J Math Imaging Vis (2012) 42:225–237
DOI 10.1007/s10851-011-0291-1

A Pruning Algorithm for Stable Voronoi Skeletons

Andoni Beristain · Manuel Graña · Ana I. Gonzalez

Published online: 3 May 2011

Abstract We present a skeleton computation algorithm for binary image shape which is stable and efficient. The algorithm follows these steps: first the shape boundary curves are subsampled, then the Voronoi Skeleton is computed from the resulting reduced boundary set of points, and finally, a novel two stage pruning procedure is applied to obtain a simplified skeleton. The first stage removes skeleton edges non fully included in the shape. The second stage applies an enhanced variation of the Discrete Curve Evolution (DCE) for Voronoi skeletons. We obtain improved skeleton stability, complexity reduction and noise robustness. Pruning computing time efficiency is improved thanks to some properties of Voronoi skeletons. Entire skeleton edges can be removed or retained on the basis of conditions tested on the edge endpoints. Pattern recognition experiments and skeleton stability experiments of the algorithm outperform previous approaches in the literature.

Keywords Skeleton · Voronoi skeleton · Pruning · Pattern recognition

A. Beristain (✉) · M. Graña · A.I. Gonzalez
Facultad de Informática de San Sebastian, Computational Intelligence Group, University of the Basque Country, Manuel Lardizabal 1, 20018 Donostia San Sebastian, Gipuzkoa, Spain
e-mail: andoni.beristain@gmail.com
url: www.ehu.es/ccwintco

M. Graña
e-mail: manuel.grana@ehu.es

A.I. Gonzalez
e-mail: ai.gonzalez@ehu.es

1 Introduction

The most intuitive definition of a shape skeleton is the line obtained through repeatedly thinning the shape [1]. Such line lies in the middle of the shape and somehow captures the shape main morphological features. The process of obtaining the skeleton can be assimilated to a grassfire burning outside-in an area corresponding to the shape [2]. The places where fires starting from opposite boundary regions meet constitute the skeleton. Skeletons are a quite primitive representation and visualization tool which are spontaneously used for visual communication among humans. There are some psychological evidence that humans use skeletons in some object recognition processes [3, 4]. A more formal definition is that of Medial Axis Transform (MAT) [5] which consists in the centers of all circles which intersect twice with the shape boundary. Skeletons and MAT are for all practical purposes equivalent concepts. They condense the main features of the original shape, which are suitable for shape description and recognition. The skeleton structure may show invariance to shape bending, widening, elongation, decomposition and warping. It provides simultaneously information about location, orientation, and size of an object. Therefore, they have been used in computer vision applications [6–8]. However, skeletons have two main drawbacks. First, conventional skeleton computation algorithms are very sensitive to noise sources like boundary noise, i.e. small boundary perturbations may introduce big changes in the skeleton, discretization noise and connectivity type. And second, the computational complexity of skeleton computation algorithms is usually too high for real-time applications.

The algorithm proposed in this paper is aimed to obtain stable skeletons and perform in real-time. We present a combination of a skeletonization and a pruning algorithm making use of a improvement of the skeleton pruning procedure presented in [9]. The algorithm has the following steps: (1) it computes the Voronoi Tessellation of an uniform subsampling point set of the shape boundary curves, (2) it subsamples the shape boundary curves obtaining a simplified polygon using a Discrete Curve Evolution (DCE) procedure (3) performs a first pruning stage removing every Voronoi segment not completely enclosed in the shape (4) it removes all the Voronoi segments not fulfilling the DCE skeleton pruning criteria. We will call this algorithm *BERIS* through the paper. We prove that the first pruning strategy can be done efficiently testing the Voronoi edge corresponding sites. Subsampling not only improves the efficiency but also the skeleton stability. The motivation to use Voronoi skeletons is that the skeleton is composed of a sequence of line segments that can be treated as units, allowing a great computational speed-up. Additionally, the skeleton connectivity is guaranteed. The skeleton stability has been tested directly, measuring the variability of the skeletons obtained on a database of hand gestures, and indirectly, performing pattern recognition experiments on this database and on other benchmark database. We compare with the result of the algorithm presented in [9] over these databases. Some of the ideas of this paper were first presented in [10].

The paper is structured as follows: Sect. 2 presents a short review of the literature on skeleton computing and pruning. Section 3 presents some definitions used in the paper. Section 4 introduces the Voronoi skeleton and some stability issues. Section 5 introduces the Discrete Curve Evolution. Section 6 gives our skeleton computing algorithm. Section 7 contains the proof of the theorems that allow efficient pruning of the Voronoi skeleton. Section 8 presents the skeleton graph matching algorithm used in the experiments. Section 9 provides experimental results. Finally, Sect. 10 gives our conclusions.

2 Literature Review

Starting with the intuitive definition of the skeleton in the introduction section, many authors have introduced different formal definitions, with similar concepts but subtle differences. Reeb Graphs [11–13] are a general precedent of the definition of the Medial Axis Transform (MAT) using the *grassfire transform* [14]. According to Blum, and taking the Symmetry Set definition [15] as starting point, the Medial Axis Transform (MAT) is defined [5] as the closure of the centers of all the circles which are bi-tangent to the shape boundary curve. The same definition is given to the skeleton in [1]. Therefore, the skeleton points are equidistant to the shape boundary curves. Jain defines skeletons in terms of Maximal Disks [16]. A disk B is said to be the maximal disk included in a set A if $B \subseteq A$, and if there is another disk D such that $B \subseteq D$, then $B \nsubseteq A$. Finally, Mather defines the *Maxwell Set* [17] as the set of locations internal to the object with more than one corresponding closest boundary point in the sense of Euclidean distance. Each point in this set is augmented with its distance to the boundary. The *Shock Graph*s [18–20] add morphological features to the skeleton for application purposes. From the literature [21–25], we have found that skeletonization procedures fall into one of these categories:

1. *Marching Front*: Simulate the grassfire transform. The *morphological thinning* performs an iterative peeling procedure on the shape boundary [2, 23, 26, 27]. This procedure is not robust to rotation of the shape. The *curve evolution* or *shocks of boundary evolution* uses a partial differential equation on a functional definition of the shape boundary [28–30]. The algorithms are computationally demanding ($O(n \log n)$ or $O(n^2)$), but provide connected and complete skeletons.
2. *Distance Transform* (DT): The DT [31, 32] function assigns to each point in the shape the minimum distance to a shape boundary point. The skeleton corresponds to the DT local maxima. Algorithms can be very fast, i.e. $O(n)$ [22], but skeleton connectivity is not guaranteed. One interesting property of skeletons is that the original object shape can be reconstructed from its skeleton and the DT transform value of each skeleton point.
3. Voronoi Skeletons [33]: The skeleton is given by the subset of the Voronoi edges and vertices contained inside the shape. The Voronoi Tessellation is computed using the shape boundary pixels as Voronoi sites. Our proposed *BERIS* algorithm uses this approach.

Most of the skeletonization procedures are highly sensitive to boundary noise, which creates spurious branches in the skeleton. In order to improve the stability of the resulting skeleton, a regularization procedure is usually carried out. This regularization procedure can be done either by smoothing the shape boundary, prior to the skeleton computation, or performing a pruning of the skeleton. The skeleton pruning removes the spurious branches [34–38] without removing any meaningful part of the skeleton. Skeleton pruning can be done as a skeleton post-processing, or during skeleton generation.

Pruning is based on salience measures. These measures are heuristics based on local or global information. Local salience descriptors include: DT function value, the maximum distance between skeleton generative points,[1] and the maximum angle formed by the skeleton point and two of its generative points. Global salience descriptors include measuring the area difference between the shape reconstruction before and after pruning, or the minimum curve length between the generative points of the skeleton point. A bound-

[1] Generative points of an skeleton point are the two closest and equidistant boundary points.

ary smoothing [39, 40] pre-processing of the shape boundary curves to denoise them can also be used, but meaningful boundary features can be removed. A pruning method [9] should preserve topology (homotopy type), be continuous, i.e., small variations in the significance measure should result in small changes to the computed skeleton, and its significance measure should be local.

The pruning procedure in [9] uses the Discrete Curve Evolution (DCE) procedure for subsampling the discrete definition of a shape boundary curves into a simple polygon which represents the most relevant geometry of the shape. Then, the convex hull of this polygon is used to build the pruning significance measure. A skeleton point is removed if its generative points are not in two different edges of this convex hull. This pruning method can be applied after any skeleton computing algorithm meeting two conditions: (1) every skeleton point must be the center of a maximal disk and (2) the generative points in the shape boundary curves must be available.

3 Preliminary Definitions

An image is a function $I : D \to \mathbb{R}$. The image domain can be continuous $D \subset \mathbb{R}^2$ or discrete $D \subset \mathbb{Z}^2$. We denote $|p, q|_A$ the adjacency relation between pixel sites, which can be 4- or 8-adjacency for discrete domain images. The corresponding neighborhoods are the N_4 or N_8 adjacent pixel sites. The transitive closure of the adjacency is the connectivity relation. A connected region is a set of connected pixel sites. An object's shape is a connected region $F \subseteq D$ identified through some segmentation procedure, which can include holes in it. The shape boundary curves, denoted C, can be constituted by one or several connected curves $C = \{C_1, \ldots, C_n\}$. In a discrete domain, the shape boundary is composed of pixels $p \in F$ having neighbors inside and outside the shape: $\exists a, b \in N_8(p)$ such that $a \in F$ and $b \notin F$. In the continuous domain, we say that any ball around the boundary pixel $B_\varepsilon(p) = \{x | \|x - p\| < \varepsilon\}$ contains pixels inside and outside the shape: $\forall \xi > 0, \exists a, b \in B_\xi(p)$ s.t. $a \in F \wedge b \notin F$. In discrete domain images, each of the boundary curves is represented as a sequence of connected pixels $C_i = \{p_1, \ldots, p_n\}$ computed visiting the boundary pixels in clockwise or anticlockwise direction, starting from an arbitrary position.

For a point m in the Medial Axis, its generative points are the intersections of the shape boundary and the maximal disk centered in m included in F.

The Hausdorff distance [41] is a measure of the similarity between two sets A, M. Let us define the minimum distance between a point x to the set A as $d_h(x, A) = \min_{y \in A} \|x - y\|$, then the Hausdorff distance between sets A and M, $d_H(A, M)$ is defined as,

$$d_H(A, M) = \max\left[\max_{x \in A} d_h(x, M), \max_{y \in M} d_h(y, A)\right]. \qquad (1)$$

4 Voronoi Skeleton

The Voronoi Skeleton [33] is the skeleton computed from the Voronoi Tessellation of the image domain induced by the shape boundary. The Voronoi Tessellation V consists of a decomposition of the image domain D into convex regions determined by the Voronoi sites $V = \{v_i\}$, called Voronoi regions $R(i)$, formally:

$$R(i) = \left\{x \in \mathbb{R}^2 \mid \|x - v_i\| \le \|x - v_j\| \; \forall v_j \ne v_i\right\}. \qquad (2)$$

Each Voronoi polygon is bounded by Voronoi edges and vertices. Each Voronoi edge s_{ij} is the locus of the intersection of exactly two and no more different Voronoi polygons, and it is therefore determined only by two Voronoi sites:

$$s_{ij} = R(i) \cap R(j). \qquad (3)$$

Each Voronoi vertex is the intersection of three or more Voronoi regions, which can be expressed as the intersection of at least two Voronoi segments. The set of Voronoi vertices is defined as:

$$T = \left\{x \in \mathbb{R}^2 \mid \exists i, j, k \in V; x \in s_{ij} \cap s_{ik} \cap s_{jk}\right\}.$$

According to [33, 34, 42, 43], for images with a continuous domain, the Voronoi Tessellation computed from $V \subset C$, where C is the set of points in the boundary curves of the shape F, the Voronoi edges lying inside the shape F

$$S_v = \bigcup_{i,j \in V} \left[s_{ij} \cap F\right], \qquad (4)$$

become a closer representation of the Medial Axis of F, as $V \to C$. When $V = C$ then S_v becomes identical to the Medial Axis.

The advantages of Voronoi skeletons are: guaranteed connectivity of the skeleton, not being affected by discretization issues and convergence to the true Medial Axis. Their disadvantages are their computational complexity for high number of Voronoi sites[2] and their sensitivity to boundary perturbations.

[2] The average Voronoi Tessellation computational complexity is $O(n \log n)$, with $n = |V|$. An optimal lower bound in $O(n)$ is proved in [44].

4.1 Stability of the Voronoi Skeleton

One way to obtain stable Voronoi skeletons is by previous subsampling of the shape's boundary curves [24].

Definition 1 An *uniform sampling* D of the shape boundary curve C is defined as a sampling so that the distance from any point on the curve C to the nearest sample in D is smaller than a constant δ.

$$\forall c_i \in C \; \exists d_j \in D \mid \|c_i - d_j\| \leq \delta. \tag{5}$$

One convenient procedure to perform a uniform sampling is to select points that are at arch length distance δ along the boundary curve. For instance, in a discrete boundary curve $C_i = \{p_1, \ldots, p_n\}$, setting an integer value as sampling rate δ, the sampling could be computed as

$$D = \left\{p_1, p_{1+\delta}, p_{1+2\delta}, \ldots, p_{1+n'\delta}\right\}, \tag{6}$$

where $n' = \lfloor \frac{n}{\delta} \rfloor$.

The question arises of characterizing when this uniform subsampling will be preserve the features of the original shape's boundary. Let us consider r-regular shapes [34, 45].

Definition 2 r-regular shape [42, 45]: Let B_0 be a circle of radius one, and rB_0 denote a circle of radius r. A shape F is said to be r-regular if it is morphologically invariant to opening and closing with a disk of radius r, with $r > 0$.

$$F = (F \ominus rB_0) \oplus rB_0 = (F \oplus rB_0) \ominus rB_0. \tag{7}$$

The most relevant property of r-regular shapes is that if the shape boundary is uniformly sampled with density $\delta < 2r$, the polygon induced by the sampling retains all the topological properties of original shape boundary curves. Therefore, the Voronoi skeleton will be topologically equivalent to the original Medial Axis. Topological equivalence means that they will have the same branching structure. To obtain a r-regular shape is enough performing a opening-closing operation with the desired resolution over the shape.

Let us consider how the uniform sampling affects the λ-Medial Axis, which is a scale-dependent representation of the shape.

Definition 3 λ-*Medial Axis* [46]: the set of Medial Axis points m for which $\beta(m) \geq \lambda$, where $\beta(m)$ is the radius of the minimum circle which includes all the generative points of m. For Medial Axis points with two generative points c_i and c_j, it is half of the distance between them $\beta(m) = \frac{1}{2}\|c_i - c_j\|$. For Medial Axis points with more generative points this value will be half of the largest distance between any two generative points.

The following Theorem 1, which is proven in [46], gives bounds on the approximation, in the sense of the Hausdorff distance, to the Medial Axis of the original shape obtained from a subsampled boundary.

Theorem 1 [46] *The set M consisting of the closure of the points v of the Voronoi diagram (including points in edges and 2D faces), for which the distance between two of the nearest samples (the generative points) to v is at least δ, is an approximation of the λ-medial axis of C, with $\lambda = O(\delta^{\frac{1}{4}})$, and where the Hausdorff distance between M and the λ-medial axis is $O(\delta^{\frac{1}{8}})$.*

It can be interpreted as follows: if the Hausdorff distance $d_H(C, D)$ between a boundary curve C and some other set D is at most a constant $\delta = O(\lambda^4)$, then the Hausdorff distance between the λ-medial axis of C and the Voronoi skeleton constructed using Voronoi sites D is $O(\sqrt{\delta})$. Therefore, the Voronoi skeleton of the uniformly subsampled boundary preserves the topology of the shape features of scale greater than the subsampling ratio.

5 Discrete Curve Evolution

Discrete Curve Evolution (DCE) [47–49] is a curve sampling method that keeps the most meaningful geometrical information for shape recognition. The main idea of the DCE procedure is to iteratively remove the point with less contribution to the global shape, reducing the boundary curve complexity while keeping the most important information of the shape. Digital curves are approximated by polygons. The DCE procedure removes one vertex of that polygon at each step replacing the two consecutive edges connected to it. The vertex with the minimum relevance $K(S_i, S_{i+1})$ is removed at each iteration, where S_i and S_{i+1} are consecutive polygon segments. Relevance is defined as:

$$K(S_i, S_{i+1}) = \frac{\beta(S_i, S_{i+1}) l(S_i) l(S_{i+1})}{l(S_i) + l(S_{i+1})}, \tag{8}$$

where $\beta(S_i, S_{i+1})$ is the angle defined by segments S_i and S_{i+1}, and l is the segment length normalized with respect to the total length of the polygon C. The value $K(S_i, S_{i+1})$ is a model of the contribution of $S_i \cup S_{i+1}$ to the shape, it is monotonically increasing with respect to the relative lengths and the total curvature of segments S_i and S_{i+1}. The DCE procedure produces no displacements or blurring of the contour features. The relevance value $K(S_i, S_{i+1})$ is stable under deformations on the shape boundary curves, because noise elimination takes place in the early stages of the evolution.

The DCE procedure converges to a convex polygon. It can be stopped when reaching this condition, a minimum number of segments or some criteria on the difference between the original shape and the DCE approximation [49]. When the shape has holes, it is feasible to specify two minimum number of vertices: θ_E for the external boundary curve and θ_I for the inner boundaries of the shape holes. In general $\theta_E \gg \theta_I$, because the external boundary curve is considered more representative for the shape description than the boundary curves of the shape holes and also because the external boundary curve is usually longer. A constant threshold can reduce the complexity of further steps for shape recognition, but also limits the maximal detail resolution of a shape.

6 The Skeleton Computation Algorithm

The algorithm steps, illustrated in Fig. 1, are as follows:

1. *Shape boundary subsampling*: We apply uniform subsampling (Definition 1) to the shape boundary curves C to obtain the set of Voronoi sites $V \subset C$ which will be used to compute the Voronoi Tessellation.
2. *Voronoi Tessellation*: We compute the Voronoi edges $\{s_{ij} \mid i, j \in V\}$ of the Voronoi Tessellation induced by V. All the lines in Fig. 2 correspond the set of Voronoi segments computed on a chicken shape.
3. *Discrete Curve Evolution computation (DCE)*: We compute C_{DCE}, the polygonal approximation of the original shape boundary curves C, with a DCE procedure (see Sect. 5). The termination criterion is a limit number of vertices for each boundary curve C_i. In our tests we used $\theta_E = 15$ and $\theta_I = 6$, for shapes in 320 × 240 resolution images. When DCE terminates before a convex polygon is obtained, the convex hull of the polygon is considered as C_{DCE}.
4. *Pruning*: We perform two pruning stages on the set of Voronoi edges:
   (a) *First Pruning Stage*: We remove Voronoi segments not entirely contained in the original shape F. Figure 2 shows the removed segments as thin segments. Formally:

$$S_{VA} = \bigcup_{i,j \in V} (s_{ij} \subset F). \tag{9}$$

   (b) *Second Pruning Stage*: We apply a DCE based pruning procedure [9]. Let us denote $C_{DCE} = \{e_1, \ldots, e_M\}$ the convex polygon obtained from the DCE, where e_k is an edge. Let us denote $\phi(C_{DCE}, v)$ a

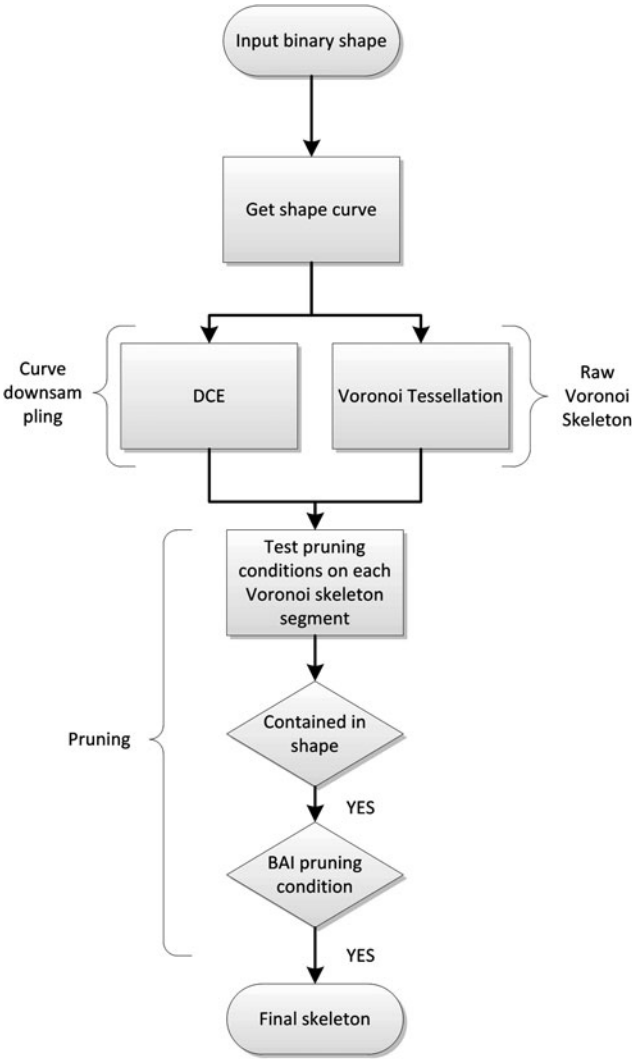

Fig. 1 Skeleton computation visual schema

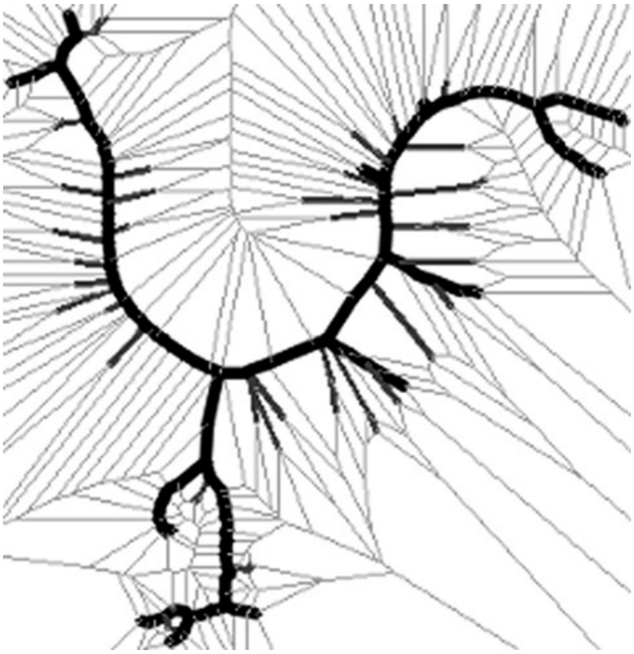

Fig. 2 *Pruning procedure* stages of BERIS of the image in Fig. 3. *Bright gray*: Voronoi segments removed in first pruning stage. *Gray*: Voronoi segments removed after the second pruning stage, DCE pruning. *Black*: final skeleton

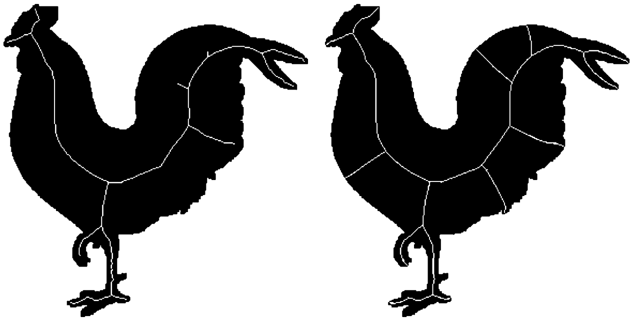

Fig. 3 *Left*: Voronoi based approach presented in this paper. *Right*: Skeleton computed using the Matlab implementation in [50]. The binary image is taken from the MPEG shape database presented in [51]. The image can be obtained from [52]

function that returns the edge in C_{DCE} where the Voronoi site $v \in V$ has been collapsed by DCE. We remove the Voronoi segments s_{ij}, whose generative points (i.e., closest Voronoi sites) v_i and v_j correspond to the same edge in C_{DCE}. The set of removed Voronoi edges is:

$$S_{VDCE} = \left\{s_{i,j} \mid |\phi\left(C_{DCE}, v_i\right) = \phi\left(C_{DCE}, v_j\right)\right\}.$$

Figure 2 shows them as bold gray segments.

5. The final pruned skeleton is given by:

$$S = S_{VA} - S_{VDCE}, \tag{10}$$

shown as the black bold segments in Fig. 2. The resulting skeleton can be seen in Fig. 3.

6.1 Comments

The first pruning stage, step (4.a) of the previous section algorithm, can be performed efficiently because we only need to test the two endpoints of the segment to remove the whole segment, as we prove in Sect. 7 below. This is critical for the real-time realization of the algorithm. The second pruning stage can also be performed efficiently by the definition of the Voronoi edge. Every point in a Voronoi edge shares the same pair of Voronoi sites, say v_i and v_j which are also their generative points. We only need to check the DCE condition once for the whole Voronoi edge. Both requirements of the DCE pruning procedure [9] are met by the Voronoi skeletonization procedure, which are: (1) every skeleton point must be the center of a maximal disk and (2) the generative points in the shape boundary curves must be available.

The most external branches of the pruned skeleton do not connect with the boundary curves, because the first pruning stage shortens the skeleton ending branches as can be appreciated in Fig. 3. This shortening can be viewed as providing additional noise robustness because most external skeleton branches are most easily affected by boundary noise.

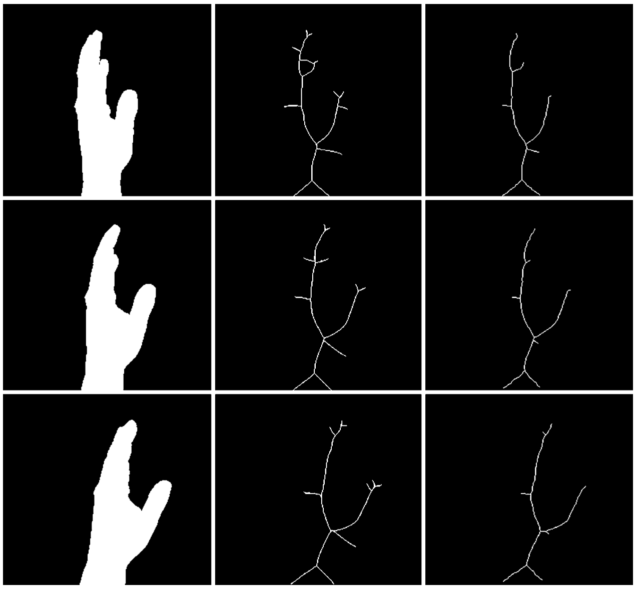

Fig. 4 *Left column*: Hand gesture binary image sequence from [53]. *Middle Column*: Skeletons obtained using the implementation of algorithm [9] provided in [50]. *Right column*: Skeletons obtained using the approach in this paper

This algorithm has two parameters: the specific sampling used to compute the initial Voronoi skeleton and the final number of edges in the DCE polygon. The stability of the algorithm can be appreciated visually in Fig. 4, showing the result of *BERIS* compared to a state-of-the-art algorithm for some hand gesture images that show slight variations. *BERIS* introduces few spurious branches as the boundary shape changes along the gesture.

6.2 Strengths and Weaknesses

The main strengths of the proposed skeletonization algorithm are its stability and real-time performance. As defined before, skeleton computing stability implies that small variations in the boundary of the shape produces small variations of the computed skeleton. Stability can be measured through the number of branches of the skeleton, so that we can give a quantitative evaluation of the changes happening between similar shapes. We have used this approach in the experiments reported below. Another strength of the algorithm is that deals naturally with shapes containing holes.

The main weakness of the algorithm is its dependence on the boundary subsampling resolution. In fact, the pruning algorithm is dependent on this resolution to provide accurate results. In other words, depending on the resolution, some skeleton components that must be pruned may be preserved, introducing some degree of error in the representation. However, at the current subsampling resolutions, such errors are very unlikely or negligible.

7 Voronoi Pruning Theorems

We prove that it is enough to test if both endpoints or Voronoi vertices of a Voronoi edge are inside the shape, to state that the whole edge is contained in the shape. This result is critical for the validity of the first pruning stage. We prove first this statement in the continuous space when $V = C$. Then for a subsampled boundary. We start stating as lemma an auxiliary result:

Lemma 1 *A Voronoi site does not belong to any Voronoi edge.*

$$\forall p \in s_{ij} \wedge \forall v_k \in V \mid \|p - v_k\| > 0. \tag{11}$$

Proof Inmediate from the definition of Voronoi edge. □

The next lemma states that a line segment whose extreme points lie in the shape either does not cross the shape boundary or does cross it an even number of times, meaning that for each boundary point that the line "exits" the shape there must be another point where the line "enters" the shape, going from one extreme point to the other. This lemma is used in the proof of the ensuing theorems.

Lemma 2 *Given a line segment S so that its end points belong to the shape, $\alpha_S \in F$ and $\omega_S \in F$, then C cuts S in an even number of points or never. Let $T = S \cap C$ be the set of cutting points, then*:

$$|T| \mod(2) = 0 \vee |T| = 0. \tag{12}$$

Proof Note that the shape's boundary is a closed curve in 2D space. By the Jordan curve theorem, any continuous path connecting a point of the region inside the curve to a point of the region outside the curve intersects the closed curve somewhere. If the line segment connecting two points inside the shape crosses its boundary, by the Jordan curve theorem there must be a corresponding boundary point whereby the points of the line outside the shape connect with the other end of the line inside the shape. Therefore the lemma is true. □

The next theorem justifies that only looking at the Voronoi edge extreme points we can decide if the whole Voronoi edge lies inside the shape and, therefore, must be preserved in the first pruning step. This result is key for the real-time performance of the algorithm.

Theorem 2 *Let $\alpha_{ij}, \omega_{ij} \in (T \cap s_{ij})$ be the end points of a Voronoi edge s_{ij} (i.e. its Voronoi vertices), with Voronoi sites v_i and v_j. If both end points α_{ij}, ω_{ij} belong to the shape F, then the whole edge s_{ij} is contained in F, when the Voronoi site set is equal to the whole shape boundary curve $V = C$, and the space is continuous. Formally,*

$$\alpha_{ij} \in F \wedge \omega_{ij} \in F \quad \Longrightarrow \quad s_{ij} \subset F. \tag{13}$$

Proof Let us start assuming that there is a point p in a Voronoi edge s_{ij} whose end points fall inside the shape F, but p does not belong to the shape,

$$\alpha_{ij} \in F, \qquad \omega_{ij} \in F, \tag{14}$$

$$\exists p \in s_{ij} \quad \text{s.t.} \quad p \notin F. \tag{15}$$

The proof by contradiction consists in showing that then s_{ij} does not conform with the definition of Voronoi edge. By (14) and (15) we have that,

$$\alpha_{ij}, \omega_{ij} \in F \wedge \exists p \in s_{ij} \wedge p \notin F$$
$$\Longrightarrow \quad \exists q \in \overline{\alpha_{ij} p} \quad \text{s.t.} \quad q \in F, \tag{16}$$

where $\overline{ab}$ denotes a segment, and we will have some subsegment included in the shape:

$$\exists \left[\overline{q'q}\right] \subset \overline{\alpha_{ij} p} \quad \text{s.t.} \quad \left[\overline{q'q}\right] \subset F. \tag{17}$$

From Lemma 2 and $\alpha_{ij}, q \in F$ we have that there is either an even number of boundary curve points or none in the segment $\overline{\alpha_{ij} q}$. We will consider both options.

(a) If there is not any boundary point in the segment $\overline{\alpha_{ij} q}$, i.e. $\overline{\alpha_{ij} q} \cap C = 0$,

$$\overline{\alpha_{ij} q} \subset F \quad \Longrightarrow \quad q \in F. \tag{18}$$

(b) If there is an even number of boundary points in the segment $|\overline{\alpha_{ij} q} \cap C| \mod(2) = 0$

$$\exists \overline{q'q} \subset \overline{\alpha_{ij} p} \quad \text{s.t.} \quad \overline{q'q} \subset F. \tag{19}$$

Finally, since the whole shape boundary curves are used as Voronoi sites $V = C$ and q belongs to the shape boundary curves $q \in C$, then it is a Voronoi site $q \in V$. Moreover $q \in s_{ij}$ implies that it is at zero distance from the Voronoi edge. Consequently, s_{ij} does not fulfill the definition of Voronoi edge. □

Prior to the demonstration of the last result, we must define the uniform sampling of the closed boundary curve as:

$$\forall v_i \in V; \quad \left|v_i - v_{(i+1)}\right|_C = K, \tag{20}$$

where $|.|_C$ is the continuous distance along the boundary curve and $K \in \mathbb{N}$ is a constant, starting from some arbitrary v_0. We donote v_J the last site selected from the contour, that is, $J = length(C)/K$. The distance between v_J and v_0 will be less or equal to K,

$$|v_0 - v_J|_C \leq K. \tag{21}$$

The second and last Theorem defines the limitation of the application of Theorem 2 when the shape boundary is subsampled. Subsampling allows to speed up the skeleton computation, however it does introduce uncertainty on the accuracy of the pruning. Theorem 2 gives a sufficient condition for pruning accuracy after boundary uniform subsampling.

Theorem 3 *Let $A_I, A_F \in (T \cap s_{ij})$ be the end points of a Voronoi edge s_{ij} (i.e. its Voronoi vertices). Let the Voronoi sites be a subsampling of the contour $V \subset C$ according to* (20). *If the minimal euclidean distance between any two Voronoi sites is greater than K, then we can guarantee that if both A_I, A_F belong to the shape F, then the whole s_{ij} is contained in F.*

Proof The proof follows a parallel argument of that in Theorem 2. Start assuming that there is a background point $p \notin F$ belonging to a Voronoi edge s_{ij} whose endpoints belong to the shape $(A_I, A_F \in F)$. The argument begins with (15) as in the previous proof but with a substantial difference: the boundary curve point q does not necessarily belong to V. But, by the definition of the uniformly downsampled boundary curve in (20), and the continuity of the boundary curves, there must be at least a $v_k \in V$ at distance over the curve $|v_k - q|_C$ less than $K/2$ from q, because the minimum distance between two points is the straight line,

$$|a-b|_C \geq \|a-b\|, \tag{22}$$

therefore,

$$\begin{aligned} q \in C \quad &\Longrightarrow \quad \exists v_k \in V_{sites} \\ \text{s.t.} \quad &\|q - v_k\| \leq \left(\frac{K}{2}\right). \end{aligned} \tag{23}$$

Note that $v_k \neq v_i$ and $v_k \neq v_j$, because, by construction, either $A_I \notin F$ or $A_F \notin F$. By (3) and (23) there can not be a Voronoi site v_k closer to any point in the Voronoi edge s_{ij} than v_i and v_j. Therefore, $\|q - v_k\| \leq (\frac{K}{2})$ implies that there is a Voronoi site closer to some point in the Voronoi edge than its two generating sites, because $\|v_i - v_j\| > K$ and, therefore,

$$\begin{aligned} &\forall p \in s_{ij} \left(\|v_i - p\| > \left(\frac{K}{2}\right) \right) \\ &\quad \Longrightarrow \quad \|v_i - q\| > \|q - v_k\|. \end{aligned} \tag{24}$$

Therefore s_{ij} is not a Voronoi edge, contradicting the theorem statement. □

8 Greedy Skeleton Graph Matching

For pattern recognition experiments, we need some skeleton matching algorithm. We apply the same matching algorithm to both compared skeletonization algorithms, so there is no classification algorithm bias in the comparison. We consider a conventional mapping of the skeleton into an undirected graph, which can be cyclic or acyclic, depending on the shape having holes or not, respectively. Skeleton points are categorized into three categories: *end points* of branches, *branch points*, and *joint points*, connecting three or more branches. A graph G_S will be constructed as follows: the skeleton end points and joint points are the graph nodes, and the skeleton branches correspond to the links of the graph, labeled with the length of the branch. Each node contains its image position in normalized polar coordinates and Distance Transform (DT) value. Polar coordinates are given relative to the skeleton centroid normalized to the maximum distance to the skeleton centroid. DT values are normalized to the maximum in the pattern.

The Node Distance Measure is a weighted distance between position difference and distance transform value difference. For two nodes $A = (A_r, A_\theta, A_{dt})$ and $B = (B_r, B_\theta, B_{dt})$, the distance between them is measured as

$$\begin{aligned} |A, B|_G = {}& \alpha \, |A_{dt} - B_{dt}| \\ & + (1-\alpha) \sqrt{A_r^2 + B_r^2 - 2 A_r B_r \cos(A_\theta - B_\theta)} \end{aligned} \tag{25}$$

so that $|A, B|_G \in [0, 1]$.

Consider that we have a set of template graphs $T = \{T_1, \ldots, T_n\}$, each $T_i = (N_{T_i}, E_{T_i})$ and a sample graph $S = (N_S, E_S)$. The graph matching problem consists of finding the most similar graph to S in T. For this purpose, we apply a time-efficient greedy approach that provides suboptimal but useful matches in reasonable time. The same algorithm is applied to all skeletons, so that there is no classification bias. Nodes in both graphs, N_{T_i} and N_s, are sorted by their normalized geometrical position, starting from the nodes closest to the image domain boundaries, visiting the graph nodes towards the center of the image domain. Each node is matched only once to its closest and most similar unmatched node. We have a list M of pairs $(A_k \in N_{T_i}, B_k \in N_S)$ of matched nodes. The similarity between two graphs is computed as

$$\mathcal{S}(T_i, S) = \frac{2 \sum_{k=1}^{|M|} (1 - |A_k, B_k|_G)}{|N_{T_i}| + |N_S|}.$$

9 Results

We have carried out two different sets of experiments to validate our skeleton computation algorithm. The first one aimed at validating the skeleton stability. The second one to evaluate them as a feature vector for shape recognition. We have used two image databases for the experiments. The first image database contains image sequences of hand gestures for the interaction on Tabletops. We have made this image database available at [53]. It contains a set of binary images corresponding to the temporal sequence of three different dynamic hand gestures. These gestures are dynamic in the sense that include motion and pose changes over time, as opposed to a static pose, which can be represented by only one image. Those gestures are: grabbing an object, turning a page and pointing to some object. This database includes 200 repetitions of the three basic gestures, each repetition represented by a sequence of 15 images, up to a total of $200 \times 3 \times 15 = 9000$ images. We will call this database *Tabletop gesture database* henceforth.

The second image database is used for benchmarking shape matching algorithms and includes binary images grouped into categories by their content, not by their appearance or shape. This database was built up according to the MPEG-7 document [51]. It is available at our research group's web page [52]. Some categories include images corresponding to the same concept, but showing noticeable different shapes. Each category includes 20 samples. Shape variations inside a category may include rotation, size and position, and even image resolution variations. Some images have holes in it, while others do not. We will denote this database as *MPEG-7 database* henceforth.

We compare the performance of *BERIS* against a state-of-the-art algorithm [9], (called BAI in the tables below) using the Matlab implementation of the algorithm provided by the authors in [50] to avoid implementation bias.

9.1 Stability Tests

We have carried out the stability tests on the Tabletop gesture database because it contains dynamic gesture sequences of hand poses. Therefore, there are many small variations that may account for boundary noise. We define two measures of the stability of our skeletons. The first one is static: the average number of branches in a gesture image sample. The second one is dynamic: the average branch number difference between consecutive frames in a dynamic gesture image sequence. Consecutive images in the sequence are similar in terms of the Hausdorff distance, so the corresponding skeletons should also be similar. We carried out different experiments with different *DCE pruning values*, namely 15, 20 and 25. The best results were obtained for the most gentle DCE pruning value, $\theta = 25$. The results in Table 1 show that our method (*BERIS*) produces systematically simpler skeletons and with less variability than BAI procedure. In some classes, the average size of *BERIS* is almost half of the skeleton size of BAI procedure. Table 2 shows that our *BERIS* procedure is less sensitive to small changes in the boundary produced by the dynamic variations of hand shape in consecutive images of the same gesture sequence.

Table 1 Static skeleton stability measure: average skeleton branch number per gesture sample

Gesture	BERIS		BAI	
	Average	Std. dev.	Average	Std. dev.
Grab	17.67	3.14	24.55	3.04
Point	12.61	2.52	22.73	3.03
Turn page	14.92	3.45	24.33	4.55
Total	15.07	3.03	23.87	3.54

Table 2 Dynamic skeleton stability measure: average skeleton branch number difference between consecutive images in a gesture sequence

Gesture	BERIS		BAI	
	Average	Std. dev.	Average	Std. dev.
Grab	2.10	1.72	2.79	2.16
Point	1.75	1.47	3.24	2.50
Turn page	2.58	2.20	3.89	3.06
Total	2.15	1.80	3.30	2.57

9.2 Pattern Recognition Experiments

The role of Classification experiments in this paper is as a comparison/validation tool, not as an end in itself. Also we have not tried to optimize the classification results tuning an specific classifier to the data, because we intend to highlight the difference between both skeletonization algorithms. For this reason we have applied simple classifiers trying to avoid introducing bias in the comparison process. We have tested two kind of classification methods using the graph similarity $\mathcal{S}(T_i, S)$ measure as the feature vector. The first one takes some representatives of the class and performs a classification based on the maximum similarity between the remaining test image skeleton graphs and the class representatives. We denote 1-R, 3-R and 5-R the cases when we select randomly one, three and five gesture samples for each gesture as the class representatives. The second classification system is built using Probabilistic Neural Networks (PNN) [54]. We have explored the sensitivity of the systems to the DCE pruning limit θ, the node similarity measure parameter α, and the intrinsic classification system parameters.

9.2.1 Tabletop Gesture Database

In this database, each of the three gestures is composed of 15 frames, therefore we may consider up to 45 image classes. We have performed and reported elsewhere the results with diverse aggregations of these classes. Here we report the results on the classification of each image into its corresponding gesture class.

Table 3 Total recognition results for 3 hand gesture classes

DCE θ	Alpha	Classifier	BAI	BERIS
15	0.25	1-R	72.06	70.74
”	0.25	3-R	80.53	80.21
”	0.25	5-R	83.67	83.81
”	0.50	1-R	78.30	80.73
”	0.50	3-R	87.01	89.17
”	0.50	5-R	89.84	91.96
”	0.75	1-R	82.30	84.78
”	0.75	3-R	90.71	92.27
”	**0.75**	**5-R**	**93.25**	**94.57**
20	0.25	1-R	72.04	70.58
”	0.25	3-R	80.93	79.61
”	0.25	5-R	83.82	82.71
”	0.50	1-R	79.77	80.85
”	0.50	3-R	88.23	88.60
”	0.50	5-R	90.79	91.08
”	0.75	1-R	83.34	84.97
”	0.75	3-R	91.29	92.03
”	0.75	5-R	93.60	94.01
25	0.25	1-R	72.81	71.40
”	0.25	3-R	81.73	79.69
”	0.25	5-R	84.14	82.83
”	0.50	1-R	81.11	81.18
”	0.50	3-R	89.41	88.79
”	0.50	5-R	91.41	91.19
”	0.75	1-R	84.95	85.30
”	0.75	3-R	92.44	92.17
”	0.75	5-R	94.39	94.12

k-R Classification Results We performed a hundred repetitions of the classification with each number of representatives in the k-R experiment. Average classification results are shown in Table 3. The results show that the classification based on the skeletons obtained with our *BERIS* procedure outperforms most of the times the BAI procedure. The best average result is obtained with a DCE pruning limit $\theta = 15$, $\alpha = 0.75$ and a 5-R classification system. One instance of the confusion matrix obtained in one repetition of the classification with these parameters is shown in Table 4. It can be appreciated that the “turn page” and “grab” gestures show the greater confusion. Regarding parameter sensitivity, higher values of the α parameter produce better results. This means that the geometrical position of nodes is more important for matching than their DT value. There is also a systematic trend of improvement as the number of representatives increases. We do not appreciate a significant effect of the DCE pruning limit θ. It seems that higher DCE pruning parameter does not preserve additional discriminating information.

Table 4 Total recognition Confusion Matrix for the best case of BERIS with 3 hand gesture classes: DCE pruning limit $\theta = 15$, $\alpha = 0.75$ and a 5-R classification system

	Grab	Point	Turn page
Grab	**266725**	603	32672
Point	82	**298783**	1135
Turn page	10607	3731	**285662**

Table 5 PNN global recognition results for 3 hand gesture classes

DCE θ	α	σ	BAI	BERIS
15	0.25	0.1	64.67	75.11
”	0.25	2.0	60.50	71.11
”	0.50	0.1	73.72	82.28
”	0.50	2.0	69.17	78.61
”	0.75	0.1	80.11	87.94
”	0.75	2.0	75.17	83.11
20	0.25	0.1	70.56	79.11
”	0.25	2.0	67.50	76.94
”	0.50	0.1	78.06	85.44
”	0.50	2.0	74.00	83.28
”	0.75	0.1	84.89	86.28
”	0.75	2.0	80.78	83.29
25	0.25	0.1	70.78	81.00
”	0.25	2.0	67.94	78.33
”	0.50	0.1	81.28	84.89
”	0.50	2.0	78.22	83.22
”	0.75	0.1	87.61	85.67
”	0.75	2.0	84.67	83.61

PNN and 2-fold Cross-Validation Results The experimental design is as follows: 20 experiments, using a 2-Fold cross-validation where 40 percent of the database was taken as training data. Two values of the PNN’s variance parameter were tested: 0.1 and 2.0. Total (all classes) average recognition results are shown in Table 5. Since our graph distance measure takes values in the range [0, 1], it is natural that results with a variance parameter value 2.0 are worse than with a value 0.1. Overall, these tests agree with the conclusions of the k-R tests. Improvement of the classification using the *BERIS* skeletons over the *BAI* skeletons is more evident than in Table 3, although the results are worse in general.

9.2.2 *MPEG-7 Database*

We have performed 100 classification experiments using 1-R, 3-R and 5-R classifiers. Since the Matlab implementation of the *BAI* algorithm made available by the author does not contemplate holes in the shapes but this database contains images with holes in them, two experimental design scenarios have been proposed. In the first one the whole

Table 6 Total recognition results for the whole MPEG database

DCE θ	α	Classifier	BAI	BERIS
15	0.25	1-R	23.31	22.59
”	0.25	3-R	38.02	36.92
”	0.25	5-R	71.98	74.60
”	0.50	1-R	28.79	28.38
”	0.50	3-R	44.93	45.26
”	0.50	5-R	70.52	70.66
”	0.75	1-R	30.38	30.85
”	0.75	3-R	46.32	48.56
”	0.75	5-R	61.57	59.86
20	0.25	1-R	25.02	23.35
”	0.25	3-R	40.28	37.62
”	0.25	5-R	72.04	69.19
”	0.50	1-R	30.33	29.14
”	0.50	3-R	46.93	46.08
”	0.50	5-R	73.80	73.36
”	0.75	1-R	32.02	32.24
”	0.75	3-R	48.42	48.18
”	0.75	5-R	62.98	59.22
25	0.25	1-R	26.06	24.00
”	0.25	3-R	37.54	34.17
”	0.25	5-R	49.50	56.06
”	0.50	1-R	31.51	29.45
”	0.50	3-R	43.23	40.88
”	0.50	5-R	55.53	53.61
”	0.75	1-R	32.65	31.49
”	0.75	3-R	44.41	43.22
”	0.75	5-R	56.32	45.87

database has been used, but the *BERIS* algorithm ignores the holes as the *BAI* algorithm does. The results for the complete database are shown in Table 6. In the second case, all of classes containing image samples with holes in them have been discarded. The results for this partial database are shown in Table 7.

From the complete database results (Table 6) the best result is obtained on the *BERIS* skeletons for $\theta = 15$, $\alpha = 0.25$ and 5-R classifier. Overall, low values of α give better results. The explanation of this behavior may lie in the fact that there are a lot of rotational variations in the image samples, and the normalization procedure of our graph computation procedure does not include orientation normalization. Therefore, rotated shapes are treated as different shapes by our recognition algorithm. Consequently, in this case the DT value is more discriminant because it is more robust against rotations in the shape. That is, in this database shapes are better identified by their distribution of DT values than by the position of the skeleton singular points (endpoints and joint points). Besides the maximum results, the performance of *BAI* skeletons seems to be comparable or better than the *BERIS* skeletons. When we consider the partial database results in Table 7, the effect of α seems to have reversed compared with the complete database results. Besides, overall results improve and the *BERIS* skeleton obtains higher optimal results.

Table 7 Total recognition results for a reduction of the MPEG database removing image classes containing holes on any of their samples

DCE θ	α	Classifier	BAI	BERIS
15	0.25	1-R	24.95	26.08
”	0.25	3-R	46.07	46.61
”	0.25	5-R	64.21	65.01
”	0.50	1-R	31.29	32.91
”	0.50	3-R	55.35	56.85
”	0.50	5-R	74.88	74.44
”	0.75	1-R	33.78	36.10
”	0.75	3-R	57.90	60.89
”	0.75	5-R	77.50	81.56
20	0.25	1-R	26.61	26.36
”	0.25	3-R	48.03	46.26
”	0.25	5-R	66.07	64.66
”	0.50	1-R	32.99	33.03
”	0.50	3-R	57.29	55.85
”	0.50	5-R	76.70	75.71
”	0.75	1-R	35.05	36.41
”	0.75	3-R	59.60	60.00
”	0.75	5-R	79.15	80.34
25	0.25	1-R	29.51	27.38
”	0.25	3-R	50.84	47.45
”	0.25	5-R	69.14	66.20
”	0.50	1-R	35.63	33.95
”	0.50	3-R	58.99	57.43
”	0.50	5-R	78.16	77.52
”	0.75	1-R	36.91	36.70
”	0.75	3-R	60.45	61.50
”	0.75	5-R	79.79	81.57

10 Conclusions

We have introduced a novel skeleton computation algorithm, combining Voronoi Skeletons, uniform subsampling of the boundary curves and pruning. Some properties of Voronoi skeletons discussed and proved in the paper allow for time-efficient realization of the pruning process. Uniform curve subsampling retains the main structural features reducing computational cost. Pruning adds stability to the resulting skeleton. We perform two pruning stages. One removing edges not completely inside the shape. The second using a

DCE polygonal approximation of the shape boundary. The algorithm has been implemented and tested in real-time.[3] We have reported results on skeleton stability and pattern recognition experiments showing that *BERIS* algorithm outperforms previous state-of-the-art algorithms. Future works may be devoted to implement pattern recognition in real-time for applications such as Tabletop interaction.

References

1. Gonzalez, R., Woods, R.: Digital Image Processing, p. 543. Addison-Wesley/Longman, Reading/Harlow (2001), Chap. 9.5.7
2. Serra, J.: Image Analysis and Mathematical Morphology. Academic Press, San Diego (1982)
3. Rock, I., Linnett, C.M.: Is a perceived shape based on its retinal image? Perception **22**(1), 61–76 (1993)
4. Hoffman, D., Richards, W.A.: Parts of recognition. Cognition **18**, 65–96 (1984)
5. Blum, H.: Biological shape and visual science. Theor. Biol. **38**, 205–287 (1973)
6. He, L., Han, C., Wee, W.: Object recognition and recovery by skeleton graph matching. In: IEEE International Conference on Multimedia and Expo, July 2006, pp. 993–996 (2006)
7. Eede, M.v., Macrini, D., Telea, A., Sminchisescu, C., Dickinson, S.S.: Canonical skeletons for shape matching. In: ICPR'06: Proceedings of the 18th International Conference on Pattern Recognition, pp. 64–69. IEEE Computer Society, Los Alamitos (2006)
8. Goh, W.-B.: Strategies for shape matching using skeletons. Comput. Vis. Image Underst. **110**(3), 326–345 (2008)
9. Bai, X., Latecki, L.J., Liu, W.Y.: Skeleton pruning by contour partitioning with discrete curve evolution. IEEE Trans. Pattern Anal. Mach. Intell. **29**(3), 449–462 (2007)
10. Beristain, A., Graña, M.: A pruning algorithm for Voronoi skeletons. Electron. Lett. **46**(1), 39–41 (2010)
11. Biasotti, S., Attali, D., Boissonnat, J., Edelsbrunner, H., Elber, G., Mortara, M., Sanniti di Baja, G., Spagnuolo, M., Tanase, M., Veltcamp, R.: Skeletal structures. In: Shape Analysis and Structuring, pp. 172–175. Springer, Berlin (2008)
12. Reeb, G.: Sur les points singuliers d'une forme de pfaff completement integrable ou dune fonction numerique. C. R. Acad. Sci. **222**, 847–849 (1946)
13. Shinagawa, Y., Kunii, T.L., Kergosien, Y.L.: Surface coding based on Morse theory. IEEE Comput. Graph. Appl. **11**(5), 66–78 (1991)
14. Blum, H.: A transformation for extracting new descriptors of shape. In: Dunn, W.W. (ed.) Models for the Perception of Speech and Visual Form, pp. 362–380. MIT Press, Cambridge (1967)
15. Bruce, J., Giblin, P.J., Gibson, C.G.: Symmetry sets. Proc. R. Soc. Edinb. **104**(A), 179–204 (1985)
16. Jain, A.K.: Fundamentals of Digital Image Processing, p. 387. Prentice-Hall, New York (1989), Chap. 9.9
17. Mather, J.: Distance from a submanifold in euclidean space. Proc. Symp. Pure Math., Ser. 2 **40**(2), 199–216 (1983)
18. Giblin, P.J., Kimia, B.B.: On the local form and transitions of symmetry sets, medial axes, and shocks. Int. J. Comput. Vis. **54**(1–3), 143–156 (2003)
19. Giblin, P., Kimia, B.B.: A formal classification of 3d medial axis points and their local geometry. IEEE Trans. Pattern Anal. Mach. Intell. **26**(2), 238–251 (2004)
20. Kimia, B.B., Tannenbaum, A.R., Zucker, S.W.: Shapes, shocks, and deformations i: the components of two-dimensional shape and the reaction-diffusion space. Int. J. Comput. Vis. **15**(3), 189–224 (1995)
21. Bai, X., Latecki, L.: Discrete skeleton evolution. In: Energy Minimization Methods in Computer Vision and Pattern Recognition. Lecture Notes in Computer Science, vol. 4679, pp. 362–374. Springer, Berlin (2007)
22. Hesselink, W., Roerdink, J.: Euclidean skeletons of digital image and volume data in linear time by the integer medial axis transform. IEEE Trans. Pattern Anal. Mach. Intell. **30**(12), 2204–2217 (2008)
23. Lam, L., Lee, S., Suen, C.: Thinning methodologies: A comprehensive survey. IEEE Trans. Pattern Anal. Mach. Intell. **14**(9), 869–885 (1992)
24. Siddiqi, K., Pizer, S.: Medial Representations: Mathematics, Algorithms and Applications. Springer, Berlin (2008)
25. Torsello, A.: Matching hierarchical structures for shape recognition. Ph.D. dissertation, University of York, July 2004
26. Giardina, C., Dougherty, E.: Morphological Methods in Image and Signal Processing. Prentice Hall, New York (1988)
27. Jang, B., Chin, R.: Analysis of thinning algorithms using mathematical morphology. IEEE Trans. Pattern Anal. Mach. Intell. **12**(6), 541–551 (1990)
28. Leymarie, F., Levine, M.: Simulating the grassfire transform using an active contour model. IEEE Trans. Pattern Anal. Mach. Intell. **14**(1), 56–75 (1992)
29. Tek, H., Kimia, B.: Symmetry maps of free-form curve segments via wave propagation. In: The Proceedings of the Seventh IEEE International Conference on Computer Vision, 1999, vol. 1, pp. 362–369 (1999)
30. Tari, Z.S.G., Shah, J., Pien, H.: Extraction of shape skeletons from grayscale images. Comput. Vis. Image Underst. **66**(2), 133–146 (1997)
31. Rosenfeld, A., Pfaltz, J.L.: Sequential operations in digital picture processing. J. ACM **13**(4), 471–494 (1966)
32. Rosenfeld, A., Pfaltz, J.: Distance functions on digital pictures. Pattern Recognit. **1**(1), 33–61 (1968)
33. Ogniewicz, R., Ilg, M.: Voronoi skeletons: theory and applications. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Proceedings CVPR'92, pp. 63–69 (1992)
34. Brandt, J.W., Algazi, V.R.: Continuous skeleton computation by Voronoi diagram. CVGIP, Image Underst. **55**(3), 329–338 (1992)
35. Choi, S.W., Seidel, H.-P.: Linear onesided stability of mat for weakly injective 3d domain. In: SMA'02: Proceedings of the Seventh ACM Symposium on Solid Modeling and Applications, pp. 344–355. ACM, New York (2002)
36. Attali, D., Lachaud, J.O.: Delaunay conforming iso-surface, skeleton extraction and noise removal. Comput. Geom., Theory Appl. **19**, 175–189 (2001)
37. Dey, T.K., Zhao, W.: Approximate medial axis as a Voronoi subcomplex. In: SMA'02: Proceedings of the Seventh ACM Symposium on Solid Modeling and Applications, pp. 356–366. ACM, New York (2002)
38. Foskey, M., Lin, M.C., Manocha, D.: Efficient computation of a simplified medial axis. In: SM'03: Proceedings of the Eighth ACM Symposium on Solid Modeling and Applications, pp. 96–107. ACM, New York (2003)
39. Mokhtarian, F., Mackworth, A.K.: A theory of multiscale, curvature-based shape representation for planar curves. IEEE Trans. Pattern Anal. Mach. Intell. **14**(8), 789–805 (1992)
40. Shaked, D., Bruckstein, A.: Pruning medial axes. Comput. Vis. Image Underst. **69**(2), 156–169 (1998)
41. Wets, R.J.-B., Tyrrell Rockafellar, R., Wets, M.: Set convergence. In: Variational Analysis, p. 117. Springer, Berlin (2009)

[3] http://www.ehu.es/ccwintco/index.php/2009-10-24-video-skeleton

42. Brandt, J.W.: Convergence and continuity criteria for discrete approximations of the continuous planar skeleton. CVGIP, Image Underst. **59**(1), 116–124 (1994)
43. Schmitt, M.: Some examples of algorithms analysis in computational geometry by means of mathematical morphology techniques. In: Geometry and Robotics, vol. 391, pp. 225–246. Springer, Berlin (1989)
44. Aronov, B.: A lower bound on Voronoi diagram complexity. Inf. Process. Lett. **83**(4), 183–185 (2002)
45. Attali, D.: r-regular shape reconstruction from unorganized points. In: SCG'97: Proceedings of the Thirteenth Annual Symposium on Computational Geometry, pp. 248–253. ACM, New York (1997)
46. Chazal, F., Lieutier, A.: The "$lambda$-medial axis". Graph. Models **67**(4), 304–331 (2005)
47. Latecki, L.J., Lakämper, R.: Convexity rule for shape decomposition based on discrete contour evolution. Comput. Vis. Image Underst. **73**(3), 441–454 (1999)
48. Latecki, L.J., Lakämper, R.: Shape similarity measure based on correspondence of visual parts. IEEE Trans. Pattern Anal. Mach. Intell. **22**(10), 1185–1190 (2000)
49. Latecki, L.J., Lakämper, R.: Application of planar shape comparison to object retrieval in image databases. Pattern Recognit. **35**, 15–29 (2002)
50. Available: http://www.cis.temple.edu/~latecki/Programs/skeletonPruning07.htm
51. Jeannin, S., Bober, M.: Description of core experiments for MPEG-7 motion/shape, MPEG-7 Std. (1999)
52. Beristain, A.: Conjunto de datos de test mpeg-7 core experiment ce-shape-1 (2009). Available: http://www.ehu.es/ccwintco/index.php/GIC-experimental-databases
53. Beristain, A.: Colección de imágenes para entrenamiento de un reconocedor de gestos de interacción con un tabletop (2009). Available: http://www.ehu.es/ccwintco/index.php/Synthetic_hand_gesture_images_for_tabletop_interaction
54. Specht, D.F.: Probabilistic neural networks. Neural Netw. **3**(1), 109–118 (1990)

Andoni Beristain received the M.Sc. degree in Computer Science in 2005 at The University of The Basque Country (UPV/EHU) and the Ph.D. degree in Computer Science in 2009 from The University of The Basque Country (UPV/EHU).
Currently, he works as researcher at Vicomtech-IK4 (Visual Interaction and Communication Technologies Centre). His current research interests, in the broad sense, include natural user interfaces, medical imaging and real-time image processing.

Manuel Graña received the M.Sc. degree in Computer Science in 1982 and the Ph.D. degree in Computer Sciences in 1989, both from Universidad del Pais Vasco, Spain. His current position is Full Professor (Catedrático de Universidad) in the Computer Science and Artificial Intelligence Department of the Universidad del Pais Vasco, in San Sebastian. He is the head of the Computational Intelligence Group (Grupo de Inteligencia Computational) which has been awarded funding as a high performance university research group since the year 2001.
Current research interests are in applications of Computational Intelligence to multicomponent robotic systems, specifically Linked Multicomponent Robotic Systems, medical image in the neurosciences, multimodal human computer interaction, remote sensing image processing, content based image retrieval, Lattice Computing, semantic modelling, data processing, classification and data mining.

Ana I. Gonzalez received her Computer Degree in 1993 at the UPV/EHU.
Currently, she works as a collaborator professor of computer science at the UPV/EHU. She is doing her Ph.D. on neural networks. She is part of Computational Intelligence Group. Has published in Neurocomputing, IEEE NN, LNCS, and IEEE Press conference proceedings. Her research interests cover self-organizing maps, GNC, clustering, neural networks, continuation methods.