- Finally, it should be noted that in most cases even the most resembling output curves are not superimposable to the input example. Moreover, even salient points or any commonly used geometric features would not have helped to retrieve those curves (actually, they might even be misleading!). The reason why the system succeeds in retrieving them all the same, resides in the highly qualitative kind of shape description and comparison due to size functions.

The demonstrator runs on a Athlon 900 MHz based PC, under Linux OS. Response time for a query is, for the moment, between 40 and 60 seconds.

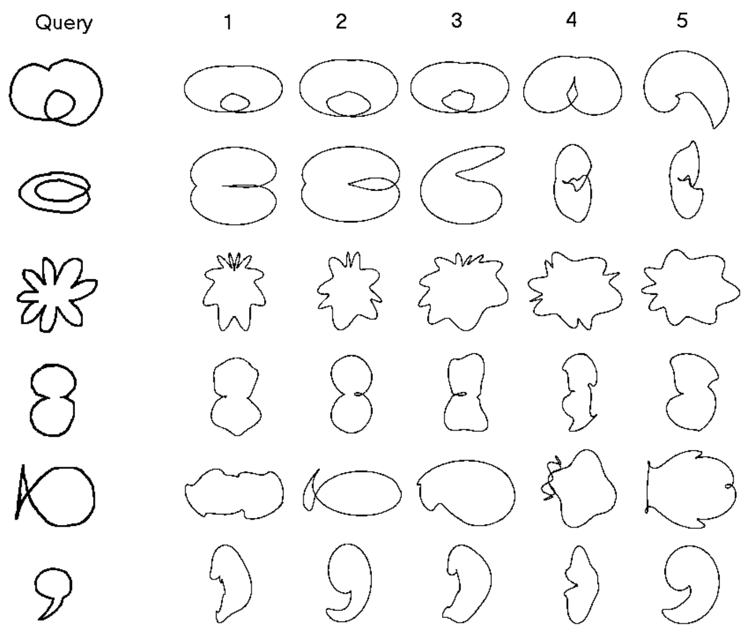

Fig. 6. The results of some sample queries in our dataset of smooth curves.

6 Comparisons

Of course, there are several shape descriptors and shape matching methods (see, e.g., [15]). As Carlsson [3] points out, there are two major categories: recognition by components (requiring some *a priori* knowledge) and recognition on similarity of views (based on geometrical features). Our system is rather in the second class, although in a generalised sense. In fact, the similarities which are caught by size functions may not be so apparent: they depend on the choice of the measuring

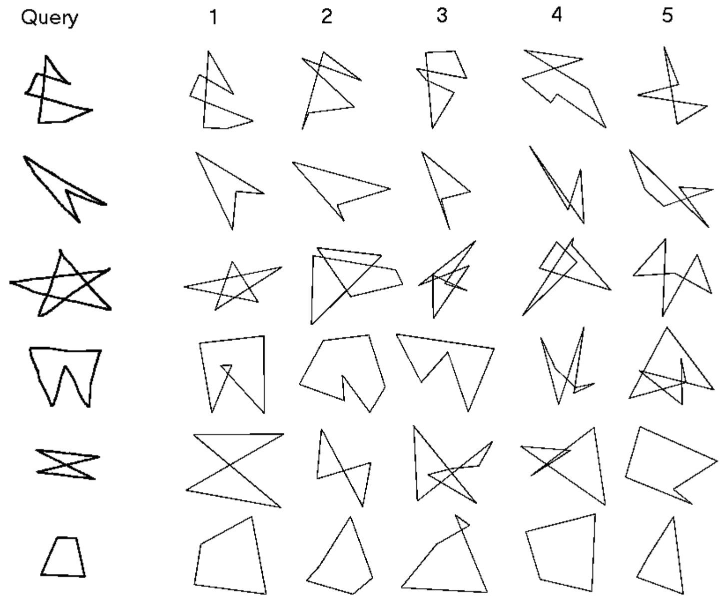

Fig. 7. The results of some sample queries in our dataset of polygonal curves.

function. That no semantics is involved, is clear in the present experiment. But also geometric features, in the usual sense, are not there to favour a matching.

Take the classical Fourier descriptors, for instance [17, 8]. A coincidence in the first few coefficients would grant that two shapes are roughly superimposable, differences being limited to the higher frequencies. Size functions (with distance from centre of mass as a measuring function) recognise the presence and size of comparable bumps even if they are differently disposed in the two images to compare; this is a simple case of similarity with no (even rough) superimposition. Such cases are easy to find already in the few examples of Figures 6 and 7. We remind that, anyway, also size functions enjoy completeness theorems like Fourier descriptors, in the case of the present experiment [4].

Other descriptors, like order structure [3], turning function [16], chain code histogram [11], need a sort of local superimposition and are mainly limited to silhouettes. A much closer relative to size functions is the Reeb Graph [9], which has been used so long — as far as we know — only for 3D objects and for heights and distances as the only involved functions.

As for the type of experiment, we are aware only of [11, 10], where irregular objects with no perceivable semantics are treated, but again working on silhouettes and with a smaller dataset.

Anyway, we do not claim that our method be better than the usual ones, but at least independent and possibly complementary.

Following the indications of the referees, whom we thank, we shall extend the experiment to a database of drawings — which we are preparing — and to

a standard trademark database. Moreover, we shall compare the ranking of our system with the one independently produced by human subjects.

7 Conclusions

An on–line demonstrator of a qualitative type of image retrieval runs on two sets of randomly generated curves. Size functions build the mathematical engine. The output shapes are remarkably similar to the input example, but they could not be retrieved by superimposition nor by comparison of currently used feature vectors. Similarity is of an intuitive kind, not relying on geometrical features nor on semantic content. This appears to be a good evidence that size functions are an efficient tool for reducing the "semantic gap" — in the words of [14] — between data and its interpretation by the user. The next steps will be the experiment with real databases and the comparison with human selectors.

Acknowledgements

Work performed within the activity of ARCES "E. De Castro", under the auspices of INdAM-GNSAGA and of the University of Bologna, funds for selected research topics.

References

1. Brown, L., Gruenwald, L.: Tree-based indexes for image data. J. Visual Comm. and Image Representation **9** (1998) 300–313.
2. Burr, D.J.: Elastic matching of line drawings. IEEE Trans. on PAMI, 3 (1981) 708–713.
3. Carlsson, S.: Order structure, correspondence, and shape based categories. in: D.A. Forsyth et al. (Eds.): Shape, Contour and Grouping in Computer Vision, LNCS 1681 (1999) 58–71.
4. Ferri, M., Frosini, P.: Range size functions. Proc. SPIE Conf. on Vision Geometry III, Boston, 1994 Nov. 2-3 (1995) 243-251.
5. Forsyth, D.A., Fleck, M.M.: Automatic detection of human nudes. Int. J. Computer Vision **32** (1999) 63–77.
6. Frosini, P., Landi, C.: Size functions and formal series. Applicable Algebra in Engineering Communication and Computing, 12 (2001) 327–349
7. Donatini, P., Frosini, P., Landi, C.: Deformation energy for size functions. In: E.R. Hancock, M. Pelillo (eds.) Energy Minimization Methods in Computer Vision and Pattern Recognition,, LNCS 1654 (1999) 44–53.
8. Granlund, G.H.: Fourier Preprocessing for hand print character recognition. IEEE Trans. Computers, C-21 (1972) 195–201.
9. Hilaga, M., Shinagawa, Y., Kohmura, T., Kunii, T.L.: Topology matching for fully automatic similarity estimation of 3D shapes. SIGGRAPH 2001, Computer Graphics Proc., Annual Conference Series (2001) 203–212.
10. Iivarinen, J., Peura, M., Särelä, J., Visa, A.: Comparison of combined shape descriptors for irregular objects. In: A.F. Clark (ed.) Proc. 8th British Machine Vision Conference, BMVC'97, Essex (1997) 430–439.

11. Iivarinen, J., Visa, A.: Shape recognition of irregular objects. In: D. P. Casasent (ed.) Intelligent Robots and Computer Vision XV: Algorithms, Techniques, Active Vision, and Materials Handling, Proc. SPIE 2904 (1996) 25–32.
12. Joyce, D.W., Lewis, P.H., Tansley, R,H., Dobie, M.R., Hall, W.: Semiotics and agents for integrating and navigating through multimedia representations. Proc. Storage and Retrieval for Media Databases, vol. 3972 (2000) 120–131.
13. Mokhtarian, F.: Silhouette–based isolated object recognition through curvature scale–space. IEEE Trans. PAMI **17** (1995) 539–544.
14. Smeulders, A.W.M., Worring, M., Santini, S., Gupta, A., Jain, R.: Content–based image retrieval at the end of the early years. IEEE Trans. PAMI **22** (2000) 1349–1380.
15. Veltcamp, R.C., Hagedoorn, M.: State–of–the–art in shape matching. Technical Report UU-CS-1999-27, Utrecht University (1999).
16. Vleugels, J., Veltkamp, R.C.: Efficient image retrieval through vantage objects. in: D.P. Huijsmans and A.W.M. Smeulders (Eds.) Visual Information and Information Systems — Proc. 3rd Int'l Conf. VISUAL'99, Amsterdam, LNCS 1614 (1999) 575–584.
17. Zahn, C.T., Roskies, R.Z.: Fourier descriptors for plane close curves. IEEE Trans. Computers, C-21 (1972) 269–281.