THE ELLIPSOIDAL SKELETON IN MEDICAL APPLICATIONS

F. BANEGAS, M. JAEGER (CIRAD, Montpellier, France)
D. MICHELUCCI, M. ROELENS (MINES St Etienne, France)

1 INTRODUCTION

MOTIVATION

Medical imaging faces increasingly huge 3D data images. Some paradigm is required to

- interactively visualize huge amount of data images (CT-scan, MRI)
- reconstruct surfaces of organs
- extract relevant features for diagnosis or surgical interventions
- automatic match objects extracted from 3D images

OVERVIEW

The paradigm used here is the **ellipsoidal skeleton**.

It is a **tree of the best partitions** of the set of 3D points extracted from segmented 3D data images.

The **best partition** in k classes ($k \in 1, 2 \ldots 10$) is the one which maximizes the homogeneity of classes, and thus the differences between classes.

A k-partition permits to **reconstruct the surface** : approximate each class by some geometric primitive (the best ellipsoid or another implicit surface) before merging a la Blinn.

Varying k gives multiple **levels of details**.

The E-skeleton is invariant through isometries, and steady against discretization noise, under-, over-sampling. This **steadyness** is used for automatic matching between shapes, computing distances between shapes, ie **recognition**.

2 BEST k-PARTITION

USING "INTRINSIC" PARTITIONS

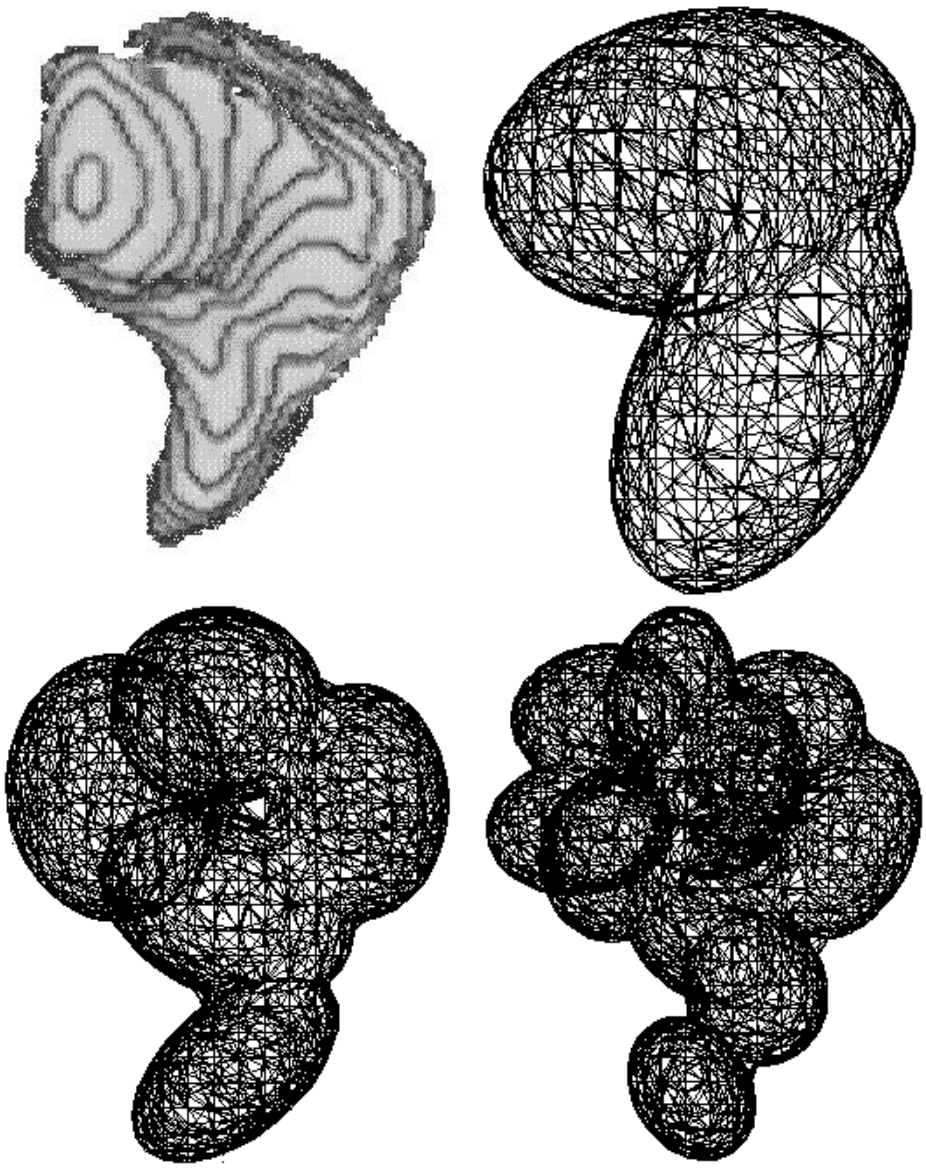

Figure 1: A carpal bone, the hamatum: the cloud, 2-, 5-, and 10-partition.

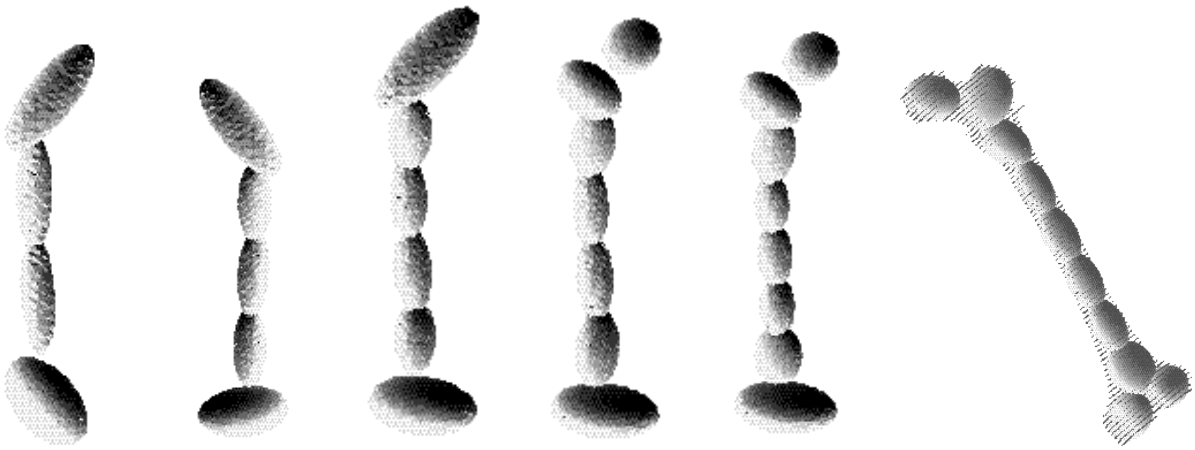

Figure 2: A femur. 4-, 5-, 6-, 7-, 8-, 10- partitions.

EXPECTATION, VARIANCE, COVARIANCE

Let $C = \text{cloud}\,(\Omega, P)$ be a cloud of n points, with weights $\Omega \in \mathbb{R}^n$ and coordinates $P = (X \in \mathbb{R}^n, Y \in \mathbb{R}^n, Z \in \mathbb{R}^n)$. (often all $\Omega[i]$ are equal).

The X expectation:

$$\mathrm{E}\,(X) \stackrel{\text{def}}{=} \frac{\sum_1^n \Omega[i] \times X[i]}{\sum_1^n \Omega[i]}$$

The covariance of X and Y:

$$\begin{aligned} \mathrm{C}\,(X, Y) &\stackrel{\text{def}}{=} \mathrm{E}\,(XY) - \mathrm{E}\,(X) \times \mathrm{E}\,(Y) \\ &= \mathrm{C}\,(Y, X) \end{aligned}$$

where XY stands for $[\, X[1] \times Y[1], \ldots X[n] \times Y[n] \,]$.

The variance of X is:

$$\begin{aligned} \mathrm{V}\,(X) &\stackrel{\text{def}}{=} \mathrm{C}\,(X, X) \\ &= \mathrm{E}\left(X^2\right) - (\mathrm{E}\,(X))^2 \end{aligned}$$

It is also **the expectation of the squares of the differences between X and $\mathbf{E}\,(X)$:**

$$\mathrm{V}\,(X) = \mathrm{E}\left(\, (X - \mathrm{E}\,(X))^2 \,\right)$$

THE EUCLIDEAN VARIANCE OF C IS:

$$\mathrm{V}(C) \stackrel{\mathrm{def}}{=} \mathrm{V}(X) + \mathrm{V}(Y) + \mathrm{V}(Z)$$

Actually it is also **the expectation of the squares of the distances** between points P and their gravity center $(\overline{X}, \overline{Y}, \overline{Z}) = (\mathrm{E}(X), \mathrm{E}(Y), \mathrm{E}(Z))$:

$$\begin{aligned}
&\mathrm{E}\left((X-\overline{X})^2 + (Y-\overline{Y})^2 + (Z-\overline{Z})^2\right) \\
&= \mathrm{E}\left((X-\overline{X})^2\right) + \mathrm{E}\left((Y-\overline{Y})^2\right) + \mathrm{E}\left((Z-\overline{Z})^2\right) \\
&= \mathrm{V}(X) + \mathrm{V}(Y) + \mathrm{V}(Z) = \mathrm{V}(C)
\end{aligned}$$

thus, since distances are invariant through isometries (translations, rotations, symmetries and their composition), the euclidean variance $\mathrm{V}(C)$ as well.

THE COVARIANCE MATRIX M OF THE CLOUD C IS:

$$M \stackrel{\text{def}}{=} \begin{pmatrix} \mathrm{V}(X) & \mathrm{C}(X,Y) & \mathrm{C}(X,Z) \\ \mathrm{C}(Y,X) & \mathrm{V}(Y) & \mathrm{C}(Y,Z) \\ \mathrm{C}(Z,X) & \mathrm{C}(Z,Y) & \mathrm{V}(Z) \end{pmatrix}$$

M is symmetric and positive, and definite in the generic case. Thus eigenvalues are real and positive. $\lambda_1 \geq \lambda_2 \geq \lambda_3 > 0$

The eigenvectors are orthogonal, and called: main axis of inertia. The coordinate system they define is unique (up to orientations) and has a strong physical meaning.

Nota: M has same eigenvectors as the matrix of inertia.

DEFINITION OF BEST PARTITIONS

If $\{C_1, \ldots C_k\}$ is a k-partition of the cloud C, the euclidean variance $\mathrm{V}(C)$ of C is decomposed into two parts:

- the intraclass variance, *ie* the sum of all euclidean variances of classes $C_1, \ldots C_k$
- the interclass variance $\mathrm{V}(\{C_1, C_2 \ldots C_k\})$:

$$\mathrm{V}(\{C_1, C_2 \ldots C_k\}) \stackrel{\text{def}}{=} \mathrm{V}(C) - \sum_{i=1}^{k} \mathrm{V}(C_i)$$

The best k-partition is the one which maximizes the interclass variance (the differences between classes), and thus minimizes the intraclass variance.

The **Progressive Dynamic Clustering** algorithm finds the best partition.

DYNAMIC CLUSTERING ALGORITHM (DCA)

It computes the best k-partition as follows:

Choose an initial k-partition
Repeat
    For each class C_i, $i = 1 \ldots k$
        Compute G_i the center of gravity of C_i
        Empty the class C_i
    EndForEach
    For each point p_j
        Add p_j to class C_i where G_i is closest to p_j
    EndForEach
Until neither of G_i change

This algorithm is guaranteed to find the optimal solution; oscillations due to degeneracies or inaccuracies may happen in some case but are easily avoided.

THE PROGRESSIVE DCA

The DCA works well but is slow. To speed-up, use the Progressive DCA:

The best 1-partition is known.

From the best k-partition, find the best $k+1$-partition:

Virtually split each class C_i by the plane through its gravity center, and orthogonal to the major main axis; then use DCA to split C_i in 2 sub-classes C_i', C_i'', and compute

$$r_i \stackrel{\text{def}}{=} \frac{\mathrm{V}(C_i') + \mathrm{V}(C_i'')}{\mathrm{V}(C_i)}$$

The initial $k+1$-partition is obtained by spliting class C_i with smallest r_i, then perform DCA on the whole cloud.

Other improvement: compute the best k-partition with an under-sampled cloud, using the underlying grid structure of the 3D data image (it divides n by 27).

In practice, time is $O(k^2 n)$, n points, $k \approx 10$ classes.

MAHALANOBIS VARIANT

To characterize and recognize shapes up to affine transforms:

Apply a scaling on the points of the cloud such that the inertia ellipsoid of the whole cloud becomes a sphere with radius 1, then perform the progressive DCA as usual.

It is equivalent to using Mahalanobis distance.

This way two shapes differing by a scaling will have isomorphic E-skeletons.

However, in medical applications, it is not so relevant: maturation and growth is not just a scaling. It is better to sample ages.

THE ELLIPSOIDAL SKELETON

A tree keeps track of the subdivision process. Each node or leaf represents a class. Level k (1 at root) countains k nodes and represent the best k-partition.

Each node or leaf stores also: the gravity center, weigth, number of points, covariance matrix, eigenvalues, eigenvectors, α, u...

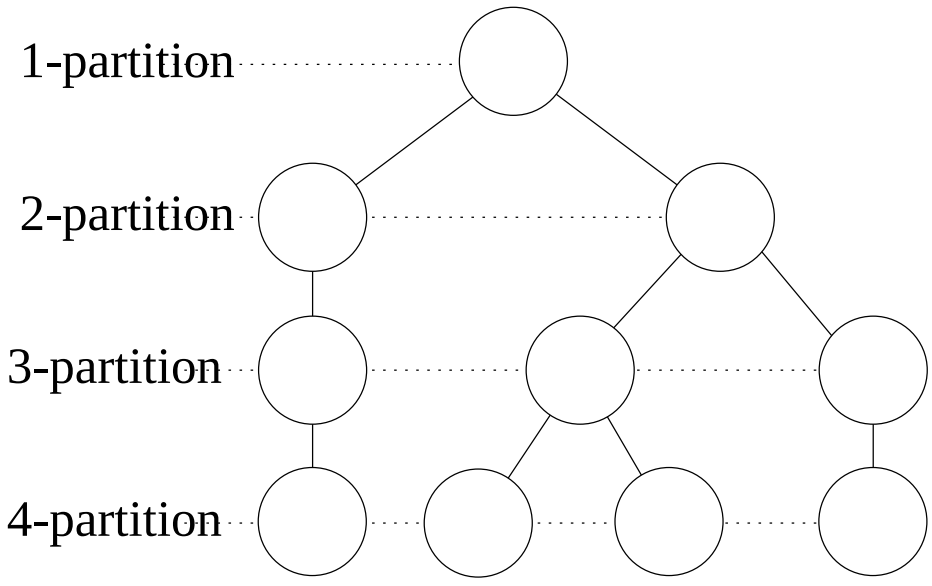

Figure 3: A typical tree of an ellipsoidal skeleton.

3 RECONSTRUCTION OF THE SURFACE

SOME PICTURES

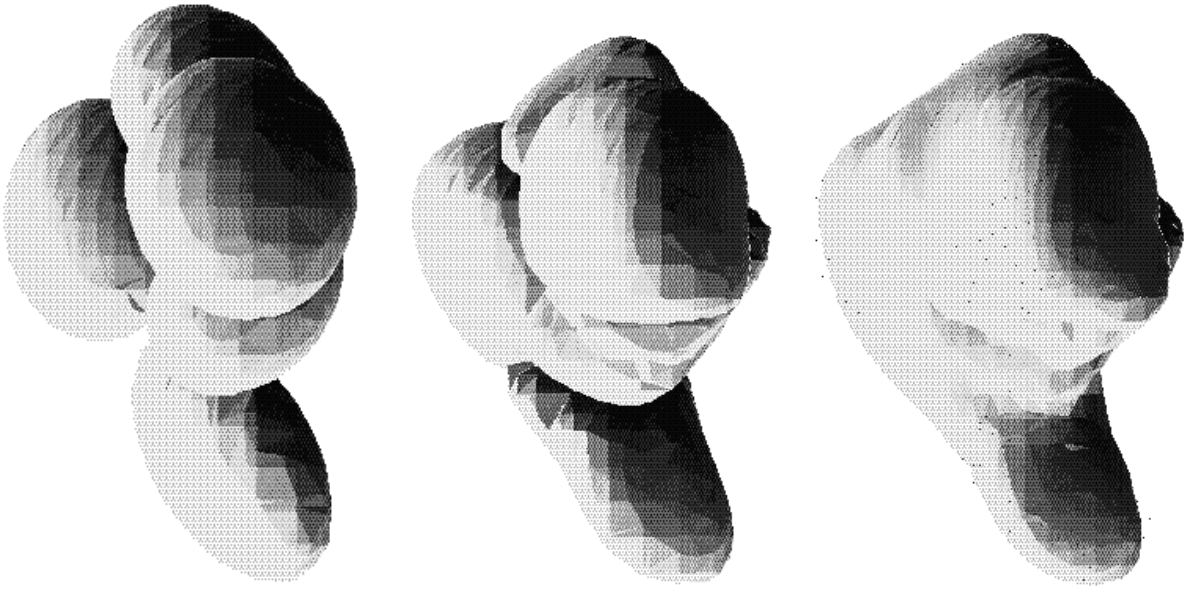

Figure 4: Reconstruction of hamatum: ellipsoids, deformed ellipsoids, blended deformed ellipsoids.

Figure 5: Deformation of ellipoids for the 4-partition of the hamatum.

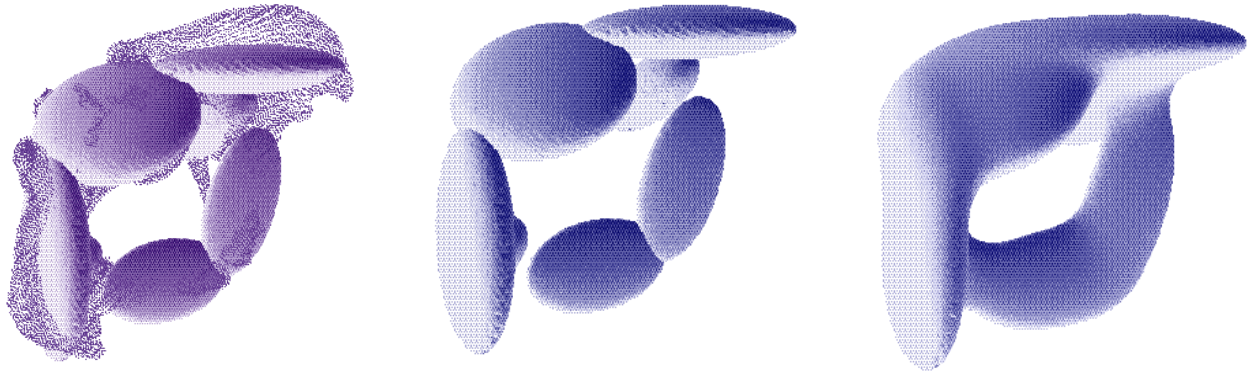

Figure 6: Reconstruction of an iliac bone: cloud and ellipsoids, ellipsoids, merged undeformed ellipsoids.

The E-skeleton, including the Blinn's surface parameters (the heaviest one), is at least 1 thousand more compact than the initial 3D data image. It is about one hundred times more compact than a medium-quality meshing of the initial 3D

Figure 7: Reconstruction of the eight carpal bones: each bone is separately blended.

data image.

SCHEMATIC RECONSTRUCTION

- either represent each class by its **best ellipsoid**, and **union** them.
- or, a bit better, **blend** the k best ellipsoids with a Blinn implicit surface.

BEST ELLIPSOID OF A CLASS

The best ellipsoid for a class C has center C gravity center, and axis the main axis of C. The radius are: $\gamma\sqrt{\lambda_i}, i = 1, 2, 3$ with 2 choices for the scaling constant γ:

- $\gamma = \sqrt{d+2} = \sqrt{5} = 2.236068...$: if C is really an ellipsoid, then the best ellipsoid is equal to C.
- $\gamma = 2$ for all d, because of the big numbers law: if random X follows a normal law, then $\mathrm{E}(X) \pm 2\sqrt{\mathrm{V}(X)}$ countains 90% samples.

BLINN'S BLENDING SURFACES

The surface is an iso-potential one : $P(x, y, z) - 1 = 0$

The potential $P(x, y, z)$ is the sum $\sum_{i=1}^{k} P_i(x, y, z)$ of all k potentials P_i emitted by each geometric primitive, one for each class.

$$P_i(x, y, z) \stackrel{\text{def}}{=} e^{-\alpha_i d_i(x,y,z)}$$

where $d_i(x, y, z)$ is 0 at the boundary of the geometric primitive, negative inside, positive outside.

For an ellipsoid,

$$d_i(x, y, z) = \frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} - 1$$

in its natural coordinate system. Express it in world coordinates.

DEFORMED ELLIPSOIDS

To best fit the surface, we deform each best ellipsoid with a modal deformation D_u, following Barr: affine transform + non linear transform: twist, bend, etc

For each primitive, the best (or at least a good value) $u \in \mathbb{R}^{30}$, to minimize the mean gap between boundary points and the surface, is computed with a tabu search (a variant of simulated annealing).

Other tried methods: descent, homotopy, Levenberg-Marquardt.

A point is on the boundary if one of its 6 neighbors (in the underlying grid of the initial 3D data image) does not belong to the same class. With under-sampling for speeding up dynamic clustering, it is the only place where the grid structure is used.

The best α_i for blending are also found with a tabu search. However $\alpha_i = 1$ is simple and often good enough.

IMPLICIT ALGEBRAIC SURFACE

(Not implemented) The geometric primitive for a class is an algebraic implicit surface, with even degree to ensure boundedness:

$$F(x,y,z) = \sum_{0 \leq i+j+k \leq 2d} c_{i,j,k} x^i y^j z^k = 0$$

Assume wlog $c_{0,0,0} = 1$.

For $p = (p_x, p_y, p_z)$ a given point, $F(p)$ is then a computable linear expression in $C = [c_{i,j,k}]$: $F(p) = 1 + C.l$, where l is the vector $[p_x^i \times p_y^j \times p_z^k]$ (in the same order than the $c_{i,j,k}$ in the C vector); if p is a boundary point of the class, we want $F(p) = 1 + C.l$ as small as possible. Expressing this constraint for all boundary points p, we have to minimize the euclidean norm of $[1, \ldots 1] + CL$.

It is a least square problem, whose solution is $C = [-1, \ldots -1]L^+$, where L^+ is the pseudo inverse of L: $L^+ = L^t(LL^t)^{-1}$. See fast methods in [Numerical recipes in C].

Nota: don't use the canonical basis, but Bernstein basis.

LEVEL OF DETAILS

Various k give several level of details for the reconstructed surface.

For fast display, marching tetrahedra tesselate the surface on fly when needed. We don't have to evaluate the potential function at all grid vertices, thus meshing requires always less than 1 second.

The step size s for marching tetrahedra is another parameter to monitor the LoD.

Do we deform or not, do we blend or not are two other parameters.

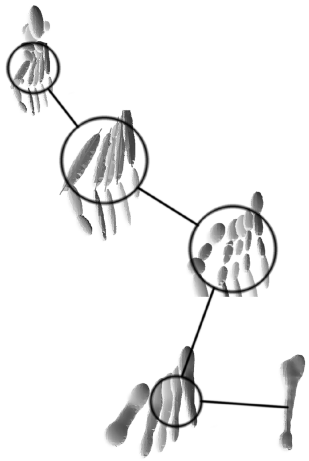

Figure 8: Dynamic rendering for visualization and exploration on tarsal bones.

4 MATCHING, RECOGNITION

MATCHING TWO k-PARTITIONS

The distance between 2 classes is the euclidean distance between their signature points.

The signature point for a class C_i in a cloud C is:

$$(\sqrt{\lambda_{i,1}}, \sqrt{\lambda_{i,2}}, \sqrt{\lambda_{i,3}}, \rho_i)$$

with ρ_i the distance between C_i and C centers.

It is possible to also use the cosines of the angle between the main axis of C_i and C.

Let $C = C_1 \cup C_2 \ldots \cup C_k$ and $C' = C'_1 \cup C'_2 \ldots \cup C'_k$.

The distance between C and C' is:

$$\text{dist}(C, C') = \min \text{dist}(C_1, \sigma(C_1)) + \ldots + \text{dist}(C_k, \sigma(C_k))$$

where σ describes the $k!$ bijections from $\{C_1, C_2 \ldots C_k\}$ to $\{C'_1, C'_2 \ldots C'_k\}$

Finding the σ which minimizes

$$\text{dist}(C_1, \sigma(C_1)) + \ldots + \text{dist}(C_k, \sigma(C_k))$$

is a classical minimal matching problem (and also a max cost min flow problem), solvable in $O(k^2 + \log k)$: in the matrix $D_{ij} = \text{dist}(C_i, C'_j)$, find k entries, one per line and one per column, with minimal sum.

RECOGNITION OF AN OBJECT

If the size is not relevant, normalize...

Recognition of C is finding the closest shape in a library $S_1, \ldots S_n$ of shapes.

All shapes are described by their E-skeletons.

First compare 1-partitions of C and of: $S_1, \ldots S_n$, discard the 90% S_i farthest from C. (When n is very large, some range searching scheme is useful, e.g. some kd-tree).

Compare 2-partitions of C and of remaining shapes in $S_1, \ldots S_n$, discard the 90% S_i farthest. Etc, until it remains about 10 S_i: compare using the maximal k. The shape S_j with smaller distance to C is the closest.

ABOUT CHIRALITY

Our recognition procedure confuses left and right, because the used signature points are invariant by symmetry.

It permits the software to match the left and right teeth of upper and lower jaws, and to detect a pathology on an incisive (left and right incisives were successfully matched, but at a greater distance due to some real pathology). Note: upper and lower teeth are different and cannot be confused...

If this feature is a bug, it is possible to define signature discriminating left and right (see paper).

EXAMPLE OF RECOGNITION

	H1	H2	H3	H4	H5	C1	C2	C3	C4	C5	L1	L2	L3	L4	L5
H1	0	8	3	3	3	67	65	64	63	65	54	53	52	53	53
H2	8	0	8	7	7	69	67	66	66	67	54	53	52	54	53
H3	3	8	0	4	4	67	65	64	63	65	53	52	52	52	52
H4	3	7	4	0	1	68	66	65	64	66	53	53	51	53	53
H5	3	7	4	1	0	68	66	65	65	66	53	53	52	53	53
C1	67	69	67	68	68	0	4	5	6	3	68	66	68	68	66
C2	65	67	65	66	66	4	0	6	7	3	66	64	65	65	64
C3	64	66	64	65	65	5	6	0	3	5	68	66	68	67	66
C4	63	66	63	64	65	6	7	3	0	6	67	65	66	66	65
C5	65	67	65	66	66	3	3	5	6	0	67	65	66	66	65
L1	54	54	53	53	53	68	66	68	67	67	0	5	2	3	4
L2	53	53	52	53	53	66	64	66	65	65	5	0	4	3	1
L3	52	52	52	51	52	68	65	68	66	66	2	4	0	2	3
L4	53	54	52	53	53	68	65	67	66	66	3	3	2	0	2
L5	53	53	52	53	53	66	64	66	65	65	4	1	3	2	0

Part of a distance matrix for hamatum, capitatum and lunatum and 5 other bones of 5 patients: bones are correctly classified: hamatums are closer than other bones, etc.

An analysis editor was developped to find correlations between organs (allometries), and between organs and other criteria (gender, age, genetic parameter).

For example, with this tool, Dr Canovas was able to find relevant correlations between hamulus (some protuberance in the hamatum) maturation and gender.

5 CONCLUSION

A GENERIC DATA STRUCTURE

The E-skeleton model presented was used in a plant modeler, for growth simulation and energy transfer computation.

The E-skeleton seems to be a promising data structure, which exhibits:

- compacity
- surface reconstruction
- multi level representation and real time visualization
- automatic matching and recognition

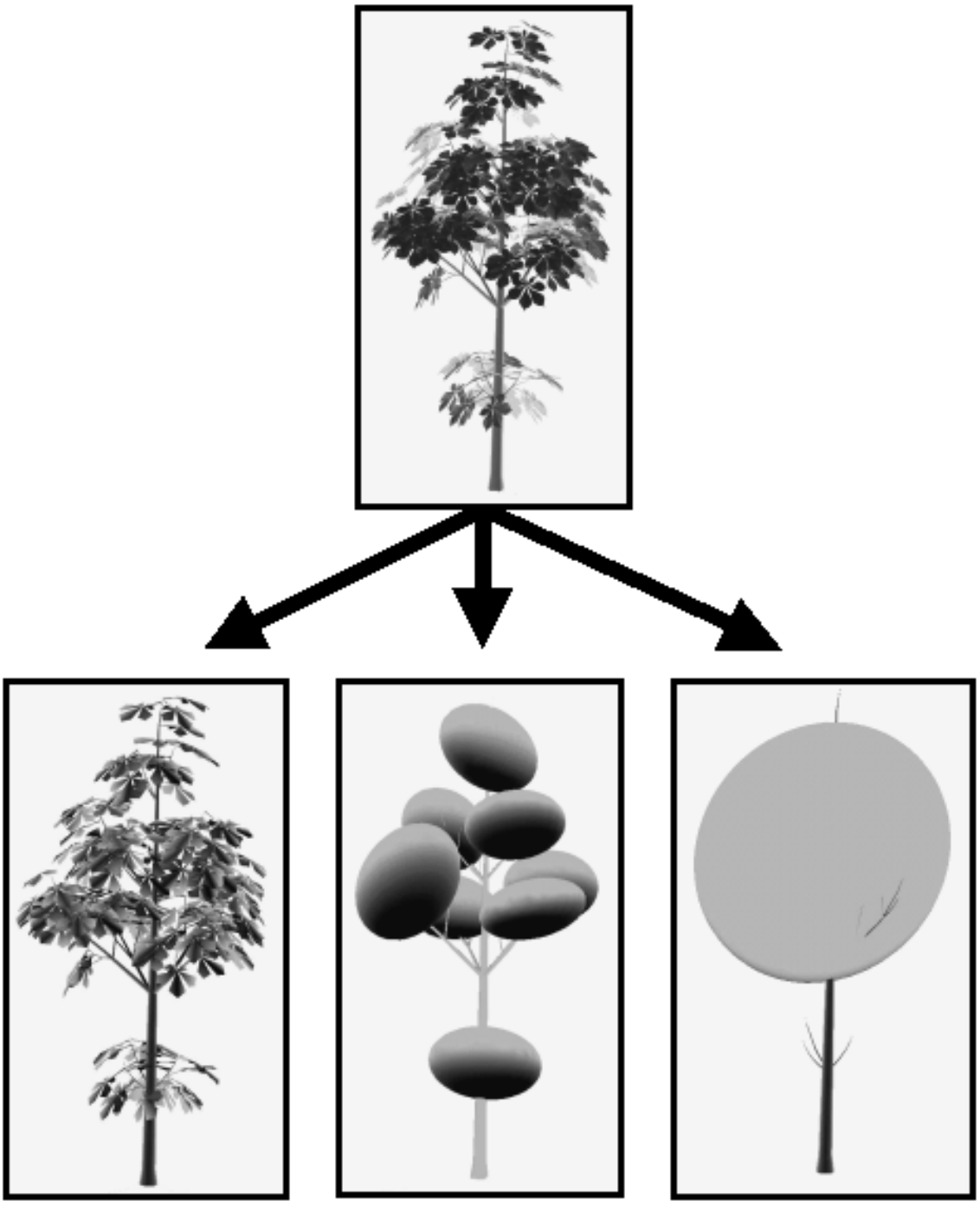

Figure 9: Multi-level representation of a tree.

FURTHER WORKS

- extension of geometric primitives for surface reconstruction;
- improvement of the E-skeleton by inserting a spring-mass pair for each ellipsoid or primitive, allowing extensions towards dynamic models or physically-based models;
- automatic detection of correlations or allometries inside data;
- automatic allometry-based synthesis of organs;
- construction of an atlas of anatomical shapes at several ages, including shapes with pathologies to be recognized;
- applications in CAD, like feature-based recognition and modelling;
- use of feature recognition to help segmentation of original images.
- etc