

A Heuristic Convexity Measure for 3D Meshes

Insert your subtitle here, otherwise leave blank

First Author · Second Author

Abstract In this paper we propose a heuristic convexity measure for 3D meshes. Built upon a state-of-the-art convexity measure that employs a time-consuming genetic algorithm for optimization, our new measure projects only once a given 3D mesh onto the orthogonal 2D planes along its principal directions for an initial estimation of mesh convexity, followed by a correction calculation based on mesh slicing. Our measure experimentally shows several advantages over the state-of-the-art one: First, it accelerates the overall computation by approximately an order of magnitude; second, it properly handles those bony meshes usually overestimated by the state-of-the-art measure; third, it improves the accuracy of the state-of-the-art measure in 3D mesh retrieval.

1 Introduction

Shape analysis has been a vigorous research field for decades, and one of its research focuses is to study how to define a scalar to holistically describe geometric properties of shapes, such as convexity [22], rectilinearity [7, 12, 21], regularity [2], ellipticity, rectangularity, triangularity [11], and concavity [20]. Among these shape properties, convexity is a crucial measure in both 2D and 3D shape analysis and plays a fundamental role in shape decomposition [1, 4, 8, 18], classification [9, 13, 17] and retrieval [6, 15]. In geometry, a planar shape is referred to as convex if an arbitrary line has no more than two intersections with the boundary of the shape

in the plane. Generally speaking, four desirable conditions have to be satisfied in justifying a convexity measure.

1. The value of the convexity measure is a real number and varies between $(0, 1]$ for all shapes;
2. the value is equal to 1 if and only if the shape is convex;
3. there exist shapes whose convexity measure is arbitrarily close to 0;
4. the convexity measure of a given shape is invariant under similarity transformations.

These four conditions can be generalized to 3D by replacing the term shapes with 3D meshes.

1.1 Convexity measures for 2D shapes

Convexity measurement for 2D shapes has been extensively studied up to now. The most commonly used convexity measure for 2D shapes is based on the area ratio of a shape over its convex hull [22], as defined below. Note that for the sake of clarity, in this paper we define all the 2D convexity measures with c in lower case, while all the 3D convexity measures with C in capital.

Definition 1 For a given 2D shape s and its convex hull $CH(s)$, its convexity measure $c_1(s)$ is formulated as

$$c_1(s) = \frac{Area(s)}{Area(CH(s))}. \quad (1)$$

c_1 is easy to evaluate and generally robust to boundary noise, but it fails to sense extremely slim dents in the shape. This problem can be overcome by introducing a perimeter-based convexity measure, which takes

F. Author
first address

S. Author
second address

boundary into account. For any planar shape, there exists an inequality between the perimeters of itself and its convex hull, *i.e.* $Per(CH(s)) \leq Per(s)$ and they are equal only if s is convex. Therefore, the perimeter-based convexity measure is defined as the perimeter ratio of a shape over its convex hull.

Definition 2 For a given 2D shape s and its convex hull $CH(s)$, its convexity measure $c_2(s)$ is defined as

$$c_2(s) = \frac{Per(CH(s))}{Per(s)}. \quad (2)$$

c_2 is based on shape boundary and more sensitive to boundary noise than c_1 [22].

There are other measures that calculate convexity regardless of the convex hull. For example, Zunic *et al* [22] proposed a boundary-based convexity measure based on boundary projection of the 2D shape. Rosin *et al* [14] proposed a symmetric convexity measure with a convex polygon that best fits the 2D shape, and their measure is defined as the area ratio of the convex polygon and the shape. Rahtu *et al* proposed a convexity measure defined as the probability that a point on any line segment formed by an arbitrary pair of points from a shape also belongs to the shape [10].

1.2 Convexity measures for 3D meshes

Unlike 2D convexity measurement that has been extensively studied, there has been far less work reported for 3D convexity measurement thus far. A few intuitive attempts for 3D meshes were built on the 2D measures mentioned above. It is worth noting that the term 3D mesh in the context of this paper is referred to as 3D closed mesh and visualized as 3D shaded objects. Similar to **Definition 1** of 2D shapes, a volume-based measure, that is the 3D generalization of c_1 , can be formulated for 3D meshes.

Definition 3 Letting M stand for an arbitrary 3D mesh and $CH(M)$ indicate its convex hull, its convexity measure $C_1(M)$ is evaluated as

$$C_1(M) = \frac{Volume(M)}{Volume(CH(M))} \quad (3)$$

Similar to its 2D counterpart, C_1 is insensitive to extremely slim dents [6] and cannot sense the difference of two meshes with identical mesh and convex hull volumes, as shown in Figure 1.

Similarly, we directly generalize the perimeter-based convexity measure c_2 into 3D. The 3D counterpart of 2D perimeter is regarded as mesh surface area. However, it is hard to construct an inequality for a 3D mesh

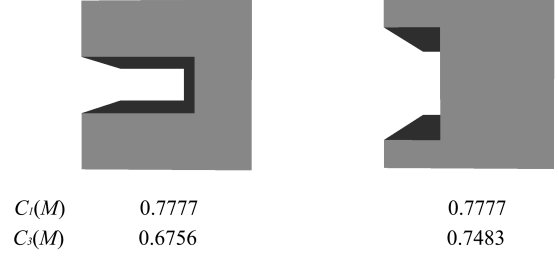


Fig. 1 Two different meshes but with the same C_1 value.

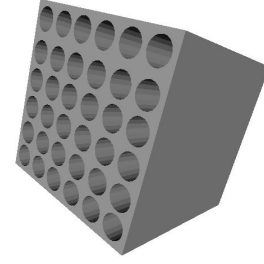


Fig. 2 A cube with many holes.

with its mesh surface area and its convex hull area. This is because for some 3D meshes, such as the one shown in Figure 2

$$Area(M) > Area(CH(M)),$$

while for others, such as the hollow cube shown in Figure 8, especially when the bars go extremely slim, the inequality becomes

$$Area(M) < Area(CH(M)).$$

Therefore, the 3D generalization of c_2 may not always hold for 3D meshes.

To resolve the problem that C_1 is insensitive to extremely slim dents, Lian *et al* [6] proposed a projection-based convexity measure for 3D meshes. Their measure was generalized from the 2D projection-based convexity measure reported by Zunic *et al* [22].

Definition 4 For a given 3D mesh M its convexity measure $C_2(M)$ is defined as

$$C_2(M) = \min_{\alpha, \beta, \gamma \in [0, 2\pi]} \frac{Pview(M, \alpha, \beta, \gamma)}{Pface(M, \alpha, \beta, \gamma)}, \quad (4)$$

where $Pview(M, \alpha, \beta, \gamma)$ and $Pface(M, \alpha, \beta, \gamma)$ are $Pview$ and $Pface$ of M after rotating α , β , and γ with respect to x , y , and z axes, respectively. $Pface$ is the summed area of mesh faces projected onto the three orthogonal planes, YOZ , ZOX , and XOY , with $Pface = Pface_x + Pface_y + Pface_z$; while $Pview$ is the summed area of mesh silhouette images projected onto six faces

of its bounding box parallel to the orthogonal planes, with $Pview = 2(Pview_x + Pview_y + Pview_z)$. Thus, it is noticeable that there exists an inequality $Pface \geq Pview$ for any mesh, and that they are equal only if a mesh is convex. Convexity is measured as a minimum value that is sought by rotating the mesh at variant angles.

Since the calculation of C_2 is a nonlinear optimization problem that traditional methods cannot deal with, a genetic algorithm is used to help seek the minimum value of C_2 . Nevertheless, the genetic algorithm is computationally expensive, and requires a plethora of iterations to reach an optimum. Therefore, a computation-friendly measure enabling efficient convexity evaluation is yet to be studied.

In this paper, we propose a heuristic convexity measure for 3D meshes. Our measure is still projection-based, but computes the summed area ratio of projected mesh silhouette images and mesh faces only once, just along the principal directions of the mesh, followed by a correction process based on mesh slicing, rather than optimizing the ratio in iterations with the genetic algorithm, accelerating the overall computation by some an order of magnitude. Compared with C_2 , our measure gives a more reasonable evaluation to bony meshes. Compared with C_1 , our measure can sense the difference of the two meshes displayed in Figure 1, and can detect slim dents that C_1 can do nothing with. Moreover, the new measure has a better performance than both C_1 and C_2 in 3D mesh retrieval.

2 Heuristic convexity measurement

To accelerate the computation of C_2 , this paper proposes to project a given 3D mesh onto the orthogonal 2D planes in a certain direction only once. The philosophy adopted here is that in this direction an initial estimate of mesh convexity is first approximated and then nicely corrected to approach the *defacto* convexity. Such a scenario can be formulated as

$$C_3(M) = Corr\left(\frac{Pview(M \cdot R)}{Pface(M \cdot R)}\right), \quad (5)$$

where R represents the rotation matrix for the initial estimation; $Corr(\cdot)$ indicates the correction process subsequently applied.

2.1 Initial estimation along principal directions

As stated above, initial estimation should result in a value as close to the *defacto* mesh convexity as possible by rotating the mesh with R only once. However, it is

difficult to figure out how much a mesh should be rotated without *a priori* knowledge. To this end, we propose a statistical method for initial estimation. From **Definition 4** we know that C_2 can reach a minimum by rotating a mesh to a certain direction. Explicitly speaking, this minimum value exists only if faces of the mesh are projected onto three orthogonal planes having a maximum overlapping area. Therefore, our goal is to approximate this minimum convexity value using the projection-based approach by rotating the mesh to a certain direction that ensures a large overlapping area of the mesh faces. Moreover, this direction should make our convexity measure invariant under similarity transformations.

Here we select Principal Component Analysis (PCA) for initial estimation mainly for three reasons. First, the mathematical meaning of PCA is to transform a given set of data to a new coordinate system having the greatest variances of data along the principal directions. To this end, PCA ensures that the mesh faces are projected onto three orthogonal planes with a relatively larger overlapping area, leading to a relatively smaller initial estimate, closer to the *defacto* mesh convexity. Second, for the following correction process, cross sections sliced along the principal directions can best characterize the geometric detail of the mesh. Third, PCA normalizes meshes and makes our convexity measure invariant under similarity transformations. In other words, an initial rotation of the mesh to an arbitrary angle will not make our convexity measure variant under similarity transformations.

Let E denote the matrix for the eigenvectors of the covariance matrix of mesh vertices. Replacing R with E , the initial estimation can be rewritten as

$$C_e(M) = \frac{Pview(M \cdot E)}{Pface(M \cdot E)}. \quad (6)$$

Figure 3 depicts the silhouette images of some meshes projected along the principal axes, where the principal directions of the meshes conform well with the human intuition and the values of C_e are close to those of C_2 .

2.2 Correction to initial estimation

In this paper, we propose to correct the initial estimation of convexity by slicing a mesh model into a series of cross sections. If we treat cross sections of a 3D mesh along its principal axes as normal 2D shapes, then a 2D convexity measure can be used to help offset the precision loss caused by the initial estimation with PCA. Take a symmetric and elongated 3D mesh as an

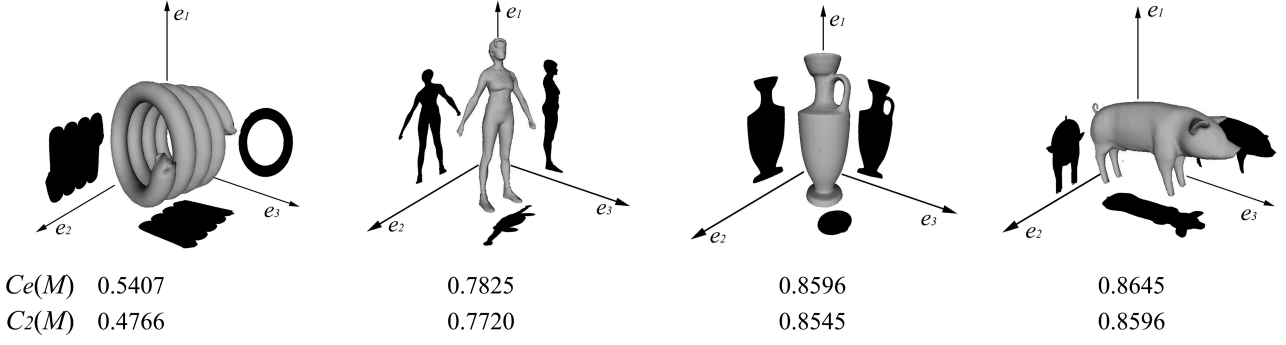


Fig. 3 Mesh silhouettes along the principal axes.

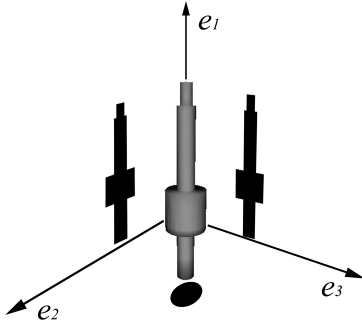


Fig. 4 An example whose convexity is overestimated by PCA.

example, as shown in Figure 4, where the PCA-based projections perfectly align with the human intuition. According to Equation 6, the initial estimate of the mesh convexity is close to 1. However, this mesh appears non-convex. As can be observed from Figure 4, one of the three silhouettes is convex, while the others are not. There is some concavity omitted by the PCA-based initial estimation, which can be recovered from cross sections of the mesh. Therefore, a 2D convexity measure to the cross sections of the 3D mesh can be readily used to correct the initial estimation of 3D convexity.

Similar to CT slicing in medical science, we slice the mesh into a sequence of 2D shapes in equal intervals along principal directions of the mesh, as illustrated shown in Figure 5. We use convexity values of these 2D shapes to compute the correction factor of each principal direction. Here we choose the area-based measure c_1 for the computation of the correction factors.

According to **Definition 1**, given that the 3D mesh is sliced along each principal direction into $N + 1$ equally spaced 2D cross sections, we can denote the general form of the correction factor r along each principal di-

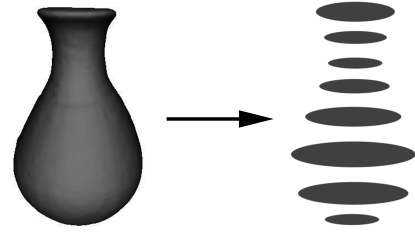


Fig. 5 An illustration of mesh slicing in one direction.

rection as

$$r = \frac{\sum_{i=0}^N \text{Area}(s_i)}{\sum_{i=0}^N \text{Area}(\text{CH}(s_i))} \quad (7)$$

where $\text{Area}(s_i)$ is the area of the i th slice, and $\text{Area}(\text{CH}(s_i))$ is the area of its corresponding convex hull. Since $\text{Area}(s_i)$ is inconvenient to compute in practice, we turn the computation of $\text{Area}(s_i)$ into that of volume as follows. We multiply both the numerator and denominator of Equation 7 by a slice step length l_{step} as

$$r = \frac{\sum_{i=0}^N \text{Area}(s_i) l_{\text{step}}}{\sum_{i=0}^N \text{Area}(\text{CH}(s_i)) l_{\text{step}}} \approx \frac{\text{Volume}(M)}{l_{\text{step}} \sum_{i=0}^N \text{Area}(\text{CH}(s_i))} \quad (8)$$

In order to make slices retain more geometric details, we compute the average edge length of the mesh, and set the slice step as a half of the average edge length. To lighten the computational burden, we cap the slice number by setting a threshold, N_{max} . The slice number along some principal direction is, thus, derived

$$N = \begin{cases} N_{\text{max}}, & \text{if } L/l_{\text{step}} \geq N_{\text{max}} \\ L/l_{\text{step}}, & \text{if } L/l_{\text{step}} < N_{\text{max}}, \end{cases} \quad (9)$$

where L is the projection length along each principal direction; N_{max} can be applied to all three principal directions. In this paper, N_{max} is by default set to 100,

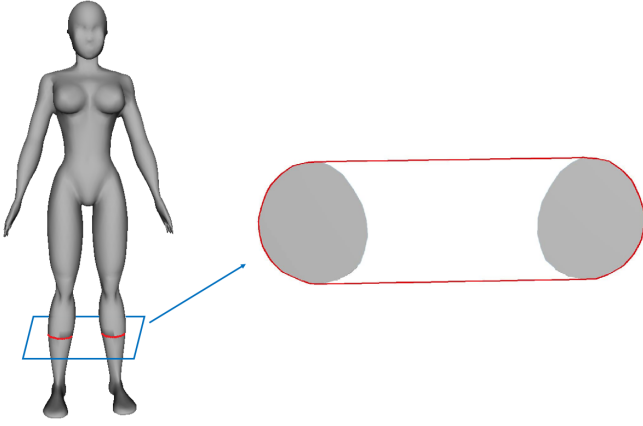


Fig. 6 A negative example of using c_2 where the close-up of a cross section into the legs shows that the value of c_2 is larger than 1.

a quantity sufficient for all the 3D meshes in our experiments. Therefore, the correction factors for three principal directions can be written as

$$r_j = \frac{\text{Volume}(M)}{l_{step} \sum_{i=0}^{N_j} \text{Area}(\text{CH}(s_i))}, j = \{x, y, z\} \quad (10)$$

Note that not every 2D convexity measure can be applied to correcting the initial estimation. For example, one may argue whether the boundary-based measure c_2 can be used to replace c_1 . One natural consideration for the replacement of Equation 7 with c_2 is

$$r' = \frac{\sum_{i=0}^N \text{Per}(\text{CH}(s_i))}{\sum_{i=0}^N \text{Per}(s_i)} \quad (11)$$

Nevertheless, we cannot guarantee that this correction factor is definitely smaller than 1. One of the negative examples is shown in Figure 6. Due to the mesh model in the figure has limbs, the value of r' evaluated along the illustrated principal direction is greater than 1 and will undermine the final convexity to be measured. Thus, only c_1 is used in this paper for the correction of initial estimation.

Now we can give our new convexity measure an ultimate definition.

Definition 5 For a given 3D mesh M its convexity measure $C_3(M)$ is defined as

$$C_3(M) = \frac{\text{Pview}(M \cdot E, r)}{\text{Pface}(M \cdot E)}. \quad (12)$$

where

$$\text{Pview}(M \cdot E, r) = 2(r_x \text{Pview}_x(M \cdot E) + r_y \text{Pview}_y(M \cdot E) + r_z \text{Pview}_z(M \cdot E)) \quad (13)$$

$$\begin{aligned} \text{Pface}(M \cdot E) &= \text{Pface}_x(M \cdot E) + \\ &\text{Pface}_y(M \cdot E) + \text{Pface}_z(M \cdot E) \end{aligned} \quad (14)$$

Theorem 1 1. $C_3(M)$ distributes in the range $(0, 1]$; 2. $C_3(M) = 1$ only when the mesh is convex; 3. $\inf_{M \in \Pi} (C_3(M)) = 0$, where Π denotes the set of all meshes; 4. $C_3(M)$ is invariant under similarity transformations.

Proof If M is convex, all the 2D slices along its principal directions must be convex. Thus, $r_x = r_y = r_z = 1$. Because $\text{Pview}(M \cdot E, r) = \text{Pface}(M \cdot E)$, it always holds $C_3(M) = 1$. When M is non-convex, there must be some non-convex slices, and thus $0 < r_x, r_y, r_z \leq 1$. Furthermore, because $\text{Pview}(M \cdot E) \leq \text{Pface}(M \cdot E)$ always holds, combining Equation 12, 13 and 14 we have $C_3(M) \leq 1$. Since $C_3(M)$ is computed along principal directions of the mesh, it is invariant under rotation and translation. Because $C_3(M)$ is a ratio, it is invariant under scaling too. Hence, the fourth condition of **Theorem 1** is satisfied. It is worth noting that an initial rotation of the mesh to an arbitrary angle will not make our convexity measure variant under rotation transformation thanks to PCA. Were PCA not applied, the values of C_3 would change as the mesh rotates. Some examples of such a hypothesis are shown in Figure 7, where rotations are made with respect to the z axis.

In order to prove the third condition of **Theorem 1**, we employ a hollow cube shown in Figure 8, where a indicates the edge length of the cube, and b denotes the edge length of the hollow. Then, we have

$$r_x = r_y = r_z = \frac{a^3 + 2b^3 - 3ab^2}{a^3}. \quad (15)$$

When increasing b to approach a , we have

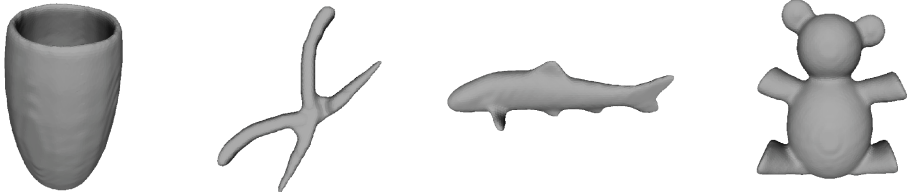
$$\lim_{b \rightarrow a} C_3(M) = 0. \quad (16)$$

The third condition of **Theorem 1** is, therefore, proved.

Furthermore, if decreasing b to 0, the hollow cube turns completely convex as a proper cube, and $C_3(M) = 1$.

Another interesting finding is that if applying Lian's measure to the same procedure above with the same hollow cube, we have

$$\begin{aligned} C_2(M) &= \min_{\alpha, \beta, \gamma \in [0, 2\pi]} \frac{\text{Pview}(M, \alpha, \beta, \gamma)}{\text{Pface}(M, \alpha, \beta, \gamma)} \\ &= \frac{a + b}{a + 3b} \end{aligned} \quad (17)$$



<i>In original direction</i>	0.1618	0.4109	0.7877	0.6578
<i>Rotation of 30°</i>	0.1599	0.5416	0.8303	0.7018
<i>Rotation of 45°</i>	0.1597	0.5588	0.8448	0.7082

Fig. 7 Without PCA convexity values calculated by C_3 change as the meshes rotate.

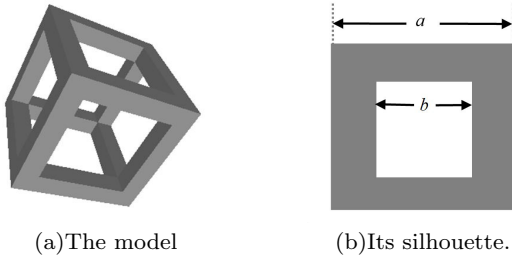


Fig. 8 A hollow cube

When decreasing b to 0, $C_2(M) = 1$. However, if inversely increasing b to approach a , Lian’s measure returns

$$\lim_{b \rightarrow a} C_2(M) = \lim_{b \rightarrow a} \frac{a+b}{a+3b} = \frac{1}{2}. \quad (18)$$

This means that the hollow cube with extremely slim bars is considered as infinitely close to being absolutely concave by C_3 , but not by C_2 .

3 Experimental evaluations

In this section, we perform experimental tests with some publicly accessible databases. The results produced by C_1 , C_2 , and C_3 are quantitatively evaluated and compared in Section 3.1. Then an application of the convexity measures in 3D mesh retrieval is performed and analyzed in Section 3.2. The computational efficiency of our method is also experimentally demonstrated in Section 3.3.

3.1 Quantitative evaluations

To demonstrate the effectiveness of our convexity measure, we apply it to two commonly used mesh databases,

the McGill Articulated 3D Shape Benchmark [19] and Princeton Benchmark [3]. Before we carry out the tests, all the meshes have to be normalized by translating their origins to the mesh centroids.

Figure 9 shows a quantitative evaluation of different measures on a group of meshes ranked by C_3 . It can be seen that for those bony meshes, such as the 1st, 3rd, 6th, 7th, and 8th meshes, their convexity values evaluated by C_3 are lower than those of C_2 , better reflecting the reality. It is also observed that C_1 hardly senses the slim dent in the 18th mesh, which is, however, noticed by both C_2 and C_3 . It is worth noting that the convexity of the sphere evaluated by C_3 is 0.9744 due to the approximation introduced by Equation 8.

Figure 10 shows a group of hand gestures ordered by C_3 , with their convexity values calculated by C_1 , C_2 , and C_3 . For the gestures with five straight fingers, their convexity values calculated by C_1 and C_3 are higher than those with fingers bending. However, C_2 cannot distinguish this nuance.

Moreover, we extend our test to the hollow cube with both C_2 and C_3 . As can be seen in Figure 11, when b goes wider, the values of two measures become smaller. Note that the convexity values computed by C_3 range from 0 to 1 while the values computed by C_2 are between 0.5 and 1. Here the value of b is in turn set to 0, $0.2a$, $0.5a$, and $0.8a$ for the hollow cube.

A cube with a deep dent shown in Figure 12 is also tested by comparing C_3 with C_1 . When $n \rightarrow 0$, the volume for the dent approaches 0. The convexity computed by $C_1(M)$ is,

$$\begin{aligned} \lim_{n \rightarrow 0} C_1(M) &= \lim_{n \rightarrow 0} \frac{\text{Volume}(M)}{\text{Volume}(CH(M))} \\ &= \lim_{n \rightarrow 0} \frac{m^3 - \frac{1}{2}m^2n}{m^3} = 1 \end{aligned} \quad (19)$$

























1	2	3	4	5	6	7	8
							
$C_1(M)$ 0.2320	0.4001	0.1824	0.4677	0.5595	0.0723	0.1279	0.1472
$C_2(M)$ 0.5239	0.5793	0.7175	0.4766	0.3545	0.7463	0.7842	0.9072
$C_3(M)$ 0.0550	0.1508	0.1760	0.2179	0.2244	0.2408	0.2574	0.2952
9	10	11	12	13	14	15	16
							
$C_1(M)$ 0.1565	0.1598	0.1732	0.1527	0.3824	0.1982	0.2871	0.5225
$C_2(M)$ 0.7755	0.7720	0.9173	0.8686	0.8064	0.8126	0.9346	0.8545
$C_3(M)$ 0.3101	0.3706	0.4219	0.4477	0.4540	0.4577	0.5177	0.7024
17	18	19	20	21	22	23	24
							
$C_1(M)$ 0.4540	0.9700	0.5478	0.5446	0.6555	0.8310	1.0000	1.0000
$C_2(M)$ 0.8848	0.7169	0.8384	0.9321	0.9599	0.9403	0.9999	0.9999
$C_3(M)$ 0.7201	0.7208	0.7741	0.8265	0.8529	0.9528	0.9744	0.9999

Fig. 9 Meshes of the quantitative evaluation.









							
$C_1(M)$ 0.2914	0.2942	0.3104	0.3954	0.2339	0.4700	0.5200	0.6195
$C_2(M)$ 0.8091	0.8070	0.8410	0.8026	0.8364	0.8694	0.8077	0.8251
$C_3(M)$ 0.4042	0.4315	0.4337	0.4403	0.4546	0.6095	0.6575	0.6722

Fig. 10 Hand gestures ordered by C_3 .

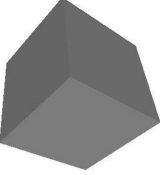
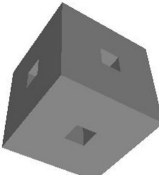


			
$C_2(M)$ 0.9999	0.7487	0.5967	0.5239
$C_3(M)$ 0.9999	0.6719	0.2995	0.0550

Fig. 11 The hollow cube with b broadening.

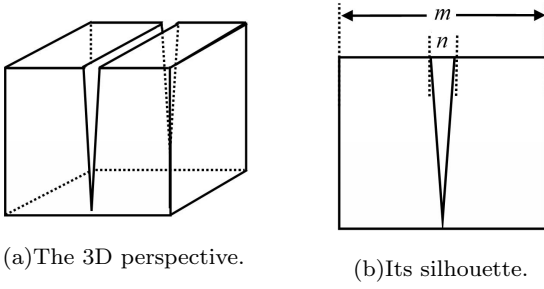


Fig. 12 A cube with a cut.

However, this value cannot reflect the cut existing in the cube. C_3 can detect this cut instead, which is computed as

$$\lim_{n \rightarrow 0} C_3(M) = \frac{6m^2}{8m^2} = 0.75 \quad (20)$$

The new measure can also handle the problem shown in Fig.1. The convexity values of the two meshes calculated by C_3 are, respectively, 0.6756 and 0.7483.

3.2 3D mesh retrieval

We apply C_1 , C_2 , and C_3 to non-rigid 3D mesh retrieval. The meshes are selected from the McGill articulated 3D shape benchmark, consisting of 10 categories of 255 watertight meshes. The retrieval performance is evaluated by four quantitative measures (NN, 1-Tier, 2-Tier, DCG) [16]. We use convexities computed by the three measures to represent the 3D meshes and employ the L_1 norm to calculate the dissimilarity between two features. The results are shown in Table 1. Note that all these convexities are calculated after the 3D meshes are converted into their canonical forms. Here, we use a method introduced in [5] to construct their feature-preserved canonical forms of the 3D meshes. As shown in Figure 13, the meshes for the same species may appear in quite different poses but have similar canonical forms. Thereby following the calculation of canonical forms, all feature extraction methods, even those specifically designed for rigid 3D meshes, can be employed to extract isometry-invariant shape descriptors from non-rigid objects. The results in Table 1 show that our measure outperforms the others in terms of retrieval rate. However, representing 3D meshes by a solo convexity measure may result in relatively poor discriminations. Instead when we use both C_3 and C_1 as dual features, better performance results are obtained.

Table 1 Retrieval performance of our measure and the other convexity measures evaluated on the McGill database.

	NN	1-Tier	2-Tier	DCG
C_1	26.3%	26.0%	43.7%	59.8%
C_2	25.9%	26.3%	45.9%	60.4%
C_3	34.1%	26.8%	50.3%	61.9%
C_1 and C_3	43.9%	33.0%	54.1%	66.0%

3.3 Computational efficiency

C_2 is computationally expensive due to the genetic algorithm with 50 individuals and 200 evolution generations [6], especially when the number of vertices in the mesh is large. In contrast, our measure needs to capture silhouette images from the frame buffer only once. Table 2 compares the time consumed by both the measures on some typical meshes ordered in vertex number. It can be seen that C_3 accelerates the overall computations by approximately an order of magnitude. The whole experimental environment is under Visual Studio 2010 in a laptop configured with Intel Core i5 CPU and 6G RAM.

4 Conclusions

Aiming to address the problems in the extant measures, we have proposed a heuristic convexity measure for 3D meshes in this paper. Our measure projects only once a given 3D mesh onto orthogonal 2D planes along its principal directions for an initial estimation of mesh convexity, followed by a correction process based on the 2D area-based convexity measurement of mesh cross sections. To this end, our measure avoids the tedious genetic algorithm adopted by C_2 , enabling highly efficient convexity measurement for 3D meshes and shortening the computational time of C_2 by some an order of magnitude. Compared with C_1 that has difficulties in detecting slim dents and sensing the difference of meshes with the identical mesh and convex hull volumes, our measure can successfully handle these issues. The experimental results have also demonstrated the advantage of our measure in 3D mesh retrieval against both C_1 and C_2 . Since our measure is computationally inexpensive, it is ready for use in many other graphics applications, such as 3D mesh partitioning, and classification, *etc.*

5 Acknowledgements

The authors would like to thank Dr. Lian for his supports and patience in answering our questions. The au-

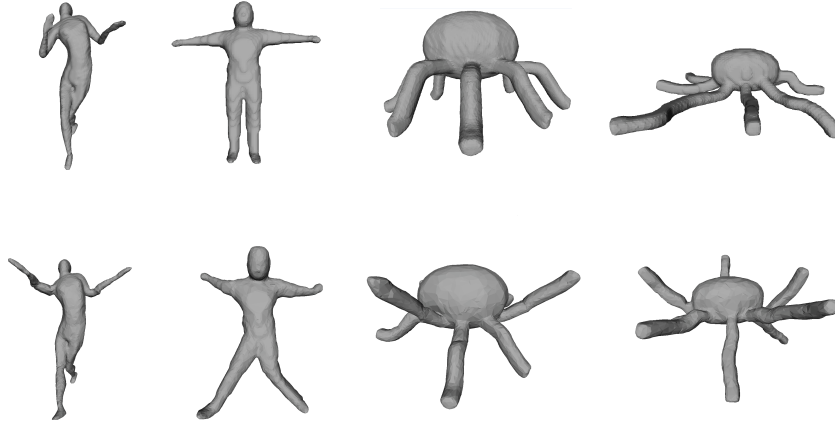







Fig. 13 Canonical Forms of the meshes. The first row shows the original non-rigid models, while the second row shows their feature-preserved 3D canonical forms.

Table 2 Comparison of time consumptions

Meshes	Vertex number	C_3 (in second)	C_2 (in second)	
			One iteration	Total
	673	11.34	0.0648	648
	4463	45.11	0.0905	905
	9261	86.08	0.1400	1400
	14872	148.4	0.1700	1700
	34817	316.1	0.3115	3115

thors would also like to thank the reviews for their precious time and constructive comments.

References

- Asafi, S., Goren, A., Cohen-Or, D.: Weak convex decomposition by lines-of-sight. *Computer Graphics Forum* **32**(5) (2013)
- Chalmeta, R., Hurtado, F., Sacristan, V., Saumell, M.: Measuring regularity of convex polygons. *Computer-Aided Design* **45**, 93–104 (2013)
- Chen, X., Golovinskiy, A., Funkhouser, T.: A benchmark for 3D mesh segmentation. *ACM Transactions on Graphics* **28** (2009)
- Ghosh, M., Amato, N.M., Lu, Y., Lien, J.M.: Fast approximate convex decomposition using relative concavity. *Computer-Aided Design* **45**(2), 494–504 (2013)
- Lian, Z., Godil, A.: A feature-preserved canonical form for non-rigid 3D meshes. In: *International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*, pp. 116–123 (2011)
- Lian, Z., Godil, A., Rosin, P.L., Sun, X.: A new convexity measurement for 3D meshes. In: *Computer Vision and Pattern Recognition*, pp. 119–126 (2012)
- Lian, Z., Rosin, P.L., Sun, X.: Rectilinearity of 3D meshes. *International Journal of Computer Vision* **89**(2-3), 130–151 (2010)
- Liu, H., Liu, W., Latecki, L.J.: Convex shape decomposition. In: *Computer Vision and Pattern Recognition*, pp. 97–104. *IEEE* (2010)
- Pao, H.K., Geiger, D., Rubin, N.: Measuring convexity for figure/ground separation. In: *IEEE International Conference on Computer Vision*, vol. 2, pp. 948–955. *IEEE* (1999)
- Rahtu, E., Salo, M., Heikkilä, J.: A new convexity measure based on a probabilistic interpretation of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **28**, 1501–1512 (2006)
- Rosin, P.L.: Measuring shape: ellipticity, rectangularity, and triangularity. *Machine Vision and Applications* **14**(3), 172–184 (2003)
- Rosin, P.L.: A two-component rectilinearity measure. *Computer Vision and Image Understanding* **109**, 176–185 (2008)
- Rosin, P.L.: Classification of pathological shapes using convexity measures. *Pattern Recognition Letters* **30**, 570–578 (2009)
- Rosin, P.L., Mumford, C.L.: A symmetric convexity measure. *Computer Vision and Image Understanding* **103**, 101–111 (2006)

15. Sarfraz, M., Ridha, A.: Content-based image retrieval using multiple shape descriptors. In: IEEE/ACS International Conference on Computer Systems and Applications, pp. 730–737 (2007)
16. Shilane, P., Min, P., Kazhdan, M., Funkhouser, T.: The princeton shape benchmark. In: Shape Modeling International, vol. 105 (2004)
17. Song, Y.Z., Pickup, D., Li, C., Rosin, P., Hall, P.: Abstract art by shape classification. IEEE Transactions on Visualization and Computer Graphics **19**(8), 1252–1263 (2013)
18. Z, R., J, Y., W, L.: Minimum near-convex shape decomposition. IEEE Transactions on Software Engineering **35**(10), 2546–2552 (2013)
19. Zhang, J., Kaleem, S., Diego, M., Sven, D.: Retrieving articulated 3D models using medial surfaces. Machine Vision and Applications **19**, 261–275 (2008)
20. Zimmer, H., Campen, M., Kobbelt, L.: Efficient computation of shortest path-concavity for 3D meshes. In: Computer Vision and Pattern Recognition, pp. 2155–2162 (2013)
21. Zunic, J., Rosin, P.L.: Rectilinearity measurements for polygons. IEEE Transactions on Pattern Analysis and Machine Intelligence **25**, 1193–1200 (2003)
22. Zunic, J., Rosin, P.L.: A new convexity measure for polygons. IEEE Transactions on Pattern Analysis and Machine Intelligence **26**, 923–934 (2004)