EUROGRAPHICS 2007 / D. Cohen-Or and P. Slavík
(Guest Editors)

Volume 26 (2007), Number 3

Variational 3D Shape Segmentation for Bounding Volume Computation

Lin Lu[1], Yi-King Choi[1], Wenping Wang[1] and Myung-Soo Kim[2]

[1]Department of Computer Science, The University of Hong Kong, Hong Kong, China
[2]School of Computer Science and Engineering, Seoul National University, Seoul, South Korea

Abstract

We propose a variational approach to computing an optimal segmentation of a 3D shape for computing a union of tight bounding volumes. Based on an affine invariant measure of e-tightness, the resemblance to ellipsoid, a novel functional is formulated that governs an optimization process to obtain a partition with multiple components. Refinement of segmentation is driven by application-specific error measures, so that the final bounding volume meets pre-specified user requirement. We present examples to demonstrate the effectiveness of our method and show that it works well for computing ellipsoidal bounding volumes as well as oriented bounding boxes.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling

1. Introduction

Complex objects are often approximated by simple primitives so as to facilitate efficient geometric computations. For this purpose, bounding volumes have been used very successfully for various applications, including ray tracing, rendering, collision detection and robust transmission of geometric data [Bou85, BK02, JTT01, KV05] (Figure 1). There are efficient intersection tests or proximity computations for commonly used bounding volumes, such as spheres [Hub96], axis-aligned bounding boxes (AABBs) [HKM95], oriented bounding boxes (OBBs) [BCG*96, GLM96], discrete-oriented polytopes (k-DOPs) [KHM*98] and ellipsoids [RB97, WCC*04].

The efficiency of a bounding volume for a given object is often defined as its bounding tightness to the object. To compute a proper decomposition of an object into components so that each can be bounded tightly by a bounding volume is a difficult task. Existing works on computing bounding volumes mainly use hierarchical subdivision and enclose each component with a bounding primitive. The conventional top-down hierarchical approach, although fast, is local and greedy, since there has been no consideration of an optimization formulation and therefore the previous methods do not allow dynamic updates of different components in an optimal manner. Consequently there is still much room for improvement in bounding tightness. A notable exception is the variational approach [WZS*06] to computing sphere bounding volumes; but it is difficult to extend the result to other bounding volumes.

Figure 1: *The dinopet with its bounding volume (33 ellipsoids) automatically generated by our algorithm.*

A variational approach to computing a partition of an object often involves data clustering based on Lloyd it-

eration [Llo82], a popular heuristic for k-means clustering [KMN*02]. The most commonly used metric in this framework is the Euclidean metric, which leads to the standard functional defining the Centroidal Voronoi Tessellation (CVT) [DFG99]. The Euclidean metric is isotropic and therefore cannot capture components with elongated shapes. As we will see in detail at the end of Section 4, the direct use of the anisotropic Mahalanobis (or elliptic) metric [GG89] is problematic, due to its lack of variational foundation. In this paper, we propose a fix to this problem by formulating a new anisotropic metric based on an affine invariant concept of e-tightness, the resemblance to ellipsoid. The new metric is more suitable to the problem of computing an optimal segmentation of a 3D object for the purpose of computing a tight bounding volume. (We caution that such a segmentation is not a generally "meaningful" segmentation, so the segmentation result may not be completely suitable for other purposes.)

The main contributions of our work can be summarized as follows:

1. **A new variational formulation is proposed based on the e-tightness**, a function we introduce to measure the resemblance of a set to an ellipsoid. This functional is used to determine an optimal partition of a 3D solid by minimizing a weighted average of the e-tightness functions of all components. We present **theoretical justification of the proposed anisotropic metric.** Moreover, we present effective computation schemes, and experimental support to show the advantage of this new formulation.
2. Integrating the above optimization method with initialization based on skeleton information and error-driven refinement of partition, we have devised a complete and robust algorithm for computing **a tight bounding volume composed of the union of ellipsoids or oriented bounding boxes** for complex articulated 3D models, as often used in computer animation.

2. Previous work

Shape approximation/segmentation. Vast amount of research work has been conducted on shape approximation or decomposition, for a wide range of applications, including object recognition and geometry processing. Many 'greedy' approaches based on local search have been proposed, such as hierarchical subdivision and region growing, which do not accommodate dynamic incremental updates of a partition in a global and optimal manner. Data clustering techniques, e.g., fuzzy clustering [KT03], have also been applied directly for shape segmentation. The variational approach has been adapted in [CSAD04], which computes a piecewise planar approximation of a surface by minimizing a functional characterizing geometric errors. Note that these methods are only for surface decomposition or approximation.

Bischoff and Kobbelt used ellipsoids to cover the interior volume of an object [BK02]. The decomposition, originally designed for surface reconstruction in robust geometry transmission, contains a larger number of ellipsoids than necessary for tight bounding. Simari and Singh achieved ellipsoidal representation of mesh surfaces using the Lloyd method with a combination of metrics that considers Euclidean radial distance, surface normals and curvatures [SS05]. They also introduced a volume metric to obtain ellipsoids approximating a 3D shape. Kalaiah and Varshney proposed the use of k-means clustering with the Mahalanobis distance for building a hierarchical Principle Component Analysis (PCA) based representation for a point set [KV05].

Bounding volume computation. Several geometric primitives are commonly used as bounding volumes. The axis-aligned bounding box (AABB) [HKM95] for a given object is easy to construct, but it does not fit tightly for many objects and it has to be recomputed if the orientation of the object is changed. The oriented bounding box (OBB) [BCG*96, GLM96] can fit an object more tightly and the class of OBBs is closed under any Euclidean transformation. A hierarchy of OBBs, called OBB-tree or box-tree, is often used to facilitate fast collision query. The splitting of a parent OBB into two smaller components is done by bisecting in the eigen-direction associated with the largest eigenvalue of PCA.

A method for building a hierarchy of spheres, called a sphere tree, is based on mid-axis surface computation [Hub96]. As an extension to this method, the adaptive medial axis approximation (AMAA) [BO04] improves the bounding efficiency by refining the segmentation iteratively with a greedy approach. A bounding sphere set approximation for an object is computed using a variational approach that minimizes the outside volume of the spheres [WZS*06].

Statistical and data clustering techniques. Besides k-means clustering, the Expectation Maximization (EM) algorithm [Har58] is another general data clustering method, which is widely used in image understanding or medical image segmentation. Nevertheless, except for the PCA technique, most statistical and data clustering techniques have not been applied in relation to computing bounding volumes.

3. Variational formulation

In this section we first define the *e-tightness* of a point set in $\mathbb{E}^3$, and then use it to formulate a functional whose minimizer defines an optimal volume segmentation of a given object to facilitate the computation of a tight bounding volume.

Given a volume $\mathcal{S} \subset \mathbb{E}^3$, its covariance matrix is

$$C(\mathcal{S}) = \frac{\int_{x\in\mathcal{S}} (x-\mu)(x-\mu)^T \, d\sigma}{\int_{x\in\mathcal{S}} d\sigma},$$

where $d\sigma$ is the differential volume and $\mu = \int_{x\in\mathcal{S}} x d\sigma / \int_{x\in\mathcal{S}} d\sigma$ is the center of mass of $\mathcal{S}$. We introduce the Legendre ellipsoid of $\mathcal{S}$, denoted by $\mathcal{K}(\mathcal{S})$, as

$\mathcal{K}(\mathcal{S}) = \{x \in \mathbb{E}^3 \mid x^T L^{-1} x \leq 1\}$ [Lei98], where $L = 5C(\mathcal{S})$. The Legendre ellipsoid is a classical concept in mechanics and is so defined that $\mathcal{K}(\mathcal{S})$=$\mathcal{S}$ when $\mathcal{S}$ is an ellipsoid. Note that the Legendre ellipsoid $\mathcal{K}(\mathcal{S})$, in general, is not a bounding ellipsoid of $\mathcal{S}$ (Fig. 2a).

Clearly, the volume of $\mathcal{K}(\mathcal{S})$ is

$$\mathrm{vol}(\mathcal{K}(\mathcal{S})) = \tfrac{4\pi}{3}\sqrt{\det(L)}. \tag{1}$$

Then we define the *e-tightness* function of $\mathcal{S}$ to be

$$e(\mathcal{S}) = \frac{\mathrm{vol}(\mathcal{K}(\mathcal{S}))}{\mathrm{vol}(\mathcal{S})}. \tag{2}$$

In other words, the e-tightness of $\mathcal{S}$ is the ratio of the volume of its Legendre ellipsoid to the volume of $\mathcal{S}$. Here we suppose that $\mathcal{S}$ is a compact set of finite but nonzero volume.

We will see next that the e-tightness of a point set $\mathcal{S}$ characterizes its deviation from the shape of an ellipsoid.

Lemma 1 The e-tightness value of any point set $\mathcal{S} \subset \mathbb{E}^3$ is invariant under affine transformations.

Proof Denote an affine transformation by $\mathcal{T} : X = MX + B$, where M is a nonsingular matrix describing the linear part of $\mathcal{T}$. Let $C(\mathcal{S})$ be the covariance matrix of $\mathcal{S}$. Then it is easy to verify that $C(\mathcal{T}(\mathcal{S})) = MC(\mathcal{S})M^T$. It follows that $\mathrm{vol}(\mathcal{K}(\mathcal{T}(\mathcal{S}))) = \det(M) \cdot \mathrm{vol}(\mathcal{K}(\mathcal{S}))$. But, since $\mathrm{vol}(\mathcal{T}(\mathcal{S})) = \det(M) \cdot \mathrm{vol}(\mathcal{S})$, by Eq. (2), we conclude that $e(\mathcal{T}(\mathcal{S})) = e(\mathcal{S})$. □

Lemma 2 $e(\mathcal{S}) \geq 1$ for any set $\mathcal{S}$ of finite but non-zero volume in $\mathbb{E}^3$. Furthermore, $e(\mathcal{S}) = 1$ if and only if $\mathcal{S}$ is an ellipsoid, assuming $\mathcal{S}$ is a compact set of finite but nonzero volume.

Proof The lemma follows from the classical inequality

$$\mathrm{vol}(\mathcal{K}(\mathcal{S})) \geq \mathrm{vol}(\mathcal{S}),$$

where the equality holds if and only if $\mathcal{S}$ is an ellipsoid [GLYZ02]. □

Example 1. Consider the rectangular box $\mathcal{R} : [-a,a] \times [-b,b] \times [-c,c]$. Its volume is $\mathrm{vol}(\mathcal{R}) = 8abc$. Its covariance matrix is $C(\mathcal{R}) = \frac{8abc}{3}\mathrm{diag}(a^2, b^2, c^2)$. Thus its Legendre ellipsoid $\mathcal{K}(\mathcal{R})$ is

$$(x,y,z)\ \mathrm{diag}\left(\tfrac{3}{5a^2}, \tfrac{3}{5b^2}, \tfrac{3}{5c^2}\right)(x,y,z)^T \leq 1.$$

Then $\mathrm{vol}(\mathcal{K}(\mathcal{R})) = \frac{4 \cdot 5^{3/2}\pi}{3^{5/2}}abc$. So the e-tightness of $\mathcal{R}$ is

$$e(\mathcal{R}) = \frac{\mathrm{vol}(\mathcal{K}(\mathcal{R}))}{\mathrm{vol}(\mathcal{R})} = \frac{5^{3/2}\pi}{2 \cdot 3^{5/2}} \approx 1.127.$$

Due to affine invariance, all rectangular boxes have the same e-tightness.

Next we use the e-tightness to define a functional to characterize an optimal partition of a volume. Given a volume $\mathcal{S} \subset \mathbb{E}^3$, a k-partition of $\mathcal{S}$ is denoted by $\mathcal{P} = \{\mathcal{S}_i\}_{i=1}^k$, where $k \geq 1$, $\bigcup \mathcal{S}_i = \mathcal{S}$ and $\mathcal{S}_i \bigcap \mathcal{S}_j = \emptyset$ for any i, j with $i \neq j$. Consequently, $\sum_{i=1}^k \mathrm{vol}(\mathcal{S}_i) = \mathrm{vol}(\mathcal{S})$. Then we define a functional $F(\mathcal{P})$ as the weighted average of the e-tightness functions of all the components of $\mathcal{P}$, given by

$$F(\mathcal{P}) = \sum_{i=1}^{k} \frac{\mathrm{vol}(\mathcal{S}_i)}{\mathrm{vol}(\mathcal{S})} e(\mathcal{S}_i). \tag{3}$$

The minimizer of $F(\mathcal{P})$ characterizes a partition of $\mathcal{S}$ that is optimal in terms of ellipsoidal decomposition, as summarized by the following lemma.

Lemma 3 For any k-partition $\mathcal{P} = \{\mathcal{S}_i\}_{i=1}^k$ of $\mathcal{S} \subset \mathbb{E}^3$, it holds that $F(\mathcal{P}) \geq 1$. Furthermore, $F(\mathcal{P}) = 1$ holds if and only if every component $\mathcal{S}_i$ of $\mathcal{P}$ is an ellipsoid.

Proof By Lemma 2, $e(\mathcal{S}_i) \geq 1$. Therefore,

$$F(\mathcal{P}) = \sum_{i=1}^{k} \frac{\mathrm{vol}(\mathcal{S}_i)}{\mathrm{vol}(\mathcal{S})} e(\mathcal{S}_i) \geq \sum_{i=1}^{k} \frac{\mathrm{vol}(\mathcal{S}_i)}{\mathrm{vol}(\mathcal{S})} = 1,$$

and $F(\mathcal{P}) = 1$ if and only if $e(\mathcal{S}_i) = 1$ for every i, or, again by Lemma 2, if and only if every $\mathcal{S}_i$ is an ellipsoid. □

An equivalent expression of $F(\mathcal{P})$ is

$$F(\mathcal{P}) = \frac{1}{\mathrm{vol}(\mathcal{S})} \sum_{i=1}^{k} \mathrm{vol}(\mathcal{K}(\mathcal{S}_i)) = \frac{20\sqrt{5}\pi/3}{\mathrm{vol}(\mathcal{S})} \sum_{i=1}^{k} \sqrt{\det(C(\mathcal{S}_i))}. \tag{4}$$

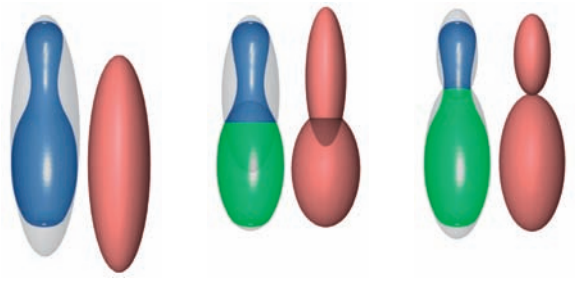
Hence, $F(\mathcal{P})$ can be interpreted as the sum of the volumes of the Legendre ellipsoids of all the components $\mathcal{S}_i$, normalized by the total volume $\mathrm{vol}(\mathcal{S})$. Obviously, $F(\mathcal{P})$ is also invariant under affine transformations.

Lemma 3 states that if an object is composed of k disjoint ellipsoids, a k-partition $\mathcal{P}$ can be found such that $F(\mathcal{P}) = 1$, i.e., each component of $\mathcal{P}$ is an ellipsoid. However, given an arbitrary object $\mathcal{S}$ and a fixed k, such a partition in general is not possible and so we have $F(\mathcal{P}) > 1$ for any partition $\mathcal{P}$ of $\mathcal{S}$ (Fig. 2). It is thus reasonable to say that the partition attaining the minimum value of $F(\mathcal{P})$ is statistically the best decomposition, in the sense that the volume ratio of the Legendre ellipsoids to the corresponding components are minimized. Hence, the minimization of $F(\mathcal{P})$ guides us to obtain a segmentation that facilitates the computation of bounding volumes, as will be explained in Section 5.

4. Minimization of $F(\mathcal{P})$

Given the complex expression of $F(\mathcal{P})$ and its dependence on the shape to be segmented, its minimization cannot be expected to be straightforward. While there could be several potential approaches to designing an effective algorithm, we have focused on an iterative method that is in spirit similar to the Lloyd iteration. There are two ingredients to such a method: *initialization* and *iterative update*. We shall first discuss our strategy for iterative update and then explain how the initialization can be set up.

For the sake of computational efficiency, we assume a



(a) (b) (c)

Figure 2: *Three partitions of a bowling pin. The value of $F(\mathcal{P})$ for (a), (b) and (c) are 1.2097, 1.1116 and 1.0129, respectively. The components and the bounding ellipsoids (grey) are shown on the left and the corresponding Legendre ellipsoids (red) on the right.*

discrete setting where the volume is given by a set of uniformly distributed points $\mathcal{S} = \{x_i\}_{i=1}^n$. Clearly, the concepts introduced in Section 3 can naturally be carried over to this discrete setting. For example, the covariance matrix of $\mathcal{S}$ is given by

$$C(\mathcal{S}) = \frac{1}{n}\sum_{i=1}^{n}(x_i-\mu)(x_i-\mu)^T,$$

where $\mu = \frac{1}{n}\sum_{i=1}^n x_i$ is the center of mass of $\mathcal{S}$.

We first use the case of a 2-partition to explain the basic idea. Let the current partition be $\mathcal{P} = \{\mathcal{S}_1, \mathcal{S}_2\}$. Then a key issue is, according to the basic variational principle, how to determine the perturbation imposed upon $F(\mathcal{P})$ if one point in $\mathcal{S}_1$ is reassigned to $\mathcal{S}_2$, or vice versa. Specifically, take an arbitrary point $x \in \mathcal{S}_1$, and consider the new partition by $\mathcal{P}' = \{\mathcal{S}_1', \mathcal{S}_2'\}$, where $\mathcal{S}_1' = \mathcal{S}_1 \setminus \{x\}$ and $\mathcal{S}_2' = \mathcal{S}_2 \bigcup \{x\}$. Clearly, $\mathcal{P}$ is a minimizer if $F(\mathcal{P}) \le F(\mathcal{P}')$ for any x. To convert this into a computational scheme, we must be able to compute the variation $\Delta F = F(\mathcal{P}') - F(\mathcal{P})$. If $\Delta F < 0$, we accept $\mathcal{P}'$ as a better partition towards the minimization of $F(\mathcal{P})$.

In the following we will show that the evaluation of ΔF leads naturally to an approximate algorithm that is similar to the k-means clustering but with its Euclidean metric replaced by the Mahalanobis metric weighted by e-tightness.

Proposition 4 The variation in $F(\mathcal{P})$, for reassigning a point x from $\mathcal{S}_1$ to $\mathcal{S}_2$ ($\mathcal{S}_1, \mathcal{S}_2 \in \mathcal{P}$), is given by

$$\Delta F = \alpha\left(\frac{\left[\left(\frac{m}{m-1}\right)^3 - \left(\frac{m}{m-1}\right)^3 \frac{d(x,\mathcal{S}_1)}{m-1} - 1\right]\det(C(\mathcal{S}_1))}{\left(\frac{m}{m-1}\right)^{\frac{3}{2}}\sqrt{\left(1-\frac{d(x,\mathcal{S}_1)}{m-1}\right)\det(C(\mathcal{S}_1)) + \sqrt{\det(C(\mathcal{S}_1))}}}\right.$$
$$\left. + \frac{\left[\left(\frac{n}{n+1}\right)^3 + \left(\frac{n}{n+1}\right)^3 \frac{d(x,\mathcal{S}_2)}{n+1} - 1\right]\det(C(\mathcal{S}_2))}{\left(\frac{n}{n+1}\right)^{\frac{3}{2}}\sqrt{\left(1+\frac{d(x,\mathcal{S}_2)}{n+1}\right)\det(C(\mathcal{S}_2)) + \sqrt{\det(C(\mathcal{S}_2))}}}\right) \quad (5)$$

where $m = |\mathcal{S}_1|$, $n = |\mathcal{S}_2|$, $\alpha = \frac{20\sqrt{5}\pi/3}{\mathrm{vol}(\mathcal{S})}$, and

$$d(x,\mathcal{S}_i) = (x-\mu_i)^T C(\mathcal{S}_i)^{-1}(x-\mu_i),\ i = 1,2,$$

are the Mahalanobis distances from x to the centres μ_1 and μ_2 of $\mathcal{S}_1$ and $\mathcal{S}_2$, respectively.

Proof The derivation is straightforward but lengthy, so is omitted due to space limitation. □

When $m = |\mathcal{S}_1|$ and $n = |\mathcal{S}_2|$ are sufficiently large, it is straightforward to obtain the following approximation:

$$\Delta F \approx \Delta\tilde{F} \equiv \alpha\left(\frac{d(x,\mathcal{S}_2)\sqrt{\det\left(C(\mathcal{S}_2)\right)}}{2(n+1)} - \frac{d(x,\mathcal{S}_1)\sqrt{\det\left(C(\mathcal{S}_1)\right)}}{2(m-1)}\right).$$

We now consider the case of a k-partition of a volume $\mathcal{S}$. Given a point $x \in \mathcal{S}_i$, let $\Delta\tilde{F}_{i,j}$ denote the variation in F due to the reassignment of x to $\mathcal{S}_j$. Then, for sufficiently large $|\mathcal{S}_i|$ and $|\mathcal{S}_j|$, we have

$$\begin{aligned}\Delta\tilde{F}_{i,j} &= \alpha\left(\frac{d(x,\mathcal{S}_j)\sqrt{\det\left(C(\mathcal{S}_j)\right)}}{2(|\mathcal{S}_j|+1)} - \frac{d(x,\mathcal{S}_i)\sqrt{\det\left(C(\mathcal{S}_i)\right)}}{2(|\mathcal{S}_i|-1)}\right)\\ &\approx \gamma\big(e(\mathcal{S}_j)\,d(x,\mathcal{S}_j) - e(\mathcal{S}_i)\,d(x,\mathcal{S}_i)\big), \quad (6)\end{aligned}$$

where $d(x,\mathcal{S}_i)$ is the Mahalanobis distance from x to the center of $\mathcal{S}_i$ and γ is a positive constant. Here we have used the definition of e-tightness by Eqn. (1) and (2), and the approximation $|\mathcal{S}_i| \approx \mathrm{vol}(\mathcal{S}_i)$, since the sampled points in $\mathcal{S}$ are sufficiently dense and uniformly distributed. Eqn. (6) implies that, approximately, a weighted Mahalanobis metric (i.e., by the e-tightness) can be used in the k-means framework to optimize the functional $F(\mathcal{P})$.

Define $h_j(x) = e(\mathcal{S}_j)\,d(x,\mathcal{S}_j)$ for a point $x \in \mathcal{S}$. A smaller value of $h_j(x)$ gives a smaller $\Delta\tilde{F}_{i,j}$, which favors reassigning x to $\mathcal{S}_j$. Note that $x \in \mathcal{S}_i$ is reassigned to $\mathcal{S}_\ell$ only if $\Delta\tilde{F}_{i,\ell} < 0$ and $\Delta\tilde{F}_{i,\ell} < \Delta\tilde{F}_{i,j}$ for all $j \neq i,\ell$. By Eqn. (6), this is equivalent to assigning $x \in \mathcal{S}_i$ to $\mathcal{S}_\ell$ if $h_\ell(x) \le h_i(x)$ for all $i \neq \ell$. For better efficiency and maintaining component connectivity, we use a flooding scheme as in [CSAD04] (assuming 8-neighbor connectivity) to compute a k-partition by minimizing the functional $F(\mathcal{P})$, as shown in the following algorithm flow.

ALGORITHM: Minimizing $F(\mathcal{P})$ for a k-partition of a volume $\mathcal{S}$

INPUT: An initial k-partition $\mathcal{P} = \{\mathcal{S}_i\}_{i=1}^k$ of $\mathcal{S}$

STEPS:

1. $\mathcal{P}_{curr} \leftarrow \mathcal{P}$
2. For each point $x \in \mathcal{S}$, assign x to $\mathcal{S}_j$ with $h_j(x)$ being the minimum among all $h_i(x)$, $i = 1,\ldots,k$. Distortion minimizing flooding is used in the assignment process.
3. If no point has been reassigned, goto step 6.
4. Let $\mathcal{P}_{new}$ be the new partition formed in step 2. Evaluate $F(\mathcal{P}_{new})$ using Eqn. (4).

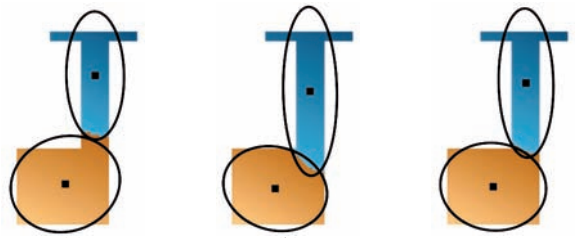


Figure 3: *A 2-partition of a 2D shape using different segmentation schemes: (a) k-means clustering with Euclidean metric; (b) k-means clustering with Mahalanobis metric; and (c) our scheme of minimizing $F(\mathcal{P})$.*

5. If $F(\mathcal{P}_{new}) < F(\mathcal{P}_{curr})$,
   $\mathcal{P}_{curr} \leftarrow \mathcal{P}_{new}$, goto step 2.
6. Randomly pick a boundary point y. Let $y \in \mathcal{S}_i$ and let $\{\mathcal{S}_j\}$ be the set of components that are adjacent to y.
   Evaluate $\Delta F_{i,j}$ as in Eqn. (5).
   If $\Delta F_{i,j} < 0$ for any $\mathcal{S}_j$,
   reassign y to $\mathcal{S}_j$ and goto step 2.
   If all boundary points of all components cannot be reassigned,
   output $F(\mathcal{P}_{curr})$ as the optimized k-partition of $\mathcal{S}$.

Step 6 above serves to check whether the re-grouping performed in step 2 is acceptable. This is needed because (*a*) the criterion $\Delta\tilde{F} < 0$ is only an approximate one; and (*b*) for efficiency reasons, the covariance matrices $C(\mathcal{S}_i)$ defining $h_i(x)$ are not updated after moving each individual point—they are updated only at the beginning of each iteration, i.e., after $\mathcal{P}_{new}$ of the last round is formed. All these affect the accuracy and therefore, correctness, of this regrouping step. Hence, in step 6, we seek to perturb the boundary points of the components in a partition, and see if the reassignment of any of these boundary points can result in a $\Delta F < 0$, which in turn will lead to a new partition with smaller $F(\mathcal{P})$.

A Centroidal Voronoi Tessellation (CVT) of k components of $\mathcal{S}$ is used as the initial k-partition for the input of the above algorithm. This partition is computed using the k-means clustering based on Euclidean distance with the flooding scheme to ensure component connectivity. The initial k-partition can also be obtained by splitting one component of an optimal $(k-1)$-partition.

We close this section by giving a geometric interpretation of the Mahalanobis metric in the k-means clustering. Figure 3 shows the 2-partitions of a 2D shape obtained by different schemes, all starting from two reasonably good initial seed points. Since our scheme is closely related to the k-means clustering using the Mahalanobis metric, we expect our method to have similar behavior (Figure 3(c)). Our tests show that the minimization of $F(\mathcal{P})$ is more robust in global convergence as it is properly derived from sound variational principles. In contrast, the k-means clustering using the Mahalanobis metric has the peculiar property [WMSX97] that the functional, $G(\mathcal{P}) = \sum_{i=1}^{k} \sum_{x \in \mathcal{S}_i} (x-\mu_i)^T C(\mathcal{S}_i)^{-1} (x-\mu_i)$, obtained by replacing the Euclidean metric with the Mahalanobis metric in the well known functional for the CVT [DFG99], is actually a constant function; that is, it is independent of the number of components in a partition or the manner of partition. This lack of clear variational interpretation is the major obstacle to using the Mahalanobis metric.

5. Complete algorithm

The goal of our complete algorithm for bounding volume computation is to segment a 3D shape into multiple components, so that each component is enclosed tightly by a bounding volume. Intuitively, one could think of minimizing the bounding volumes of the components as an objective function to seek an optimal partition; but that would lead to a functional without explicit expression, making its optimization intractable. Hence, the algorithm in Section 4 comes into place by providing a k-partition that minimizes $F(\mathcal{P})$, for a fixed k, as defined in Eqn. (3). The global optimization is therefore driven so that each component tends to resemble the shape of an ellipsoid. However, minimizing $F(\mathcal{P})$ alone is not sufficient for producing a good segmentation for bounding volume computation, due to the following issues.

1. *The minimizer of $F(\mathcal{P})$ may be over-optimistic*. Consider the case where a component $\mathcal{S}_i$ is an ellipsoid with a long but thin stick attached to it. Since the thin stick is statistically negligible when computing the covariance matrix of $\mathcal{S}_i$, by Lemma 2, $e(\mathcal{S}_i) \approx 1$, which suggests a good fitting. However, since the bounding volume is much larger than its Legendre ellipsoid and contains much empty space, $\mathcal{S}_i$ should be further segmented for a tighter bounding.
2. *The minimizer of $F(\mathcal{P})$ may be over-conservative*. Consider a rectangular block $\mathcal{R}$ (with $e(\mathcal{R}) \approx 1.127$; see Example 1 in Section 3). If it is segmented into two smaller rectangular boxes (which have the same e-tightness 1.127, by Lemma 1), then $F(\mathcal{P})$ is still 1.127, indicating no improvement. However, due to overlapping between the Legendre ellipsoids of the two smaller boxes, the actual bounding tightness has become better if bounding volumes are computed based on this 2-partition.

Hence, our complete algorithm consists of an iteration of two stages. Firstly, we obtain an optimal k-partition $\mathcal{P}$ that minimizes $F(\mathcal{P})$. Next, we validate $\mathcal{P}$ by considering some errors measuring the empty space inside a bounding volume, and refine the segmentation by increasing k, if necessary.

Let $\mathcal{BV}(\mathcal{S}_i)$ denote the bounding volume of $\mathcal{S}_i$ computed by our algorithm. The global error, $\mathrm{TOV}(\mathcal{P})$, measures the normalized *total outside volume*, i.e., the space outside $\mathcal{S}$ but

inside the union of all bounding primitives, and is defined as

$$\mathrm{TOV}(\mathcal{P}) = \frac{\mathrm{vol}\big(\bigcup_i \mathcal{BV}(\mathcal{S}_i) \setminus \mathcal{S}\big)}{\mathrm{vol}(\mathcal{S})}.$$

The *local outside volume* of a component $\mathcal{S}_i$, denoted by $\mathrm{OV}(\mathcal{P}, \mathcal{S}_i)$, measures the normalized volume of the space outside $\mathcal{S}$ but inside the bounding volume of $\mathcal{S}_i$, and is defined as

$$\mathrm{OV}(\mathcal{P}, \mathcal{S}_i) = \frac{\mathrm{vol}\big(\mathcal{BV}(\mathcal{S}_i) \setminus \mathcal{S}\big)}{\mathrm{vol}(\mathcal{S}_i)}.$$

We then define the maximum local outside volume $\mathrm{MOV}(\mathcal{P}) = \max_i\{\mathrm{OV}(\mathcal{P}, \mathcal{S}_i)\}$, i.e., the maximum local error over all the components of a partition. This volume is evaluated by counting the number of sample points in the regions in question.

We iteratively refine a partition of $\mathcal{S}$ by increasing the number of components by one at each step, until $\mathrm{TOV}(\mathcal{P})$ and $\mathrm{MOV}(\mathcal{P})$ are smaller than two pre-defined tolerances σ_G and σ_L, respectively (Figure 4). The flow of the complete algorithm is as follows.

ALGORITHM: Computing the bounding volumes of a 3D shape

INPUT: A 3D shape $\mathcal{S}$ and two tolerances σ_G and σ_L for the global and the local error measures, respectively.

STEPS:

1. Sample $\mathcal{S}$ by a set of uniformly distributed points.
2. Pick a random seed point and form a 1-partition of $\mathcal{S}$. $k \leftarrow 1$.
3. Obtain an optimized k-partition, $\mathcal{P}^k$, of $\mathcal{S}$ using the algorithm in Section 4 by minimizing $F(\mathcal{P}^k)$.
4. Compute the bounding volumes $\mathcal{BV}(\mathcal{S}_i)$. Evaluate $\mathrm{TOV}(\mathcal{P}^k)$ and $\mathrm{MOV}(\mathcal{P}^k)$.
5. If $\mathrm{TOV}(\mathcal{P}^k) < \sigma_G$ and $\mathrm{MOV}(\mathcal{P}^k) < \sigma_L$,
   goto step 7.
6. If $\mathrm{TOV}(\mathcal{P}^k) < \mathrm{TOV}(\mathcal{P}^{k-1})$,
   split $\mathcal{S}_j$ having the maximum $\mathrm{OV}(\mathcal{P}^k, \mathcal{S}_j)$
   $k \leftarrow k+1$ and go to step 3.
   Else
   backtrack to $\mathcal{P}^{k-1}$ and find the component $\mathcal{S}_b$ with the next largest local outside volume.
   If no $\mathcal{S}_b$ can be found, i.e., all components in $\mathcal{P}^{k-1}$ have been subject to split but failed,
   split $\mathcal{S}_j$ having the maximum $\mathrm{OV}(\mathcal{P}^k, \mathcal{S}_j)$
   $k \leftarrow k+1$ and go to step 3.
   Else
   split $\mathcal{S}_b$ and go to step 3.
7. Perform components *merge*.
8. Output the set of bounding volumes $\mathcal{BV}(\mathcal{S}_i)$.

5.1. Splitting components for partition refinement

Step 6 of the above algorithm is a refinement step: we identify the component $\mathcal{S}_j \in \mathcal{P}^k$ with the maximum local error and split it into two, and target at reducing both the total outside volume and the maximum local outside volume. A split is done by adding a new seed point which is farthest from the centre of $\mathcal{S}_j$; then the points in $\mathcal{S}_j$ are regrouped into two components using our optimization scheme in Section 4. Such a split will lead to a minimization of a new partition $\mathcal{P}^{k+1}$. However, in rare cases the optimized partition after a split may have a larger total outside volume than the previous one. In this case, we backtrack to the old partition $\mathcal{P}^k$ and select the component with the second largest local outside volume for the next splitting. If all components have been attempted but yet the total outside volume cannot be reduced, then we will proceed to split the component with the largest local outside volume which would lead to an increase in the total outside volume. This situation happens rarely because in most cases a split will result in two tighter bounding volumes than the old one. Our tests show that even a backtrack occurs very infrequently.

5.2. Merging components

In a post-processing step of our algorithm, to prevent oversegmentation, we seek to reduce the final number of components without incurring an increase in the total outside volume. We determine the pair of adjacent components whose merging will lead to the largest decrease in the total outside volume. Merging is then performed for such a pair, and continues iteratively until no possible merge can be identified.

5.3. Computing bounding volumes

Once a 3D volume has been segmented into multiple components, the next step is to compute the bounding volume for each component. An approximate minimum-volume enclosing ellipsoid of a component can be computed using CGAL [FGH*06] and used as a bounding ellipsoid; however, the method is computationally expensive (almost 6 seconds for 10k points). Hence, for efficiency reasons, during the intermediate steps of computing the bounding volumes for evaluating the error measures, we simply apply uniform scalings to the Legendre ellipsoids to bound their corresponding components. Only the final bounding ellipsoids for output will be computed by CGAL. We also consider the use of OBB [GLM96] as a bounding volume, since the Legendre ellipsoid provides the same information as by a PCA. Figure 5 shows that either OBBs or ellipsoids can be a better choice of bounding volumes, depending on the type of the model under consideration. While synthetic objects are better bounded by OBBs, our experiments show that objects of organic forms, such as human characters, are bounded by ellipsoids more efficiently.

Our segmentation scheme can be used to set up a bound-

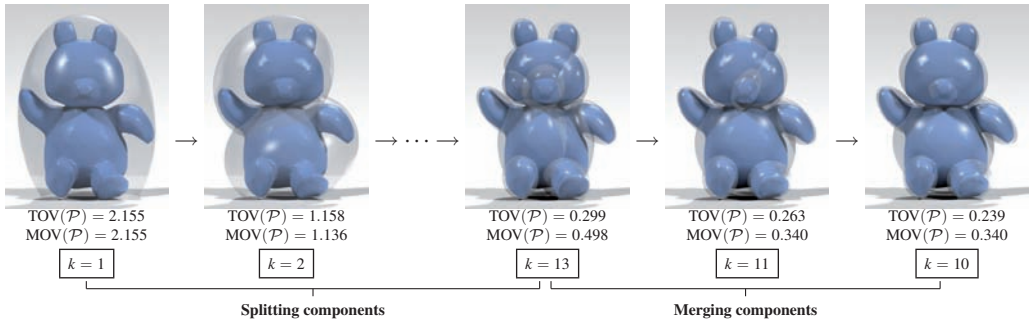


Figure 4: *The process of segmenting and computing the bounding volume of a teddy bear model, starting from one component that ends at a 10-partition. The tolerances used are* $\sigma_G = 0.3$, $\sigma_L = 0.5$.

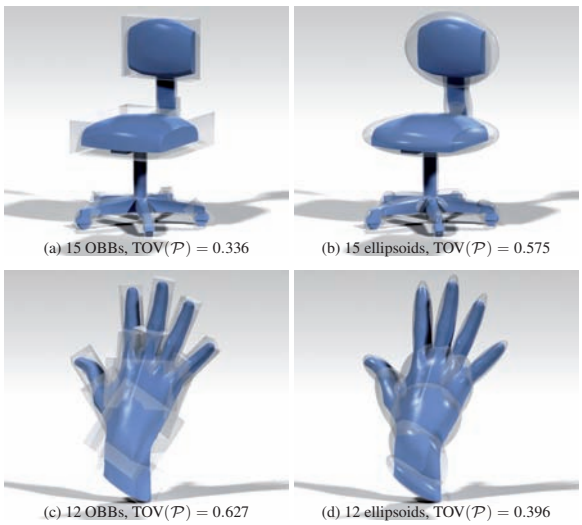


Figure 5: *OBBs and ellipsoids achieve different degrees of tightness for different types of objects. The chair ((a) & (b)) is bounded more tightly by OBBs while the hand ((c) & (d)) is bounded more tightly by ellipsoids.*

ing volume hierarchy. We first obtain a *k*-partition (and hence *k* bounding primitives) to attain a sufficiently tight bounding. We then build the hierarchy bottom up; the components of two adjacent bounding volumes are grouped and are enclosed tightly by a parent bounding volume. A greedy approach is used where the grouping is first performed on the pair of ellipsoids that results in the smallest parent bounding volume. If needed, the *k*-partition, which comprises the leave nodes of the tree, can be further split in a top-down manner to obtain lower levels of bounding. Here, splitting of parent bounding volume is determined by a 2-partition optimization of its component.

5.4. Other error measures

The above volume-based error measures used to govern segmentation refinement can also be replaced by other kinds of error measurement, depending on specific applications. While the volume-based errors are relevant to collision detection, restricting the Hausdorff distance between the bounding volume $\mathcal{B}$ and the object boundary $\mathcal{O}$ can be useful for shadow computation, for example, since it provides a more perception-sensitive measure. Different error measures can be used in combination. In this case, the segmentation refinement terminates when the thresholds for all the metrics are satisfied.

6. Implementation issues

6.1. Segmentation for skeleton-based volumes

Extracting the skeleton of an object, especially an articulated object as commonly used in animation, has been well studied. Any approximate skeletal structure is good enough as an initial input to our algorithm. We assign a seed point to each link that naturally defines a component in the partition. A CVT is used as an initial partition for the optimization as described in Section 4 for obtaining a *k*-partition with fixed *k*. This saves much computation time that would otherwise be spent on building the *k*-partition from a single initial component (Figure 6).

6.2. Efficiency vs. quality

The computational efficiency depends highly on the number of samples taken from the 3D volume. While an overly high sampling rate leads to slow computation, a sparse sampling could easily induce errors to the bounding volumes, since fine parts of an input object, often important as features, may receive too few sample points. We use two techniques, multi-grid and adaptive sampling, in a preprocessing step to balance segmentation quality and computational time.

Multi-grid. We apply a discrete multi-grid technique where a progressively finer sampling is used as the Lloyd iterations

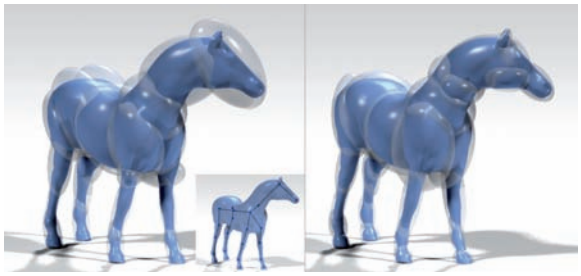


Figure 6: *Segmentation of a 3D shape with skeleton. (Left) Initial fitting with 22 bounding ellipsoids (*$\mathrm{TOV}(\mathcal{P}) = 0.434$, $\mathrm{MOV}(\mathcal{P}) = 1.175$*); (right) Segmentation output with 30 bounding ellipsoids using* $\sigma_G = 0.3$ *and* $\sigma_L = 0.4$ *(*$\mathrm{TOV}(\mathcal{P}) = 0.298$, $\mathrm{MOV}(\mathcal{P}) = 0.336$*).*

Figure 7: *More points are sampled in regions of small features, e.g., the fingers, so that they can be bounded tightly.*

proceed. A pre-computed multiresolution point sampling of the volume is maintained. The results of an optimized partitioning of a coarse level is transferred to the next finer level by copying the assignments of sample points to their corresponding points at the finer grid, thus providing a good initialization to facilitate faster convergence.

Adaptive sampling. Each sample point is associated with its distance to the shape boundary by applying a medial-axis transform [ACK01] at a preprocessing step. Whenever an optimized partition is computed, we evaluate the average distance to the shape boundary over all points in each component. An average distance smaller than a predefined threshold indicates a possible feature and therefore a denser sampling is used. Each sample point then carries a weight proportional to the volume it represents, so that a point in a more densely sampled region contributes less in the Legendre ellipsoids computations and error evaluations. Figure 7 shows a segmentation of a human character with each finger properly and tightly bounded by an ellipsoid using adaptive sampling.

7. Experimental results and discussions

Figure 8 shows the 20-partitions of a human model (15K sample points) obtained by a CVT (1.8 seconds) and our

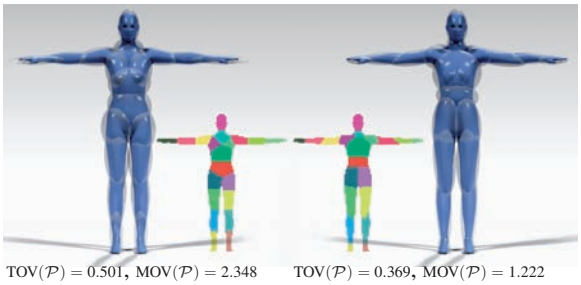


Figure 8: *Segmentation of a human character obtained by (left) the CVT method; and (right) our algorithm. Both contain 20 components, and our method can automatically decompose the limbs properly.*

method (4 seconds with optimization on the partition only, without any splitting or merging of components). The CVT method generates components corresponding to the Voronoi cells and hence cannot capture some elongated shapes especially at the limbs. Our algorithm, on the other hand, shows superiority over the commonly used CVT method, due to the anisotropic nature in the metrics that we use. There is a significant difference between the anisotropy of our method and that of some other anisotropic data clustering methods which use a fixed Riemannian metric; the anisotropy used in this paper varies as the partition is improved progressively.

Figure 9 shows the results of applying our algorithm to some complex models. The running time of our algorithm depends on several factors, such as the number of sample points, the type of bounding volumes, and the user-specified tolerances that control the degree of segmentation refinement. With 20K sample points, our algorithm takes 2 minutes to generate the 33 bounding ellipsoids for the dinopet as shown in Figure 1. The use of skeletal information results in significant speedup—the bounding ellipsoids for the horse (20K points) (Figure 6) are computed in 30 seconds only. All timing results are taken on a 1.66GHz Pentium IV computer with 1GB RAM.

8. Conclusion

We presented a novel algorithm in computing a segmentation for bounding volume computation based on a new functional, for which we provided theoretical justifications. Our experimental results show that the proposed algorithm can produce a shape decomposition that is more shape adaptive, as compared to segmentation using k-means clustering with the Euclidean metric. Combined with other error measure, such as the total outside volume, our algorithm is capable of producing tight bounding volumes.

We point out that the formulation of the functional $F(\mathcal{P})$ in Eq.(3) is not the only way of utilizing the concept of e-tightness. Other formulations and their behaviors remain to

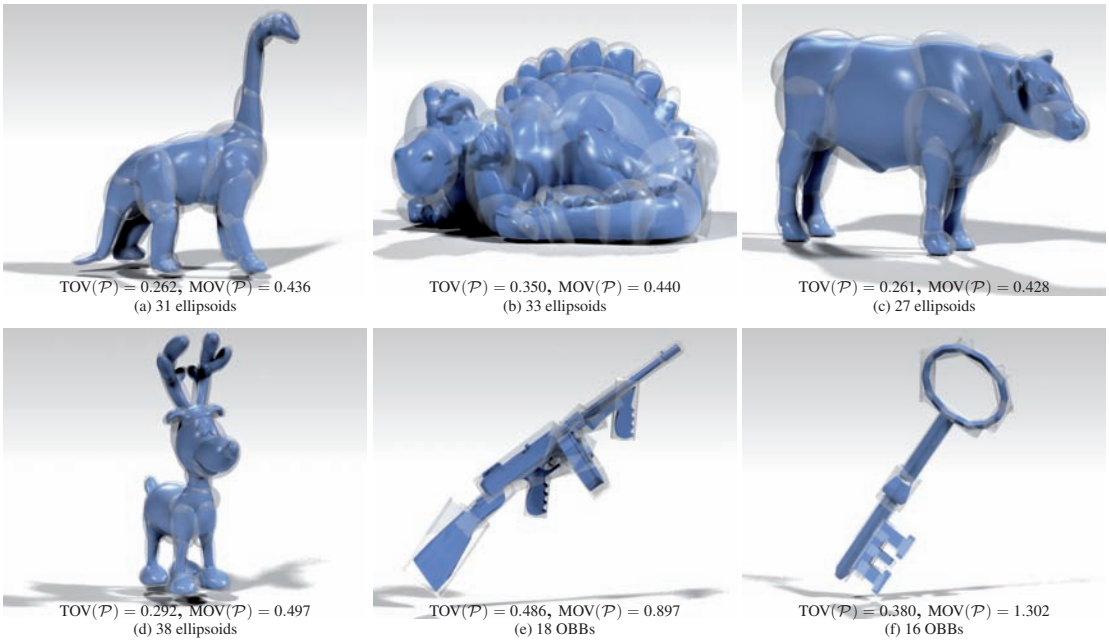


Figure 9: *Results of our algorithm on different models.*

be studied. We also wish to study the effect of using additional local geometric features, such as surface curvature, for a better control in the resulting partition. Finally, the feasibility of adopting our algorithm for computing the bounding volume of dynamic or deformable objects will be explored in future work.

Acknowledgements

We would like to thank anonymous reviewers for their invaluable comments. The work of Wenping Wang was partially supported by the National Key Basic Research Project of China (2004CB318000), the Research Grant Council of Hong Kong (HKU 7178/06E), and the Innovative and Technology Fund of Hong Kong (ITS/090/06). This research was also supported in part by the Korean Ministry of Information and Communication (MIC) through the IT Research Center for CCGVR.

References

[ACK01] AMENTA N., CHOI S., KOLLURI R. K.: The power crust, unions of balls, and the medial axis transform. *Comput. Geom. 19*, 2-3 (2001), 127–153.

[BCG*96] BAREQUET G., CHAZELLE B., GUIBAS L. J., MITCHELL J. S. B., TAL A.: BOXTREE: A hierarchical representation for surfaces in 3D. *Comput. Graph. Forum 15*, 3 (1996), 387–396.

[BK02] BISCHOFF S., KOBBELT L.: Ellipsoid decomposition of 3D-model. In *IEEE 3DPVT* (2002), pp. 480–489.

[BO04] BRADSHAW G., O'SULLIVAN C.: Adaptive medial-axis approximation for sphere-tree construction. *ACM Trans. Graph. 23*, 1 (2004), 1–26.

[Bou85] BOUVILLE C.: Bounding ellipsoids for ray-fractal intersection. In *SIGGRAPH '85*, pp. 45–52.

[CSAD04] COHEN-STEINER D., ALLIEZ P., DESBRUN M.: Variational shape approximation. In *SIGGRAPH '04*, pp. 905–914.

[DFG99] DU Q., FABER V., GUNZBURGER M.: Centroidal Voronoi tessellations: Applications and algorithms. *SIAM Rev. 41*, 4 (1999), 637–676.

[FGH*06] FISCHER K., GARTNER B., HERRMANN T., HOFFMANN M., PACKER E., SCHONHERR S.: Geometric optimisation. In *CGAL-3.2 User and Reference Manual*, Board C. E., (Ed.). 2006.

[GG89] GATH I., GEVA A. B.: Unsupervised optimal fuzzy clustering. *IEEE Trans. Pattern Anal. Mach. Intell. 11*, 7 (1989), 773–780.

[GLM96] GOTTSCHALK S., LIN M. C., MANOCHA D.: OBBTree: A hierarchical structure for rapid interference detection. In *SIGGRAPH '96*, pp. 171–180.

[GLYZ02] GULERYUZ O. G., LUTWAK E., YANG D., ZHANG G.: Information-theoretic inequalities for contoured probability distributions. *IEEE Trans. on Information Theory 48*, 8 (Aug. 2002), 2377–2383.

[Har58] HARTLEY H.: Maximum likelihood estimation from incomplete data. *Biometrics 14* (1958), 174–194.

[HKM95] HELD M., KLOSOWSKI J. T., MITCHELL J. S. B.: Evaluation of collision detection methods for virtual reality fly-throughs. In *Seventh Canadian Conference on Computational Geometry* (1995), pp. 205–210.

[Hub96] HUBBARD P. M.: Approximating polyhedra with spheres for time-critical collision detection. *ACM Trans. Graph. 15*, 3 (1996), 179–210.

[JTT01] JIMÉNEZ P., THOMAS F., TORRAS C.: 3D collision detection: a survey. *Computers & Graphics 25*, 2 (2001), 269–285.

[KHM*98] KLOSOWSKI J. T., HELD M., MITCHELL J. S. B., SOWIZRAL H., ZIKAN K.: Efficient collision detection using bounding volume hierarchies of k-DOPs. *IEEE Trans. Vis. Comput. Graph. 4*, 1 (1998), 21–36.

[KMN*02] KANUNGO T., MOUNT D. M., NETANYAHU N. S., PIATKO C. D., SILVERMAN R., WU A. Y.: An efficient k-means clustering algorithm: Analysis and implementation. *IEEE Trans. Pattern Anal. Mach. Intell. 24*, 7 (2002), 881–892.

[KT03] KATZ S., TAL A.: Hierarchical mesh decomposition using fuzzy clustering and cuts. *ACM Trans. Graph. 22*, 3 (2003), 954–961.

[KV05] KALAIAH A., VARSHNEY A.: Statistical geometry representation for efficient transmission and rendering. *ACM Trans. Graph. 24*, 2 (2005), 348–373.

[Lei98] LEICHTWEISS K.: *Affine Geometry of Convex Bodies*. Johann Ambrosius Barth, Heidelberg, 1998.

[Llo82] LLOYD S. P.: Least squares quantization in PCM. *IEEE Transactions on Information Theory 28*, 2 (1982), 129–136.

[RB97] RIMON E., BOYD S. P.: Obstacle collision detection using best ellipsoid fit. *J. Intell. Robotics Syst. 18*, 2 (1997), 105–126.

[SS05] SIMARI P. D., SINGH K.: Extraction and remeshing of ellipsoidal representations from mesh data. In *Proceedings of Graphics Interface '05*, pp. 161–168.

[WCC*04] WANG W., CHOI Y.-K., CHAN B., KIM M.-S., WANG J.: Efficient collision detection for moving ellipsoids using separating planes. *Computing 72*, 1-2 (2004), 235–246.

[WMSX97] WANG S., MA F., SHI W., XIA S.: The hyperellipsoidal clustering using genetic algorithm. In *IEEE ICIPS '97* (1997), pp. 592–596.

[WZS*06] WANG R., ZHOU K., SNYDER J., LIU X., BAO H., PENG Q., GUO B.: Variational sphere set approximation for solid objects. *The Visual Computer 22*, 9-11 (2006), 612–621.