Shape Matching Using the Geodesic Eccentricity Transform - A Study [1)]

Adrian Ion [a], *Gabriel Peyré* [b], *Yll Haxhimusa* [a], *Samuel Peltier* [a], *Walter G. Kropatsch* [a], *Laurent Cohen* [b]

[a] Pattern Recognition and Image Processing Group 183/2, Vienna University of Technology

[b] CEREMADE, Université Paris Dauphine

[a]{ion,yll,sam,krw}@prip.tuwien.ac.at,

[b]{gabriel.peyre,cohen}@ceremade.dauphine.fr

Abstract:
This paper makes use of the continuous eccentricity transform to perform shape matching. The eccentricity transform has already been proven useful in a discrete graph-theoretic setting. We show how these ideas extend naturally to the continuous setting thus bringing a higher geometrical fidelity. The continuous eccentricity transform is used to compute multiscale descriptors for shapes. These descriptors are defined as histograms of the eccentricity transform of a scale-space representation of the shape. These multiscale descriptors are naturally invariant to euclidean motion and bending. They show promising results for shape discrimination.

1 Introduction

Shape recognition is a central topic in computer vision. It requires to set up a signature that characterizes the properties of interest for the recognition [16]. The invariance of this signature to local deformations such as articulations is important for the identification of 2D shapes. Matching can then be carried out over this reduced space of signatures.

Most shape descriptors are computed over a transformed domain that amplifies the important features of the object while throwing away ambiguities such as translation, rotation or local deformations. The Fourier transform of the boundary curve [17] is an example of such transformed domain descriptor adapted to smooth shapes. Shape transformations computed with geodesic distances [2] lead to signatures invariant to isometric deformations such as bending or articulation. To capture salient features of objects, local quantities such as curvature [9] or shape contexts [1] can be computed. They can be extended to bending invariant signatures using geodesic distances [7]. More global features include the Laplace spectra [11] and the skeleton [15].

Some transformations involve the computation of a 2D function defined on the shape, for instance the solution to a linear partial differential equation [3] or geometric quantities [10]. One can also use geodesic distance information such as the mean-geodesic transform [4]. In this paper, we propose to use the eccentricity transform [5], computed using continuous geodesic distances [13], for shape matching. In Sec. 2 the eccentricity transform is presented. Sec. 3 presents our eccentricity histogram descriptors. Experimental results are shown in Sec. 4.

[1)] Partially supported by the Austrian Science Fund under grants S9103-N04 and P18716-N13.

2 Eccentricity Transform

A shape is a planar, connected, bounded and closed set $S \subset \mathbb{R}^2$ with a piecewise-smooth boundary ∂S. In practice we consider a discretized version of S which can be represented using an image f_S of n pixels where f_S is the indicator of the shape: $f_S(x) = 1$ for $x \in S$ and $f_S(x) = 0$ otherwise. Such a discretized representation is usually acquired from some digitized continuous shape.

The geodesic inner distance between two points $x, y \in S$ is defined as

$$d_S(x,y) \overset{\text{def.}}{=} \min_{\gamma \in \mathcal{P}(x,y)} L(\gamma) \qquad \text{where} \qquad L(\gamma) \overset{\text{def.}}{=} \int_0^1 |\gamma'(t)| \mathrm{d}t,$$

where $\mathcal{P}(x,y)$ is the set of paths $\gamma(t) \subset S$ joining x to y: $\gamma(0) = x$ and $\gamma(1) = y$. The computation of the distance function $U(x) \overset{\text{def.}}{=} d_S(x_0, x)$ to some starting point $x_0 \in S$ can be computed efficiently as the solution of the non-linear Eikonal equation:

$$\forall\, x \in S, \quad \|\nabla U(x)\| = 1 \qquad \text{and} \qquad U(0) = 0. \tag{1}$$

We note that this resolution is restricted to S, which makes the function U different from the traditional euclidean distance function $x \mapsto \|x_0 - x\|$ for non-convex shapes. Fig. 1 shows a comparison of the geodesic and euclidean distances. The Eikonal equation (1) can be solved in $O(n \log(n))$ operations for a grid of n points using the Fast Marching algorithm [13].

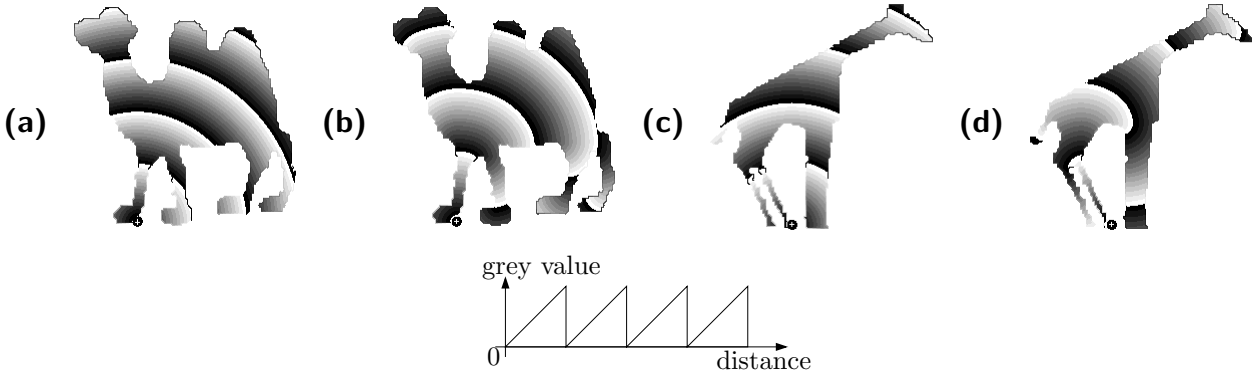

Figure 1: ***(a,c) Two examples of euclidean distance functions*** $\|x_0 - x\|$ ***restricted to*** S***. (b,d) The geodesic distances*** $d(x_0, x)$ ***to some starting point*** x_0***.***

The eccentricity transform of a shape S, assigns to each point $p \in S$ the shortest geodesic distance to the point of S farthest away from it. The eccentricity of the shape S is defined as

$$\mathrm{ECC}_S(x) \overset{\text{def.}}{=} \max_{y \in S} d_S(x,y) = \max_{y \in \partial S} d_S(x,y). \tag{2}$$

This is a continuous and piecewise smooth function. A point y that reaches the global maximum in (2) is called *eccentric*. We denote $\mathcal{E}(S)$ the set of eccentric points. An important property due to [5] is that $\mathcal{E}(S)$ is included in the boundary of S: $\mathcal{E}(S) \subset \partial S$. The set of eccentric points allows to define a segmentation of S into eccentric regions

$$S = \bigcup_{x \in \mathcal{E}(S)} A_x \qquad \text{where} \qquad A_x = \{y \setminus \mathrm{ECC}_S(y) = d_S(y,x)\}.$$

The eccentricity is computed by performing a Fast Marching propagation from each $x \in \partial S$ in order to compute the set of distances $\{d(x,y)\}_{x\in\partial S}$. This method requires $O(|\partial S| n \log(n))$ operations where $|\partial S|$ is the number of pixels on the boundary of S. Fig. 2 shows the eccentricity transform of two shapes.

The eccentricity transform has already been studied on a discrete graph [5]. The set of values $\{\mathrm{ECC}_S(x)\}_{x\in S}$ is invariant under rigid motion and isometric transform of S, which includes bending. It is also nearly-invariant under articulations, see [7, 5].

Another property of ECC_S is that it is robust to salt and pepper noise that might create holes in S, because a hole or a segmentation error of size ε only modifies the eccentricity ECC_S by no more than ε. It is very different from local descriptors such as the curvature [9] or even global ones such as the structure of the skeleton [15] which are not robust to this kind of noise.

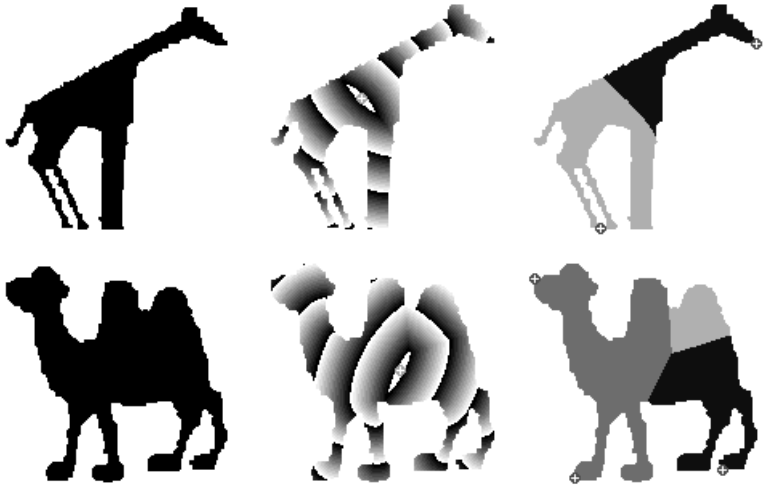

Figure 2: ***Left: the binary shape indicator function f_S for two different shapes. Center: eccentricity transform ECC_S, the values range from 0 to $\max_x ECC_S(x)$. The eccentric center $C(S) = \mathrm{argmin}_x E_S(x)$ is displayed as a point. Right: eccentric points $\mathcal{E}(S)$ together with the segmentation in eccentric regions A_x for $x \in \mathcal{E}(S)$.***

3 Eccentricity Histogram Matching

In order to match two shapes from two binary images we first create a shape descriptor for each of them and then match these descriptors to obtain a similarity measure. This paper proposes two approaches based on a single and a multiscale descriptor.

Mono-scale descriptor. The basic building block for our shape descriptor is the histogram h_S of the eccentricity transform ECC_S of the shape S. We use m bins to estimate the histogram and in numerical applications, we use $m = 200$. The histogram descriptor is then the vector $h_S \in \mathbb{R}^m$ defined by

$$\forall\, i = 1, \ldots, m, \quad h_S(i) = \frac{1}{|S|} \# \left\{ x \in S \setminus \frac{i-1}{m} \leqslant \frac{ECC_S(x) - \min(ECC_S)}{\max(ECC_S) - \min(ECC_S)} < \frac{i}{m} \right\},$$

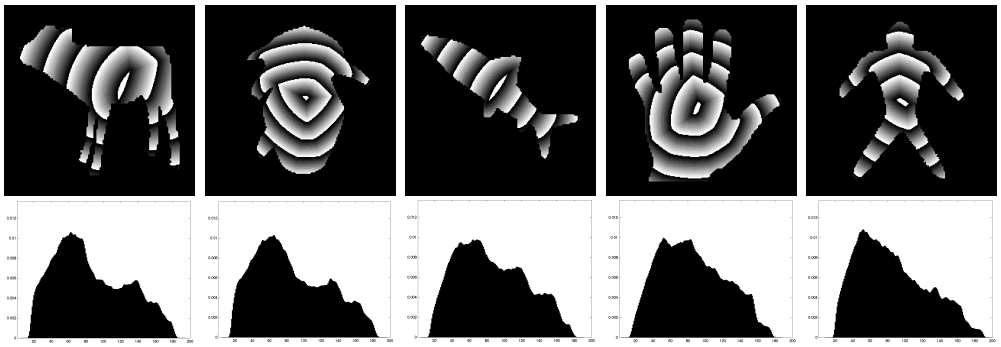

Figure 3: Top: eccentricity transform $\mathbf{ECC}_S$ for some shapes S used during experiments. Bottom: corresponding histograms h_S.

where $|S|$ is the number of pixels in S. Fig. 3 shows examples of histograms for shapes with different geometric features. We note that the histogram h_S is invariant under euclidean transformations, scaling and isometric bending of S.

Multiscale descriptor. In order to capture more geometric information about a shape S, we compute a non-linear scale-space of S and extract the histograms of the eccentricity over a scale-space domain.

In order to smooth the shape, we perform the following mean curvature evolution [13][1)]

$$\frac{\partial \gamma_t}{\partial t}(u) = \kappa_t(u)\, \vec{n}_s(u) \qquad \text{with} \qquad \gamma_0(u) = \partial S(u),$$

where $\kappa_t(u)$ is the curvature of the curve $\gamma_t(u)$ and $\vec{n}_t(u)$ is the normal vector to the curve. The curve γ_s is thus a smoothed version of the boundary of S after a diffusion during a time t. Fig. 4, left, shows the process of shape smoothing.

This curve smoothing allows to define a set of shapes $\{S^k\}_{k=1}^K$ where S^k is the shape whose boundary is $\partial S^k \stackrel{\text{def.}}{=} \gamma_{t_k}$ with dyadic time steps $t_k = \tau 2^k$. The number of shapes is set to $K = 3$ in our numerical tests and τ is of the order of 5 pixels.

Our shape descriptor is composed of the histograms of the eccentricity transform over the scale space domain

$$D_S \stackrel{\text{def.}}{=} \{h_{S^1}, \ldots, h_{S^k}\}.$$

Fig. 4 shows an example of descriptor. During smoothing, elongated parts tend to disappear, which affects the histograms.

Comparison of histograms. In order to match two descriptors D_S and $D_{\tilde{S}}$ of two shapes S and $\tilde{S}$, we need to compute a meaningful distance between histograms. Let $h \in \mathbb{R}^m$ and $\tilde{h} \in \mathbb{R}^m$ be two histograms. We propose to use the simple L_2-norm defined by

$$\delta(h, \tilde{h}) \stackrel{\text{def.}}{=} \sqrt{\sum_{i=1}^{m} (h(i) - \tilde{h}(i))^2}.$$

[1)] Note that more elaborate methods, like the one presented in [9], could be used.

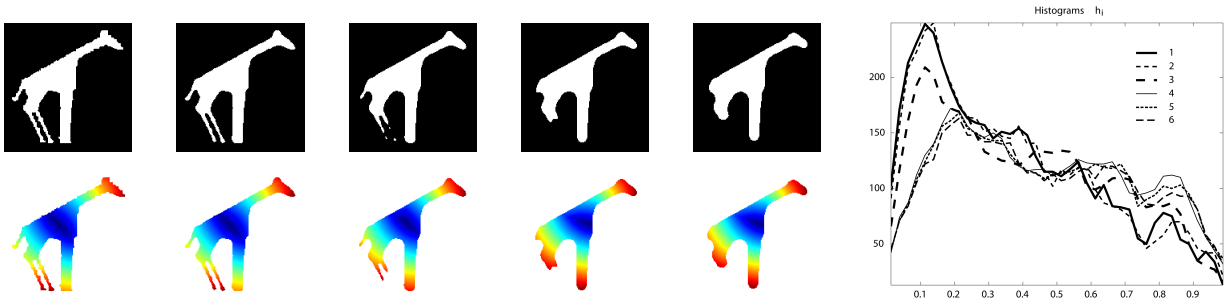

Figure 4: ***Upper left : smoothed shapes*** S^k ***for*** $0 \leqslant k \leqslant 5$***. Bottom left : corresponding eccentricity transform*** ECC_{S^k}***. Right : overlay of the histograms*** h_{S^k}***.***

One could use more elaborate metrics such as the χ^2 metric, the earth mover distance or those defined in [10], but we found in numerical experiments that all these metrics give results similar to δ.

We can compute the distance $\Delta(S, \tilde{S})$ between two shapes $S, \tilde{S}$ as the weighted sum of the distance of their histogram descriptors $D_S = \{h_{S^1}, \ldots, h_{S^K}\}$ and $D_{\tilde{S}} = \{h_{\tilde{S}^1}, \ldots, h_{\tilde{S}^K}\}$

$$\Delta(S, \tilde{S}) \stackrel{\text{def.}}{=} \sum_{k=1}^{K} w_k \delta(h_{S^k}, h_{\tilde{S}^k}).$$

In our experiments we used $w_k = \frac{1}{2^{k-1}}$.

4 Experiments

For the experiments we have used three shape databases: Kimia 25 [14], Kimia 99 [12] and MPEG7 CE-Shape-1 [6]. For the Kimia 25 and Kimia 99 databases, the eccentricity ECC_S is computed directly on the images as shown in section 2. For the MPEG 7 database, which contains 1400 images with varying sizes, we rescale the images to 256×256 pixels.

A shape database is composed of q shapes $\{S_i\}_{i=1}^q$ and each shape S_i has a label $\ell(i) \in \{1, \ldots, \ell_{\max}\}$. Each label value $1 \leqslant \ell \leqslant \ell_{\max}$ defines a class of shapes $\mathcal{C}_\ell \stackrel{\text{def.}}{=} \{S_i \setminus \ell(i) = \ell\}$. The left column of the three blocks of Fig. 5 shows some shapes from the Kimia database, ordered by classes (such as fish, planes, rabbits, etc.). Any shape matching algorithm α assigns to each shape S_i a vector of best matches Φ_i where $\Phi_i(1)$ is the shape the more similar to S_i, $\Phi_i(2)$ is the second hit, and so on.

For the Kimia 99 database, $\ell_{\max} = 9$ and $q = 99$. We measure the efficiency of various matching algorithms on Kimia databases by the number of correct matches for each ranking position k

$$\mathrm{Match}_k(\Phi) \stackrel{\text{def.}}{=} \sum_{i=1}^{q} 1_{\ell(i)=\ell(\Phi_i(k))} \leqslant q.$$

Tables 1 and 2 give the value of Match_k for various shape matching algorithms.

In the case of the MPEG7 database, which contains $\ell_{\max} = 70$ classes with 20 images each, so $q = 70 \times 20$, the efficiency of matching algorithms is computed using the standard Bullseye

Algorithm α	k=1	2	3
Sharvit et. al	23	21	20
ECC-Hist	25	20	16
ECC-M.-Hist	25	21	15
Gdalyahu and Weinshall	25	21	19
Belongie et. al	25	24	22
ID-Shape Context [1]	25	24	25

Table 1: The value of $\text{Match}_k(\Phi)$ for various algorithms on the Kimia 25 database. See [7] for a description of these algorithms.

Algorithm α	k=1	2	3	4	5	6	7	8	9	10
ECC-Hist	99	87	74	67	64	49	52	45	38	33
ECC-M.-Hist	99	88	79	68	72	60	55	44	43	40
Shape Context	97	91	88	85	84	77	75	66	56	37
Gen. Model	99	97	99	98	96	96	94	83	75	48
Shock Edit	99	99	99	98	98	97	96	95	93	82
ID-Shape Context [1]	99	99	99	98	98	97	97	98	94	79

Table 2: The value of $\text{Match}_k(\Phi)$ for various algorithms on the Kimia 99 database. See [7] for a description of these algorithms

test. This test counts the number of correct hits (same class) in the first 40 hits

$$\text{Bullseye}(\Phi) \stackrel{\text{def.}}{=} \frac{1}{20q} \sum_{k \leqslant 40} \sum_{i=1}^{q} 1_{\ell(i)=\ell(\Phi_i(k))} = \frac{1}{20q} \sum_{k \leqslant 40} \text{Match}_k(\Phi) \leqslant 1.$$

For each image there can be at most 20 correct hits i.e. a maximum of 20×1400 hits can be obtained and thus $\text{Bullseye}(\Phi) \leqslant 1$. Table 3 gives the value of Bullseye for various shape matching algorithms.

The overall results over both Kimia 25 and Kimia 99, and over MPEG 7 are slightly bellow the state of the art. It is important to consider that the proposed methods are simple since they only require the computation of geodesic distances and histograms. In contrast, the most efficient shape matching algorithms [1, 7] are more complicated and require extraction of salient features and local signatures that need to be aligned or registered.

Case Study - Kimia 25 Fig. 5 shows the retrieval results for Kimia 25 using the single scale descriptor. The first column shows the 25 shapes S_i. The following set of shapes forms an array where the shape at row i and column k is $\Phi_i(k)$, the rank-k shape associated to S_i.

The Kimia 25 database has shapes from 6 classes: 5 classes with 4 images each, and one (hands) with 5 images (1 simulating a segmentation error). As one can see looking at the first 5 columns, the class with the best results are rabbits, followed by tools, hands, fishes,

Algorithm α	Bullseye(Φ)
ECC-Hist	44.28%
ECC-M.-Hist	45.90%
Shape Context	64.59%
ID-Shape Context	68.83%

Table 3: The value of Bullseye(Φ) for various algorithms on the MPEG 7 databases.

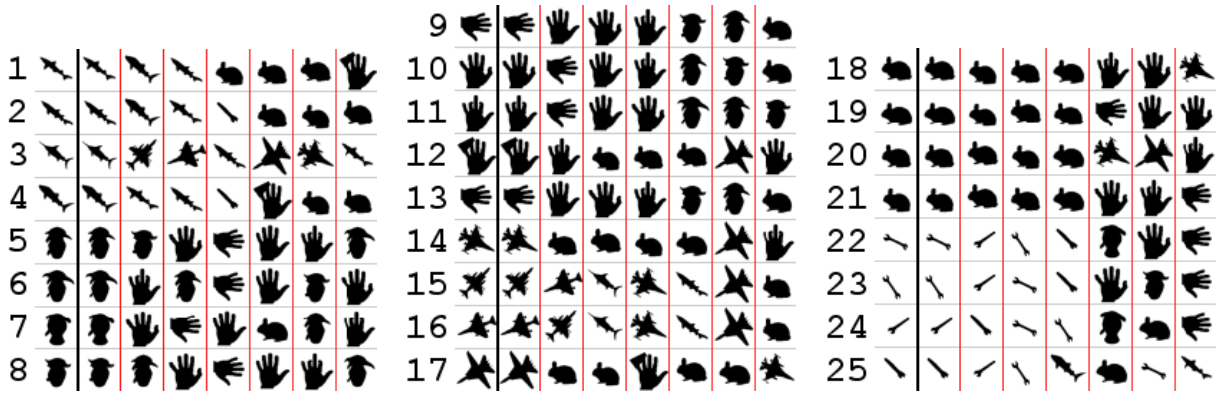

Figure 5: Retrieval results for the single scale descriptor on the Kimia 25 database.

airplanes and greebles (the shapes that do not look like anything we know). Two questions immediately rise when looking at these results:

1. Why are the greebles considered to be more similar to the hands than to other greebles?
2. Why does a rabbit appear in so many cases when the matching has failed?

For the first question, look at the histograms of the greebles and the unoccluded hands (see Fig. 6.a). The histograms are very similar even though the shapes are different, e.g. the histogram of the first greeble (Fig. 6.a top-left) looks more similar to the hands, than the second and third greeble. This is due to the abstraction of a 2D shape to a 1D histogram, which disregards the structure of distances/paths.

For the second question, we consider the shapes in Fig. 6.b (a rabbit - S_{19}, and two tools - S_{25} and S_{22}), and the results, Φ_{25}, in the bottom row of Fig. 5. When matching S_{25}, the rabbit has a better score than S_{22}, even though one might say that the histograms of S_{25} and S_{22} reveal more similar distance characteristics than the histogram of S_{19} (see Fig. 6.b) i.e. both S_{25} and S_{22} have more long distances than medium, and short, while S_{19} has a peak in the medium. This is due to histogram matching methods, inherently low level, failing to capture the high level context of the task. Besides the L_2-norm we have also looked at the χ^2 statistic [7] and diffusion distance [8] and both produce similar results.

Geometrical properties of the shapes are well captured by our low-dimensional descriptors. For instance, elongated shapes are well separated from more compact shapes. However, more advanced geometrical features, such as intricate structural properties are thrown away by our signature extraction. This is for instance why the class "greebles" is not separated enough from the class "hands".

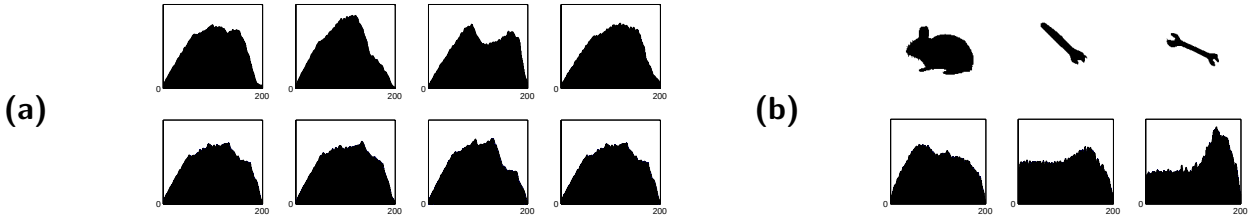

Figure 6: (a) Histograms for: top: greebles, and bottom: unoccluded hands. (b) top: Three shapes from the Kimia 25 database, and bottom: their eccentricity histograms.

5 Conclusion

In this paper we have presented the continuous eccentricity transform. We studied it's application to shape matching using descriptors computed with the continuous eccentricity transform. These descriptors are defined as histograms of the eccentricity transform of a scale-space representation of the shape. Results of the proposed method on three well known databases are presented and compared with the state of the art methods. We plan to study decomposition of articulated shapes using the eccentricity transform and extend the current approach to part based shape matching.

References

[1] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(4):509–522, 2002.

[2] A. M. Bronstein, M. M. Bronstein, A. M. Bruckstein, and R. Kimmel. Matching two-dimensional articulated shapes using generalized multidimensional scaling. In *Proc. Conf. on Articulated Motion and Deformable Objects*, pages 48–57, 2006.

[3] L. Gorelick, M. Galun, E. Sharon, R. Basri, and A. Brandt. Shape representation and classification using the poisson equation. In *CVPR (2)*, pages 61–67, 2004.

[4] A. Ben Hamza and H. Krim. Geodesic object representation and recognition. In *DGCI: International Workshop on Discrete Geometry for Computer Imagery*, 2003.

[5] W. G. Kropatsch, A. Ion, Y. Haxhimusa, and T. Flanitzer. The eccentricity transform (of a digital shape). In *13th DGCI*, pages 437–448, Hungary, October 2006. Springer.

[6] L. J. Latecki, R. Lakämper, and U. Eckhardt. Shape descriptors for non-rigid shapes with a single closed contour. In *CVPR 2000, 13-15 June 2000, Hilton Head, SC, USA*, pages 1424–1429. IEEE Computer Society, 2000.

[7] H. Ling and D. W. Jacobs. Using the inner-distance for classification of articulated shapes. In *CVPR 2005, 20-26 June 2005, San Diego, CA, USA*, pages 719–726, 2005.

[8] H. Ling and K. Okada. Diffusion distance for histogram comparison. In *CVPR 2006, 17-22 June 2006, New York, NY, USA*, pages 246–253, 2006.

[9] F. Mokhtarian and A. K. Mackworth. A theory of multiscale, curvature-based shape representation for planar curves. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 14(8):789–805, August 1992.

[10] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin. Shape distributions. *ACM Trans. Graph.*, 21(4):807–832, 2002.

[11] M. Reuter, F-E. Wolter, and N. Peinecke. Laplace-spectra as fingerprints for shape matching. In L. Kobbelt and V. Shapiro, editors, *Symp. on Solid and Physical Modeling*, pages 101–106. ACM, 2005.

[12] Thomas B. Sebastian, Philip N. Klein, and Benjamin B. Kimia. Recognition of shapes by editing their shock graphs. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(5):550–571, 2004.

[13] J.A. Sethian. *Level Sets Methods and Fast Marching Methods.* Cambridge Univ. Press, 2nd edition, 1999.

[14] D. Sharvit, J. Chan, H. Tek, and B. Kimia. symmetry-based indexing of image databases, 1998.

[15] K. Siddiqi, A. Shokoufandeh, S. Dickinson, and S. W. Zucker. Shock graphs and shape matching. *International Journal of Computer Vision*, 30:1–24, 1999.

[16] R. C. Veltkamp and L.J. Latecki. Properties and performance of shape similarity measures. In *Proc. of IFCS 2006*, 2006.

[17] C. T. Zahn and R. Z. Roskies. Fourier descriptors for plane closed curves. *IEEE Trans. Computer*, 21(3):269–281, March 1972.