Int J Comput Vis (2010) 89: 130–151
DOI 10.1007/s11263-009-0295-0

Rectilinearity of 3D Meshes

Zhouhui Lian · Paul L. Rosin · Xianfang Sun

Received: 25 September 2008 / Accepted: 14 September 2009 / Published online: 23 September 2009

Abstract In this paper, we propose and evaluate a novel shape measure describing the extent to which a 3D polygon mesh is rectilinear. The rectilinearity measure is based on the maximum ratio of the surface area to the sum of three orthogonal projected areas of the mesh. It has the following desirable properties: 1) the estimated rectilinearity is always a number from (0,1]; 2) the measure is invariant under scale, rotation, and translation; 3) the 3D objects can be either open or closed meshes, and we can also deal with degenerate meshes; 4) the measure is insensitive to noise, stable under small topology errors, and robust against face deletion and mesh simplification. Moreover, a genetic algorithm (GA) can be applied to compute the approximate rectilinearity efficiently. We find that the calculation of rectilinearity can be used to normalize the pose of 3D meshes, and in many cases it performs better than the principal component analysis (PCA) based method. By applying a simple selection criterion, the combination of these two methods results in a new pose normalization algorithm which not only provides a higher successful alignment rate but also corresponds better with intuition. Finally, we carry out several experiments showing that both the rectilinearity based pose normalization preprocessing and the combined signatures, which consist of the rectilinearity measure and other shape descriptors, can significantly improve the performance of 3D shape retrieval.

Keywords Rectilinearity · Shape measurement · 3D shape retrieval · Pose normalization

Z. Lian (✉) · X. Sun
School of Automation Science and Electrical Engineering, Beihang University, Beijing, P.R. China
e-mail: lianzhouhui@yahoo.com.cn

X. Sun
e-mail: Xianfang.Sun@cs.cardiff.ac.uk

Z. Lian · P.L. Rosin · X. Sun
Cardiff School of Computer Science, Cardiff University, Wales, UK

P.L. Rosin
e-mail: Paul.Rosin@cs.cardiff.ac.uk

1 Introduction

How to quantify shape is an important research area with many applications in computer vision (Loncaric 1998; Zhang and Lu 2004; Tangelder and Veltkamp 2008; Yang et al. 2007). Usually it is preferable if the shape measures have direct intuitive meanings, for example, compactness (Haralick 1974), triangularity (Rosin 2003), ellipticity (Proffitt 1982), rectangularity (Rosin 1999), rectilinearity (Žunić and Rosin 2003; Rosin 2008), convexity (Žunić and Rosin 2004), symmetry (Leou and Tsai 1987) and chirality (Petitjean 2003). Having intuitive shape measures means that the results of shape retrieval queries can be explained in a manner readily understandable by the user. Although many approaches have been proposed for these kinds of 2D shape measures, many of them cannot be directly generalized to 3D.

Up to now, just a few global shape descriptors with direct meanings for 3D models have been developed. In this paper, we investigate the rectilinearity of 3D meshes. The potential benefits of rectilinearity were demonstrated recently through a psychophysical experiment which showed that humans appeared to discriminate artifacts and animals based on rectilinearity (Levin et al. 2001). The basic idea of rectilinearity is demonstrated by a simple example using some commonly seen objects shown in Fig. 1. Intuitively, we would consider

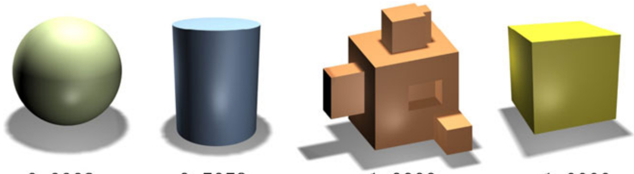

Fig. 1 Sphere, cylinder, rectilinear object and cube; underneath are the corresponding rectilinearity values estimated by our method

the leftmost sphere as the least rectilinear model, while the second one, a cylinder, is more rectilinear than the sphere, and the third object, as well as the rightmost cube, should have the greatest value of rectilinearity. We would like to define a shape measure describing the extent to which a 3D model is rectilinear, which for convenience should be a value between 0 and 1.

The most relevant work is presented in the paper (Žunić and Rosin 2003) where Žunić and Rosin proposed two rectilinearity measurements for 2D polygons. They used values which consist of the ratio of the L_2 norm perimeter and L_1 norm perimeter of a polygon. Motivated by their research, this paper introduces a novel shape measure describing the extent to which a 3D polygon mesh is rectilinear. The rectilinearity measure is based on the maximum ratio of the surface area to the sum of three orthogonal projected areas of the mesh. To the best of our knowledge, our work (Lian et al. 2008) is the first to quantify the rectilinearity of 3D objects, and this article is the extended version of the conference paper.

The rectilinearity measurement proposed in this paper has following advantages:

1. The measure corresponds with the intuitive notion of rectilinear 3D shapes and the estimated rectilinearity is always a number from (0, 1];
2. The measurement is insensitive to noise, stable under small topology errors, and robust against face deletion and mesh simplification;
3. The measurement can be applied to both open and closed meshes. Moreover, we can also deal with poor quality meshes, which often occur in practice, such as flipped normals, degenerate elements, zero area triangles and so on;
4. The measure is straightforward to compute and is invariant under scale, rotation, and translation.

The rectilinearity of 3D meshes can be applied in several different fields, such as, computer vision, artificial intelligence and pattern recognition. This paper demonstrates the utility of rectilinearity calculation for *3D shape retrieval* and *pose normalization*.

The explosion in the number of available 3D models has led to the development of 3D shape retrieval systems that, given a query object, retrieve similar 3D models (Tangelder and Veltkamp 2008). The rectilinearity measure can be directly used as a shape descriptor for 3D shape retrieval, and our experiments show that the combination of rectilinearity measure and other descriptors can markedly improve 3D shape retrieval performance.

Since a large number of shape descriptors are not invariant under similarity transformations (scaling, translation and rotation) pose normalization is often necessary during the preprocessing stages of 3D shape retrieval systems. To normalize a 3D mesh for scale, one possible approach is to scale the average distance of the points on its surface to the center of mass to a constant, while translation invariance is accomplished by translating the center of mass to the origin. Securing rotation invariance is usually more difficult than securing scale and translation invariance. The most prominent tool for accomplishing rotation invariance is principal component analysis (PCA) (Paquet and Rioux 1999; Vranić et al. 2001). The PCA algorithm is fairly simple and efficient. However, it may erroneously assign the principal axes and produce inaccurate normalization results, especially when the eigenvalues are equal or close to each other, which is its intrinsic drawback that cannot be overcome easily (Tangelder and Veltkamp 2003). Moreover, many principal axis based pose normalization results conflict with our intuition. For example, applying PCA, a cabinet will be normalized as Fig. 2(a), while it is obvious that the normalization result in Fig. 2(b) corresponds better with human perception. Usually, when we express or design a 3D model in the 2D domain, drawing three images from the left, top and front directions, which is termed the three-view-drawing method, might be the first choice. At the same time, we may recall that when drawing these images, the object should be placed in a properly aligned pose. So, it is reasonable to infer that the pose suitable for drawing three-view images could set up a canonical coordinate system for shape matching. We find that, compared to principal axis based methods, the calculation of the rectilinearity measure provides a better tool for pose normalization of a huge number of models, especially whose shape characteristics are dominated by rectilinearity. Furthermore, we demonstrate that, using a minimum-area selection criterion, the combination of PCA and the rectilinearity based method can properly normalize the pose for almost all models (more than 98.2%, which has been approximately verified by an experiment with 39 participants) in the Princeton Shape Benchmark (PSB) database (Shilane et al. 2004). However, there are still some models that can neither be correctly normalized by PCA nor rectilinearity.

Assume that all models can be normalized to draw three-view images corresponding well with intuition, then the positions of three coordinate axes are fixed, that means there are 24 possible choices for the right-handed coordinate sys-

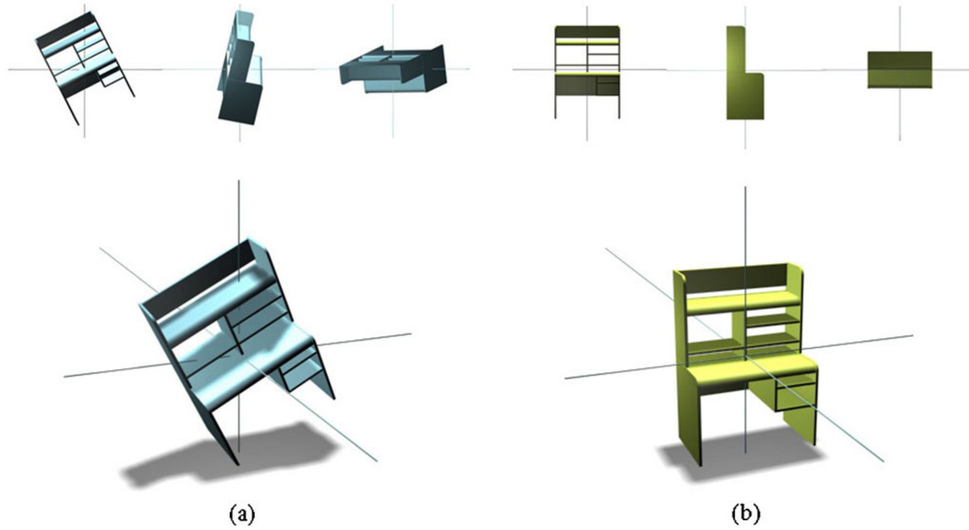

Fig. 2 Normalized cabinets. Figure (**a**) and figure (**b**) show two different poses of a cabinet normalized by the PCA based method and our method, respectively. *Above* are their corresponding front, left, and top views

tem, namely, 24 different poses are still plausible for a normalized model. Using different resolution geodesic spheres generated from the unit octahedron, we can extract different number of shape descriptors from the depth-buffers or silhouettes captured on their vertices. Moreover, according to the properties of these geodesic spheres, corresponding efficient shape matching can be carried out. The complexity of this matching algorithm is O(N), while the well-known LFD method (Chen et al. 2003) is O(N^3) (here N denotes the number of views compared). Experiments show that our method not only dramatically outperforms the LFD method in retrieval speed but also in the capability of discrimination.

The major contributions of this paper are threefold. First, we provide the complete definition of a rectilinearity measure describing the extent to which a 3D mesh is rectilinear. Second, we describe a novel pose normalization scheme whose result is suitable for drawing standard three-view images, and thus corresponds better with our intuitive perception than other methods. Third, we demonstrate how to apply the calculation of rectilinearity to further improve the performance of 3D shape retrieval.

The remainder of the paper is organized as follows. Section 2 discusses previous work. Section 3 describes notations and basic concepts of rectilinearity for 3D meshes. Then, we present the definition of a rectilinearity measure for 3D meshes and demonstrate how to calculate it approximately in Sect. 4 where corresponding theorems are also explicitly proved. Afterwards, Sect. 5 illustrates some experimental results which validate the effectiveness and robustness of our shape measurement. Furthermore, applications of 3D rectilinearity to pose normalization and shape retrieval are introduced in Sects. 6 and 7, respectively. Finally, we provide the conclusion of this paper as well as some future research directions in Sect. 8.

2 Related Work

Shape measurement So far, just a few global shape descriptors with intuitive meanings for 3D models have been proposed. Zhang and Chen (2001a) introduced methods to calculate global features such as volume-area ratio, statistical moments, and Fourier transform coefficients efficiently and then applied these descriptors for 3D shape retrieval (Zhang and Chen 2001b). Paquet et al. (2000) used the bounding box and other descriptors for 3D shape matching. Fink and Wood (1996) developed a restricted-orientation convexity which was defined in terms of the intersection of a geometric object with lines parallel to the elements of a fixed orientation set. Corney et al. (2002) described convex-hull based indices like hull crumpliness, hull packing, and hull compactness to carry out a preliminary coarse filtering of candidates. Kazhdan et al. (2003) presented a 3D objects' reflective symmetry descriptor as a 2D function associating a measurement of reflective symmetry to every plane through the model's centroid. Bribiesca (2008) proposed a compactness measure which corresponds to the sum of the contact surface areas of the face-connected voxels for 3D shapes. However, most of these 3D shape measures cannot be applied to open meshes and they usually need voxelization which involves expensive computation.

3D shape retrieval As the number of 3D models is increasing rapidly, "shape-based 3D model retrieval" methodology, concentrating on the representation, recognition and matching of 3D models based on their intrinsic shapes, has become a new hot topic in computer vision (Yang et al. 2007). A growing number of researchers have been involved in this area, and they have already made substantial progress. Feature extraction is the key issue for an efficient retrieval system and a considerable number of shape descriptors (Bustos et al. 2005), such as: D1 (Ankerst et al. 1999), D2

(Osada et al. 2002), spherical harmonic descriptor (Kazhdan et al. 2003), skeleton based shape descriptor (Sundar et al. 2003), and view based features (Chen et al. 2003) have been proposed. For more details, we refer the reader to recent surveys (Shilane et al. 2004; Bustos et al. 2005; Yang et al. 2007; Tangelder and Veltkamp 2008).

Among these shape descriptors, view-based methods, considering that two models are similar when they look similar from all view angles, generally outperform others. Moreover they are suitable for implementing query interfaces using sketches. Therefore, a huge number of view-based signatures (Chen et al. 2003; Chaouch and Blondet 2006; Shih et al. 2007) have been developed to extract 2D descriptors (e.g. Zernike moment, Fourier coefficient, elevation descriptor, etc.) from the silhouettes or depth buffers captured around 3D models. As a more recent example, Chaouch and Blondet (2007) proposed a representation of a 3D model by 20 depth images rendered from the vertices of a regular dodecahedron and then a special depth sequence was developed to describe each image. The depth line descriptors were compared by dynamic programming distance (DPD) which can cope with presence of local shifting on the shape. Their experiment showed that the method using depth line descriptors and dynamic programming distance generally outperformed other state-of-the-art methods. However, dynamic programming is computationally expensive and the computational complexity of shape matching exponentially increases as the number of depth buffers grows. In the past, it is almost impractical to improve the discrimination by markedly increasing the number of viewpoints. In contrast, the multi-view based shape matching method applied in this paper has the ability to deal with huge amounts of views. The complexity of the algorithm is just O(N), therefore it provides a potential to significantly improve retrieval performance.

Since no single descriptor outperforms others in all situations (Shilane et al. 2004), a popular approach is to construct composite shape signatures, which consist of several different descriptors. For instance, Vranić (2005) described a composite 3D shape feature vector named DESIRE, which was constructed using depth buffer images, silhouettes, and ray-extents of a mesh. Ohbuchi and Hata (2006) combined both multiresolution and heterogeneous sets of shape descriptors to form new signatures. The shape descriptors were integrated via linear combinations of the distance values they produce, using either fixed or adaptive weights. Both of them demonstrated (Vranić 2005; Ohbuchi and Hata 2006) that using the combination of different descriptors could significantly improve the retrieval performance. Since our rectilinearity measure describes 3D objects in a quite different manner compared to existing signatures, thereby providing new and independent information, it is well suited to be incorporated into composite descriptors.

Also recently, transformation invariant descriptors have attracted many attentions. Laga et al. (2006) observed that the so-called rotation-invariant spherical harmonic descriptor (Kazhdan et al. 2003) varied when rotating, mainly because spherical harmonic analysis is based on latitude-longitude parameterization of a sphere which has singularities at each pole and variations of the polar axis markedly affects the spherical function. The spherical wavelet descriptor (Laga et al. 2006) they proposed is based on uniform spherical sampling and thus the energies of the wavelet transform are rotation invariant. Gal et al. (2007) introduced a pose-oblivious shape signature that is not only rotation invariant but also insensitive to deformations such as skeletal articulations. Their descriptor is a 2D histogram which is a combination of the distribution of two scalar functions: the local-diameter function and centricity function. The first function measures the diameter in the neighborhood of each vertex while the second calculates the average geodesic distance from one vertex to all other vertices on the mesh. Ruggeri and Saupe (2008) applied farthest point sampling to select a set of reference points which are evenly distributed on the surface, next calculated the geodesic distance between these reference points, normalized and stored them in a matrix from which they obtained a set of histograms. Afterwards bipartite graph matching was carried out to match two histogram sets to calculate the dissimilarity of two models. Since geodesic distances were used to construct the histograms, their method can classify and recognize objects deformed with isometric transformations, e.g., non-rigid and articulated objects in different postures. Transformation invariance is a desirable character for shape signatures, however, compared to other descriptors using pose normalization, they are often less discriminative when mainly precessing normal models. Moreover, some of them use local feature extraction and partial matching, which are computationally expensive. Therefore, effective pose normalization is still a commonly used approach to accelerate the searching speed and enhance the discrimination for 3D shape retrieval systems.

Pose normalization In 3D shape matching applications, usually we have to register two models to calculate their dissimilarity. So far, methods for model alignment would either be pose normalization or searching for the rotation that best aligns each pair of models. Although marked improvements have been made for the latter kind of approaches (Kazhdan 2007), it is still too slow to be applied in practical 3D shape retrieval systems. Therefore, pose normalization, which aims to align an object into a canonical coordinate frame is often the best choice for a large number of shape descriptors that are not rotation invariant. So far, pose normalization methods are mainly based on *principal axes* and *symmetry* of a 3D shape. PCA-alignment is the most popular method and has been widely used in applications of 3D shape retrieval (Paquet and Rioux 1999; Vranić et al. 2001; Tangelder and Veltkamp 2003). As PCA is sensitive to small shape variance and thus is unstable to find the correct principal axes, many other more robust methods have been proposed to solve the problem. For example, Krinidis and Chatzis (2008) estimated the principal axes of objects based on a physics-based deformable model that parameterizes the shape. However, as Fig. 2 shows, principal axes are not suitable for pose normalization of many models. Podolak et al. (2006) described a planar reflective symmetry transform (PRST) that captures a continuous measure of the reflectional symmetry of a shape with respect to all possible planes. They demonstrated that, compared to principal axes, it is more robust to normalize symmetric objects by principal symmetry axes. Symmetry is a very strong cue for shape orientation, however it does not necessarily imply correct orientations for asymmetry models. More recently, Chaouch and Blondet (2008) introduced a new alignment approach combining PCA techniques and symmetry properties. They also illustrated the effectiveness of their method by the improvement of retrieval results comparing to PCA-alignment. Fu et al. (2008) demonstrated how to learn upright orientation for man-made objects from their functionality-related geometric properties including static stability, symmetry, parallelism, and visibility. They presented a novel methodology for pose normalization, however, only the upright orientation of man-made objects is concerned and the prediction accuracy is too low (about 90%) to be used in practical applications. In this paper, we introduce rectilinearity, which is also a strong cue for shape orientation of 3D meshes, to provide a new effective tool to normalize 3D meshes and help to make the alignment result correspond better with intuitive perception.

Viewpoint selection How to automatically select good view images for 3D models is an important and ongoing problem, especially when dealing with a huge database of 3D models. Some methods tried to capture important views by maximizing the interesting information content using measures such as viewpoint entropy (Vazquez et al. 2003) and view saliency (Lee et al. 2005). While Podolak et al. (2006) and Yamauchi et al. (2006) selected viewpoints by measuring symmetry and similarity, respectively, to minimize visible redundant information. Recently, Shilane and Funkhouser (2007) developed a method for generating salient views that display the most distinctive region with respect to a chosen database. Fu et al. (2008) suggested that their orientation method could help to view an object in a natural way, since humans usually associate an upright orientation with objects. Similarly, our pose normalization method can place objects in the way that they are most commonly seen in our surrounding, therefore, it helps to select different kinds of salient images for different applications. For example, images captured in the directions of three axes in canonical coordinate system can be used to generate standard three-view drawings which is the most popular representation method in mechanical engineering domain.

3 Definition and Notations

In this section, we first describe a formal definition of rectilinear 3D meshes and then give some notations used in this paper. For convenience, the meshes we describe here are 3D triangle meshes, but the following definitions and theorems can also be adopted to other 3D polygon meshes.

Definition 1 A 3D mesh M is rectilinear if the angles between the normals of every two faces belong to $\{0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}\}$.

Given a 3D mesh M which consists of N triangles $\{T_1, T_2, \ldots, T_N\}$, the surface area of the mesh M is represented as $S(M)$, while three projected areas corresponding to the *YOZ*, *ZOX* and *XOY* planes are $P_x(M)$, $P_y(M)$, $P_z(M)$, respectively, defined by

$$P_x(M) = \sum_{i=1}^{N} S_{ix}, \qquad P_y(M) = \sum_{i=1}^{N} S_{iy}, \qquad P_z(M) = \sum_{i=1}^{N} S_{iz} \tag{1}$$

where S_{ix}, S_{iy}, S_{iz} are the projected areas of triangle T_i on the plane *YOZ*, *ZOX* and *XOY*, respectively (see Fig. 3).

If we rotate the coordinate frame, we will get new projected areas of the mesh M. Therefore, we will use

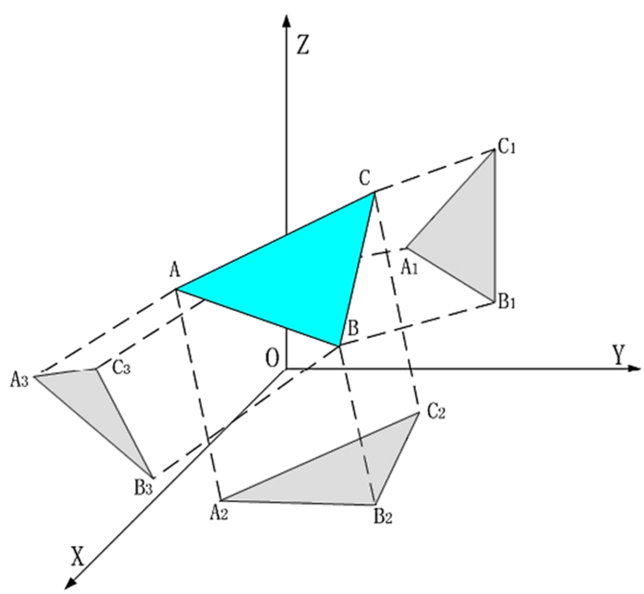

Fig. 3 Projecting a triangle on three orthogonal planes

$P_x(M,\alpha,\beta,\gamma)$, $P_y(M,\alpha,\beta,\gamma)$, $P_z(M,\alpha,\beta,\gamma)$ for these three projected areas which are obtained after successively rotating the coordinate frame around its x, y, z axes by angles α, β, γ. Here we denote the sum of these three projected areas by

$$\begin{aligned} P(M,\alpha,\beta,\gamma) &= \sum_{i=1}^{N} P(T_i,\alpha,\beta,\gamma) \\ &= P_x(M,\alpha,\beta,\gamma) + P_y(M,\alpha,\beta,\gamma) + P_z(M,\alpha,\beta,\gamma) \quad (2) \end{aligned}$$

where $P(T_i,\alpha,\beta,\gamma) = S'_{ix} + S'_{iy} + S'_{iz}$ is the sum of three projected areas $(S'_{ix}, S'_{iy}, S'_{iz})$ of the triangle T_i in the rotated coordinate frame.

Let that the original coordinates of the vertices of the triangles $\{T_1, T_2, \ldots, T_N\}$ be denoted by (x_{i0}, y_{i0}, z_{i0}), (x_{i1}, y_{i1}, z_{i1}), (x_{i2}, y_{i2}, z_{i2}), $i = 1, \ldots, N$. After successively rotating the coordinate frame around its x, y, z axes with angles α, β, γ, we get their new coordinates, represented as $(x'_{i0}, y'_{i0}, z'_{i0})$, $(x'_{i1}, y'_{i1}, z'_{i1})$, $(x'_{i2}, y'_{i2}, z'_{i2})$, $i = 1, \ldots, N$, specified by formulae

$$\begin{aligned} &(x'_{ij}, y'_{ij}, z'_{ij})^T = R(\alpha,\beta,\gamma)(x_{ij}, y_{ij}, z_{ij})^T, \\ &\quad i = 1, \ldots, N;\ j = 0, 1, 2. \end{aligned} \quad (3)$$

where $R(\alpha,\beta,\gamma)$ stands for the rotation matrix. Let

$$\begin{aligned} &\vec{X}_{ij} = (x_{ij}, y_{ij}, z_{ij}), \vec{X}'_{ij} = (x'_{ij}, y'_{ij}, z'_{ij}), \\ &\quad i = 1, \ldots, N;\ j = 0, 1, 2, \end{aligned} \quad (4)$$

Define

$$\vec{\eta}'_i = (\eta'_{ix}, \eta'_{iy}, \eta'_{iz}) = (\vec{X}'_{i1} - \vec{X}'_{i0}) \times (\vec{X}'_{i2} - \vec{X}'_{i0}). \quad (5)$$

Then, the area of the triangle T_i is

$$S_i = S'_i = \frac{1}{2}|\vec{\eta}'_i| = \frac{1}{2}\sqrt{(\eta'_{ix})^2 + (\eta'_{iy})^2 + (\eta'_{iz})^2} \quad (6)$$

and the projected areas of triangle T_i on the plane YOZ, ZOX and XOY are

$$S'_{ix} = \frac{1}{2}|\eta'_{ix}|, S'_{iy} = \frac{1}{2}|\eta'_{iy}|, S'_{iz} = \frac{1}{2}|\eta'_{iz}|, \quad (7)$$

respectively. Thus, we have

$$S_i = \sqrt{(S'_{ix})^2 + (S'_{iy})^2 + (S'_{iz})^2} \le S'_{ix} + S'_{iy} + S'_{iz}. \quad (8)$$

Using the Root Mean Square-Arithmetic Mean Inequality, we obtain

$$S'_{ix} + S'_{iy} + S'_{iz} \le \sqrt{3}\sqrt{(S'_{ix})^2 + (S'_{iy})^2 + (S'_{iz})^2} = \sqrt{3}S_i. \quad (9)$$

Thus $S_i \le S'_{ix} + S'_{iy} + S'_{iz} \le \sqrt{3}S_i$. Since

$$S(M) = \sum_{i=1}^{N} S_i = \sum_{i=1}^{N}\sqrt{(S'_{ix})^2 + (S'_{iy})^2 + (S'_{iz})^2}, \quad (10)$$

$$P(M,\alpha,\beta,\gamma) = \sum_{i=1}^{N}\left(S'_{ix} + S'_{iy} + S'_{iz}\right). \quad (11)$$

Finally, we get

$$S(M) \le P(M,\alpha,\beta,\gamma) \le \sqrt{3}S(M). \quad (12)$$

Theorem 1 *A given* 3*D mesh M is rectilinear if and only if there exists a choice of the coordinate system such that the surface area of M and the sum of three projected areas of M coincide*, *i.e.*

$$S(M) = P(M,\alpha,\beta,\gamma) \quad \textit{for some } \alpha,\beta,\gamma \in [0, 2\pi]. \quad (13)$$

Proof On the one hand, if M is rectilinear then a rotation of coordinate frame, such that all faces of M become parallel to one of three planes YOZ, ZOX, XOY, ensures the equality $S(M) = P(M,\alpha,\beta,\gamma)$, where α, β, γ are the rotation angles. On the other hand, $S(M) = P(M,\alpha,\beta,\gamma)$ implies

$$\sum_{i=1}^{N}\sqrt{(S'_{ix})^2 + (S'_{iy})^2 + (S'_{iz})^2} = \sum_{i=1}^{N}\left(S'_{ix} + S'_{iy} + S'_{iz}\right). \quad (14)$$

Furthermore, we derive

$$\sqrt{(S'_{ix})^2 + (S'_{iy})^2 + (S'_{iz})^2} = S'_{ix} + S'_{iy} + S'_{iz} \quad (15)$$

$$\Rightarrow S'_{ix}S'_{iy} + S'_{iy}S'_{iz} + S'_{iz}S'_{ix} = 0, \quad i = 1, \ldots, N. \quad (16)$$

Therefore, at least two of three projected areas $S'_{ix}, S'_{iy}, S'_{iz}$ of a triangle T_i are 0, which means all triangles T_i $(i = 1, \ldots, N)$ of the given mesh M are parallel to one of these three planes YOZ, ZOX, XOY, and so every triangle is parallel or orthogonal to all other triangles, i.e. M is rectilinear. □

4 Measuring Rectilinearity for 3D Meshes

4.1 Basic Idea

Theorem 1 gives the basic idea for the rectilinearity measurement of 3D meshes. Theorem 1 together with $S(M) \le P(M,\alpha,\beta,\gamma)$ suggests that the maximum ratio function

$$Ratio(M) = \max_{\alpha,\beta,\gamma\in[0,2\pi]} \frac{S(M)}{P(M,\alpha,\beta,\gamma)} \quad (17)$$

can be used as a rectilinearity measure, which is invariant under similarity transformations, for the mesh M.

Since $S(M) \le P(M,\alpha,\beta,\gamma)$, it follows that $\frac{S(M)}{P(M,\alpha,\beta,\gamma)} \le 1$. However, the infimum for the set of values of $\frac{S(M)}{P(M,\alpha,\beta,\gamma)}$ is not zero. So, for our purpose, it is necessary to determine the maximal possible μ such that the function (17) belongs to the interval $[\mu, 1]$ for any mesh M. Theorem 2 shows that $\mu = \frac{2}{3}$ and there is no mesh satisfying

$$\max_{\alpha,\beta,\gamma\in[0,2\pi]} \frac{S(M)}{P(M,\alpha,\beta,\gamma)} = \frac{2}{3}. \tag{18}$$

Theorem 2

1) *The inequality*

$$\max_{\alpha,\beta,\gamma\in[0,2\pi]} \frac{S(M)}{P(M,\alpha,\beta,\gamma)} > \frac{2}{3} \tag{19}$$

holds for any 3D mesh M.

2) *For any $\varepsilon > 0$, there is a mesh M such that*

$$\max_{\alpha,\beta,\gamma\in[0,2\pi]} \frac{S(M)}{P(M,\alpha,\beta,\gamma)} < \frac{2}{3} + \varepsilon \tag{20}$$

or equivalently

$$\inf_{M\in\Pi} \left\{ \max_{\alpha,\beta,\gamma\in[0,2\pi]} \frac{S(M)}{P(M,\alpha,\beta,\gamma)} \right\} = \frac{2}{3}, \tag{21}$$

where Π denotes the set of all 3D meshes.

Proof For convenience, we use an alternative expression of $P(M,\alpha,\beta,\gamma)$ by

$$\begin{aligned} P(M,\alpha,\beta,\gamma) &= \sum_{i=1}^{N} P(T_i,\alpha,\beta,\gamma) \\ &= \sum_{i=1}^{N} P(T_i, \psi_i(\alpha,\beta,\gamma), \varphi_i(\alpha,\beta,\gamma)) \\ &= \sum_{i=1}^{N} \Big(|\cos(\psi_i(\alpha,\beta,\gamma))| S_i \\ &\quad + |\sin(\psi_i(\alpha,\beta,\gamma))| |\cos(\varphi_i(\alpha,\beta,\gamma))| S_i \\ &\quad + |\sin(\psi_i(\alpha,\beta,\gamma))| |\sin(\varphi_i(\alpha,\beta,\gamma))| S_i \Big) \end{aligned} \tag{22}$$

where $\psi_i(\alpha,\beta,\gamma)$ denotes the angle between the z axis and the normal of triangle T_i after the coordinate frame has rotated by the angles α,β,γ around its x,y,z axes, while the angle between the x axis and the perpendicular plane of T_i is represented by $\varphi_i(\alpha,\beta,\gamma)$ (see Fig. 4). The last equality of the above equation comes from the fact that the projected areas of a triangle on *YOZ*, *ZOX*, and *XOY* planes are directly proportional to the components of the normal on x, y, and z axis, respectively. For simplicity of notation, we will use ψ_i and φ_i instead of $\psi_i(\alpha,\beta,\gamma)$ and $\varphi_i(\alpha,\beta,\gamma)$ in the following part of this paper.

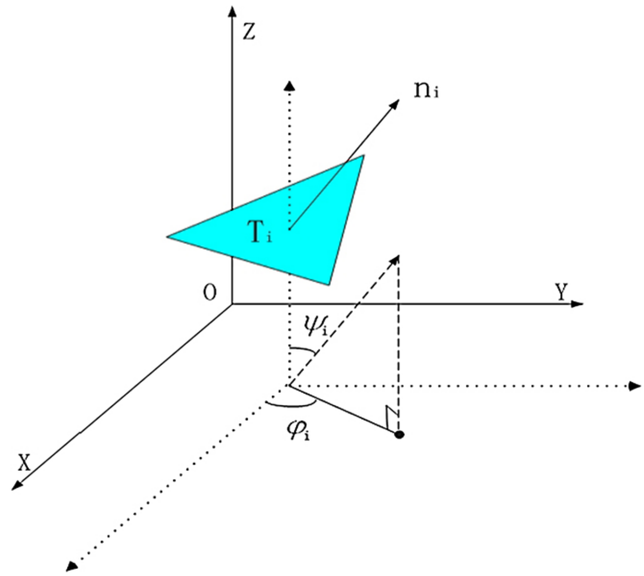

Fig. 4 Geometric relationships between a triangle and the coordinate system

We prove statement 1) by a contradiction. Let us assume the contrary, i.e., there exists a mesh M, which consists of N triangles, such that $\frac{S(M)}{P(M,\alpha,\beta,\gamma)} \le \frac{2}{3}$, or equivalently, $\frac{P(M,\alpha,\beta,\gamma)}{S(M)} \ge \frac{3}{2}$, for any $\alpha,\beta,\gamma \in [0,2\pi]$.

Since $\frac{P(M,\alpha,\beta,\gamma)}{S(M)}$ is a continuous nonconstant function defining on $\alpha,\beta,\gamma \in [0,2\pi]$, the equality $\frac{P(M,\alpha,\beta,\gamma)}{S(M)} = \frac{3}{2}$ cannot be always satisfied for all $\alpha,\beta,\gamma \in [0,2\pi]$. (see Appendix Lemma 1). So we have

$$\begin{aligned} &\iint_{\Sigma} \frac{P(M,\alpha,\beta,\gamma)}{S(M)} \cdot ds \\ &= \iint_{\Sigma} \frac{\sum_{i=1}^{N} P(T_i,\psi_i,\varphi_i)}{S(M)} \cdot ds \\ &= \sum_{i=1}^{N} \int_0^{2\pi}\int_0^{\pi} \frac{P(T_i,\psi_i,\varphi_i)}{S(M)} \cdot \sin\psi_i d\psi_i d\varphi_i \\ &= \int_0^{2\pi}\int_0^{\pi} \frac{\sum_{i=1}^{N} P(T_i,\psi,\varphi)}{S(M)} \cdot \sin\psi d\psi d\varphi \\ &> \int_0^{2\pi}\int_0^{\pi} \frac{3}{2} \cdot \sin\psi d\psi d\varphi = 6\pi \end{aligned} \tag{23}$$

where Σ denotes the surface of the unit sphere. Using the last inequality, we derive

$$6\pi < \int_0^{2\pi}\int_0^{\pi} \frac{\sum_{i=1}^{N} P(T_i,\psi,\varphi)}{S(M)} \cdot \sin\psi d\psi d\varphi$$

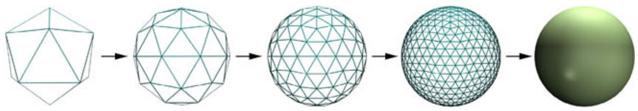

Fig. 5 n-triangle meshes which converge to a unit sphere when n tends to infinity

$$
\begin{aligned}
&= \frac{1}{S(M)} \sum_{i=1}^{N} \int_0^{2\pi}\int_0^{\pi} P(T_i,\psi,\varphi)\cdot \sin\psi d\psi d\varphi \\
&= \frac{1}{S(M)} \sum_{i=1}^{N} \int_0^{2\pi}\int_0^{\pi} \big(|\cos\psi| + |\sin\psi||\cos\varphi| \\
&\quad + |\sin\psi||\sin\varphi|\big) S_i \sin\psi d\psi d\varphi \\
&= \frac{\sum_{i=1}^{N} S_i \int_0^{\pi}(2\pi|\cos\psi|\sin\psi + 8|\sin\psi|^2)d\psi}{\sum_{i=1}^{N} S_i} \\
&= \frac{\sum_{i=1}^{N} S_i(2\pi + 4\int_0^{\pi}(2\sin^2\psi - 1 + 1)d\psi)}{\sum_{i=1}^{N} S_i} \\
&= \frac{\sum_{i=1}^{N}(6\pi)S_i}{\sum_{i=1}^{N} S_i} = 6\pi.
\end{aligned}
\tag{24}
$$

This yields the contradiction $6\pi < 6\pi$ which proves the statement 1).

It is enough to prove statement 2) if we can find a sequence of meshes $\ldots M_{99}, M_{100}, \ldots, M_n, \ldots$ (M_n denotes a mesh consisting of n triangles) such that

$$\lim_{n\to\infty}\left(\max_{\alpha,\beta,\gamma\in[0,2\pi]} \frac{S(M_n)}{P(M_n,\alpha,\beta,\gamma)}\right) = \frac{2}{3}. \tag{25}$$

Intuitively the sequence of n-triangle mesh M_n (Fig. 5 shows some examples of them) inscribed into the unit sphere satisfies the previous equality. According to the properties of the sphere, we get the surface area $\lim_{n\to\infty} S(M_n) = 4\pi$ and the sum of three projected areas $\lim_{n\to\infty} P(M_n,\alpha,\beta,\gamma) = 2\pi + 2\pi + 2\pi = 6\pi$, both of them hold independently on the choice of α,β,γ. Therefore, we have

$$
\begin{aligned}
\lim_{n\to\infty}\left(\max_{\alpha,\beta,\gamma\in[0,2\pi]} \frac{S(M_n)}{P(M_n,\alpha,\beta,\gamma)}\right) &= \lim_{n\to\infty} \frac{S(M_n)}{P(M_n,\alpha,\beta,\gamma)} \\
&= \frac{2}{3}
\end{aligned}
\tag{26}
$$

which proves statement 2). □

4.2 A Rectilinearity Measure

Motivated by the properties of the maximum ratio function (17) we define a rectilinearity measure for 3D meshes.

Definition 2 For an arbitrary 3D mesh M we define its rectilinearity $R(M)$ as

$$R(M) = 3 \times \left(\max_{\alpha,\beta,\gamma\in[0,2\pi]} \frac{S(M)}{P(M,\alpha,\beta,\gamma)} - \frac{2}{3}\right). \tag{27}$$

According to the definitions and theorems introduced above, we obtain the following theorem which summarizes the properties of the 3D mesh rectilinearity measure proposed here.

Theorem 3 *For any 3D mesh M, we have*:

1. *$R(M)$ is well defined and $R(M) \in (0,1]$*;
2. *$R(M) = 1$ if and only if M is rectilinear*;
3. $\inf_{M\in\Pi}(R(M)) = 0$;
4. *$R(M)$ is invariant under similarity transformations.*

4.3 Computation

Unlike the computation of the accurate rectilinearity for 2D polygons (Žunić and Rosin 2003), because of the complexity of $P(M,\alpha,\beta,\gamma)$ it is difficult to calculate the exact value of rectilinearity for 3D meshes. From the introduction of preceding sections, we can see that the computation of rectilinearity is actually a nonlinear optimization problem which can be efficiently solved by intelligent computing methods. In this paper we choose the *genetic algorithm* (GA) which is an optimization technique based on natural evolution (Holland 1992).

First, we define a population including N_g individuals. Each individual consists of a value of fitness and three different chromosomes which are presented by binary codes. The fitness of an individual is defined as $fit(\alpha,\beta,\gamma) = \frac{S(M)}{P(M,\alpha,\beta,\gamma)}$ and rotating angles α,β,γ are encoded in the three chromosomes. The stopping criterion here is the number of evolution generations N_{gen}.

Then, iterating the genetic algorithm process including encoding, evaluation, crossover, mutation and decoding for N_{gen} generations, we get an individual with the group's greatest fitness which can be used as the approximate value of

$$\max_{\alpha,\beta,\gamma\in[0,2\pi]} \frac{S(M)}{P(M,\alpha,\beta,\gamma)}.$$

Finally, we calculate the rectilinearity of the mesh by (27). Figure 6 demonstrates the convergence curve of some calculation examples using GA, and we observe that usually the stable value of rectilinearity measure can be found after approximately 150 evolution generations.

Note that:

1. Unless otherwise specified, the parameters of the genetic algorithm in this paper are chosen as follows. The number of individuals $N_g = 50$ and evolution generations

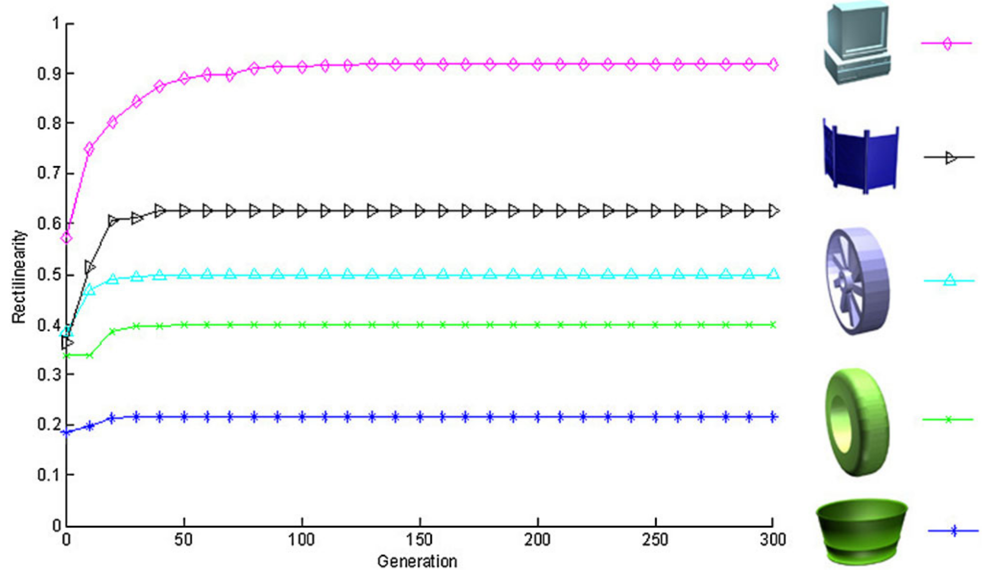

Fig. 6 The value of rectilinearity measure versus the evolution generation of the GA

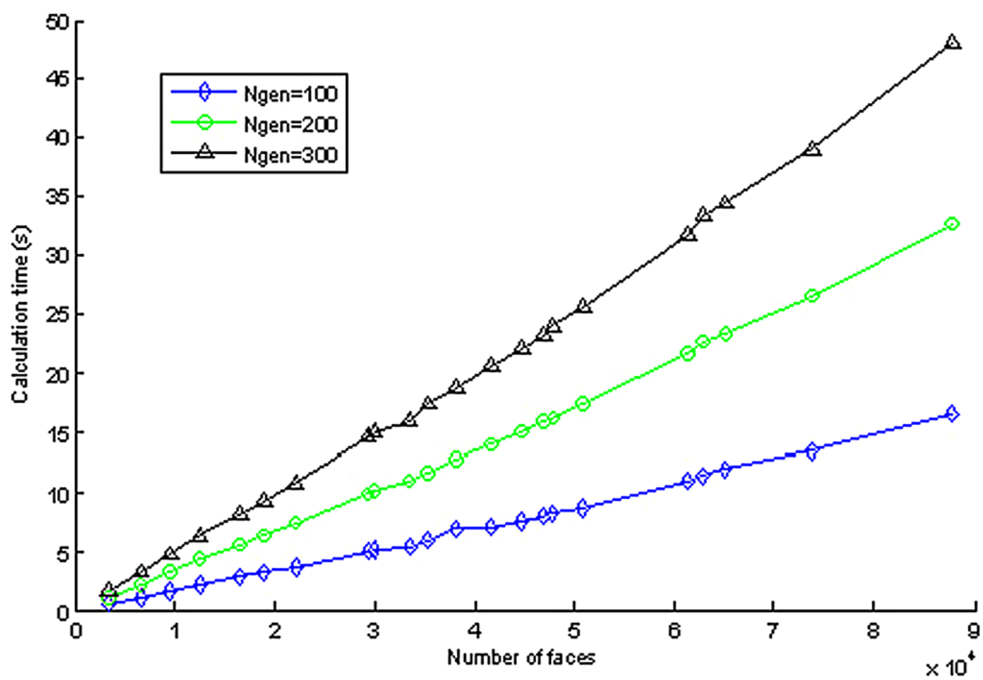

Fig. 7 The number of faces versus computation time, for three different numbers of evolution generations

$N_{gen} = 200$. The length of each chromosome's binary code $L_c = 20$. The probability of crossover $p_c = 0.800$ and mutation $p_m = 0.005$.

2. Since our rectilinearity measure is invariant to flipping of the coordinate axes, that is to say,

$$\max_{\alpha,\beta,\gamma\in[0,2\pi]} \frac{S(M)}{P(M,\alpha,\beta,\gamma)} = \max_{\alpha,\beta,\gamma\in[0,\pi/2]} \frac{S(M)}{P(M,\alpha,\beta,\gamma)}, \tag{28}$$

the search range of a rotation angle (α, β, or γ) can be therefore narrowed from $[0, 2\pi]$ to $[0, \pi/2]$.

3. As our results demonstrate, the GA provides an adequate solution to computing the rectilinearity measure. However, we note that it is neither the only, nor necessarily the best, numerical optimization method to compute the rectilinearity measure. For instance, other intelligent computing algorithms such as Artificial Neural Network, Simulated Annealing Algorithm, and Ant Colony Optimization etc. all have their particular advantages over GA in some cases. Even exhaustive searching methods are more suitable for some applications requiring only low accuracy data and these results can also be used as initial

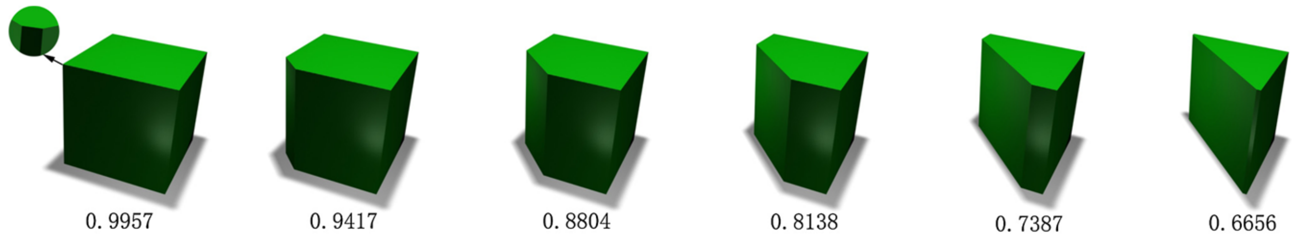

Fig. 8 3D meshes have identical inner angles but differ in rectilinearity. Underneath are the corresponding rectilinearity values

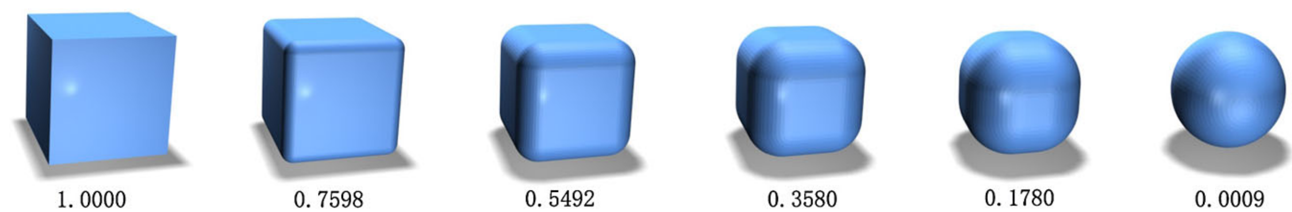

Fig. 9 3D meshes derived from a cube by filleting all of its edges with increasing radius. Underneath are the corresponding rectilinearity values

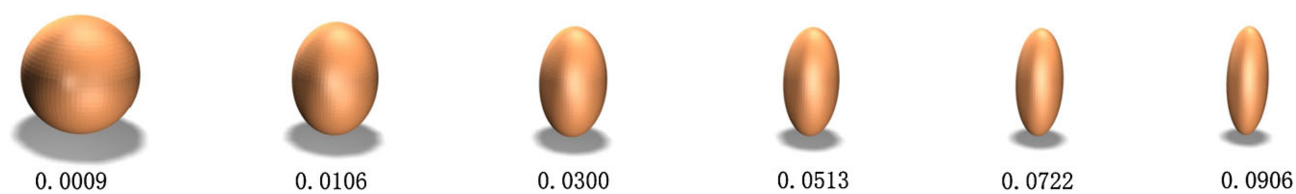

Fig. 10 Ellipsoids that are progressively elongated. Underneath are the corresponding rectilinearity values

searching points in the search space of intelligent computing algorithms to further improve performance.

5 Experimental Results

Since the rectilinearity measure proposed in this paper is calculated based on the area of triangles, together with the definitions and theorems described in the previous sections we can summarize the primary advantages (already described in Sect. 1) of this measurement. While these four properties indicate that the rectilinearity measurement may be well suited for 3D shape analysis tasks in theory, it is necessary to demonstrate that these desirable properties are also satisfied in practice. As the third and fourth properties are obvious, in this section we only illustrate experimental results to investigate the following questions:

1. How well does this measure correspond with the intuitive perception of rectilinear 3D shapes?
2. How robust is this measure with respect to geometric noise and small errors or changes in topology?

We implemented the calculation of rectilinearity described in Sect. 4.3 in Visual C. The experiments were run on a Windows XP Laptop with a 2.0 GHz Intel Core 2 Duo CPU, 1.0 GB DDR2 memory and an NVIDIA Quadro NVS 140M graphics card. After the parameters of the GA have been chosen, the total computing time is proportional to the number of faces. When the number of generations is selected as $N_{gen} = 200$, the average time to compute the rectilinearity of each model in the PSB database is 2.3 seconds. The relationship between computing time and the number of faces of some individual meshes is demonstrated in Fig. 7. We can see that usually the calculation can be finished within seconds and the computing complexity is $O(N)$ where N denotes the number of faces.

5.1 Rank of the Rectilinearity

In order to give the reader an indication of the perceptual quality of our rectilinearity measurement, we demonstrate it by three particular cases. First, it is applied to some meshes which are obtained from a cube by cutting off, in a given direction, parts of increasing size. Second, we calculate the rectilinearity values of a set of 3D meshes which are derived from a cube by filleting all of its edges with increasing radius. The third example is the rectilinearity calculation of several ellipsoids that are progressively elongated. The computed rectilinearity values are shown in Figs. 8, 9, and 10, respectively. All these three examples show that our rectilinearity measure is well behaved.

Fig. 11 Shapes ranked by rectilinearity. Underneath are the corresponding rectilinearity values

The rectilinearity measure is now applied to a wide range of 3D meshes which are then ranked in order of decreasing rectilinearity (Fig. 11). Comparison against our intuitive notion shows that a similar ordering has been generated.

5.2 Robustness

It is often desirable that the shape descriptor is insensitive to noise and small extra features, and robust against arbitrary small topological degeneracies. To demonstrate the robustness of the calculation of this measure, we first add small amounts of noises to a model or change its important topology structure, and then compute the rectilinearity for them. Results (see Fig. 12) show that our rectilinearity measurement of 3D meshes is robust to small noise or changes in topology, this is mainly because the measure corresponds to the surface area and projected areas which are hardly affected by small changes of a mesh.

Fig. 12 Robustness against topology changes and small noise. An original chair is displayed in the *leftmost*, the topology of some important structures of the middle one are varied, while small amounts of noises are introduced to form the *rightmost* model. Underneath are the corresponding rectilinearity values

Our next example shows more precisely how the rectilinearity measurement can tolerate deletion of random faces. As shown in Fig. 13, we first randomly eliminate different percentages of faces for a synthetic model (teapot in the first row) and a real scanned model (dragon in the second row) (Stanford 3D Scanning Repository 2008), and then measure their rectilinearity. From the curves in the rightmost column,

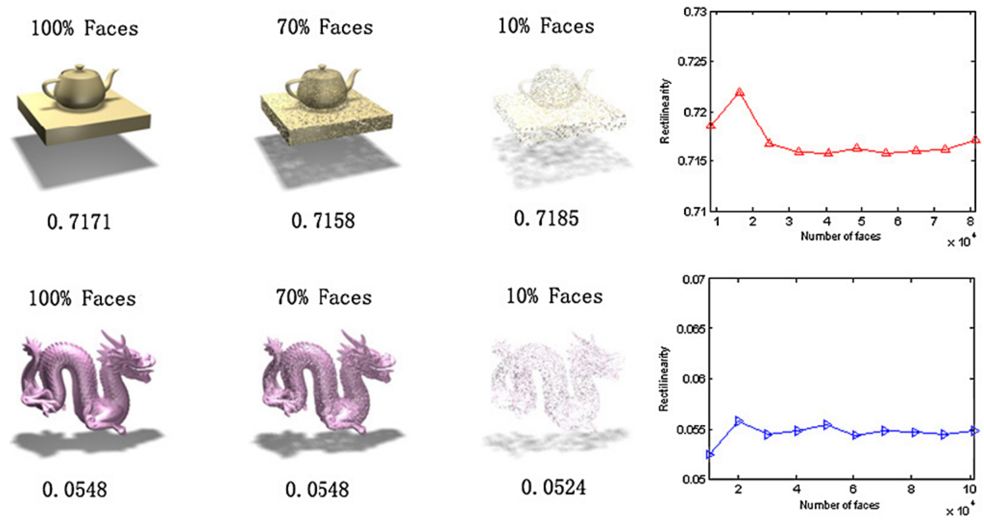

Fig. 13 Robustness against random face deletion, Underneath are the corresponding rectilinearity values

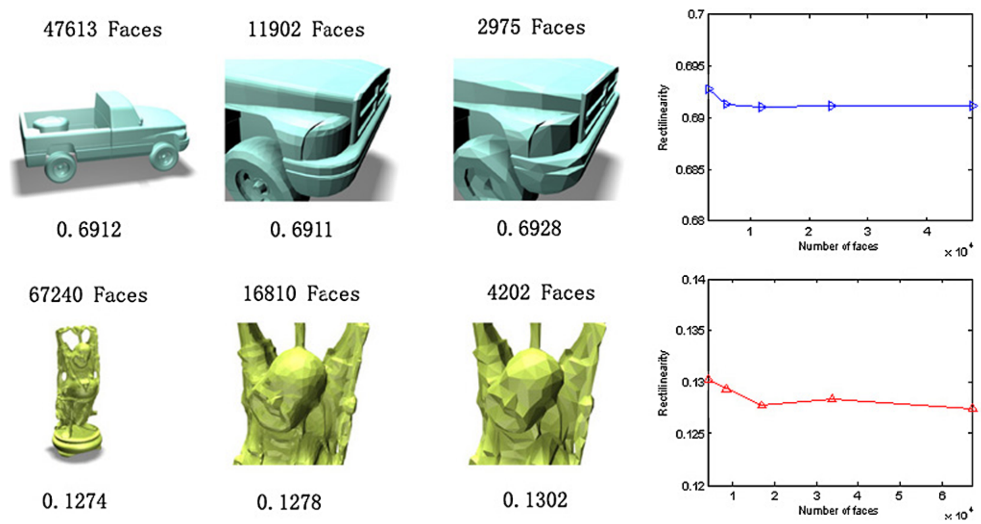

Fig. 14 Robustness against mesh simplification. For representation sake, the simplified meshes in the middle are zoomed in to focus on local parts. Underneath are the corresponding rectilinearity values

we can see that the value of rectilinearity remains stable with respect to the large amounts of information missing. Even if 90% faces of the mesh are discarded, the rectilinearity value varies by less than 0.005.

The last example demonstrates how robust our method is with respect to mesh simplification. In the first column of Fig. 14, original meshes including a car (synthetic model) and the famous Happy Buddha (Stanford 3D Scanning Repository 2008) (scanned model) are displayed. Then the examples of simplified meshes, applying the Quadric Edge Collapse Decimation provided by MeshLab1.1.0 (2008), are displayed in the second and third columns. The rightmost graphs show the relationship between the rectilinearity measure and the number of faces. We observe that simplification of the meshes has little effect on the calculated rectilinearity values for both the synthetic models and the real scanned objects.

5.3 Limitations

There are two major limitations in our rectilinearity measuring method. In this subsection, we will discuss them briefly and point out some potential solutions.

First, as described in Sect. 4.3, the rectilinearity measure here is calculated by GA that may converge to a local maximum and the computation time is still expensive compared to the 2D approach proposed by Žunić and Rosin (2003). Therefore, developing a more efficient numerical computation algorithm might be the best choice if we finally find out that it is impossible or more costly to calculate the optimal value of rectilinearity analytically.

Second, our rectilinearity measure is sensitive to some kinds of changes in a mesh. For example, a mesh could appear almost exactly as a pyramid as the number of its components tends to infinity (shown in Fig. 15), while the rectilinearity value of the pyramid-like mesh remains 1. In contrast, the original pyramid which consists of five vertices would

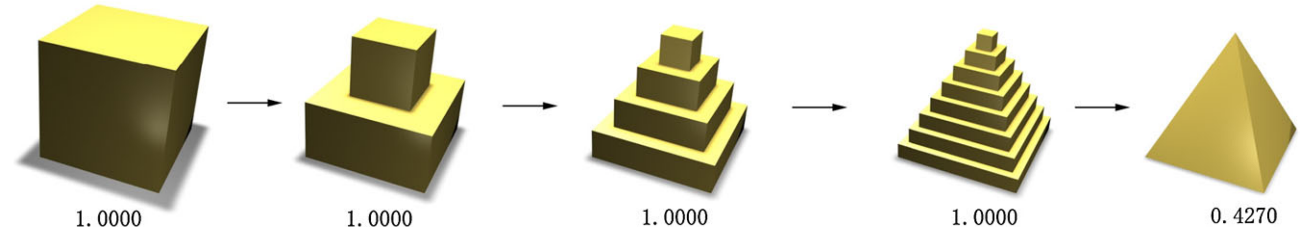

Fig. 15 Meshes whose appearances tend to a pyramid, but not the rectilinearity values measured by our method. Underneath are the corresponding rectilinearity values

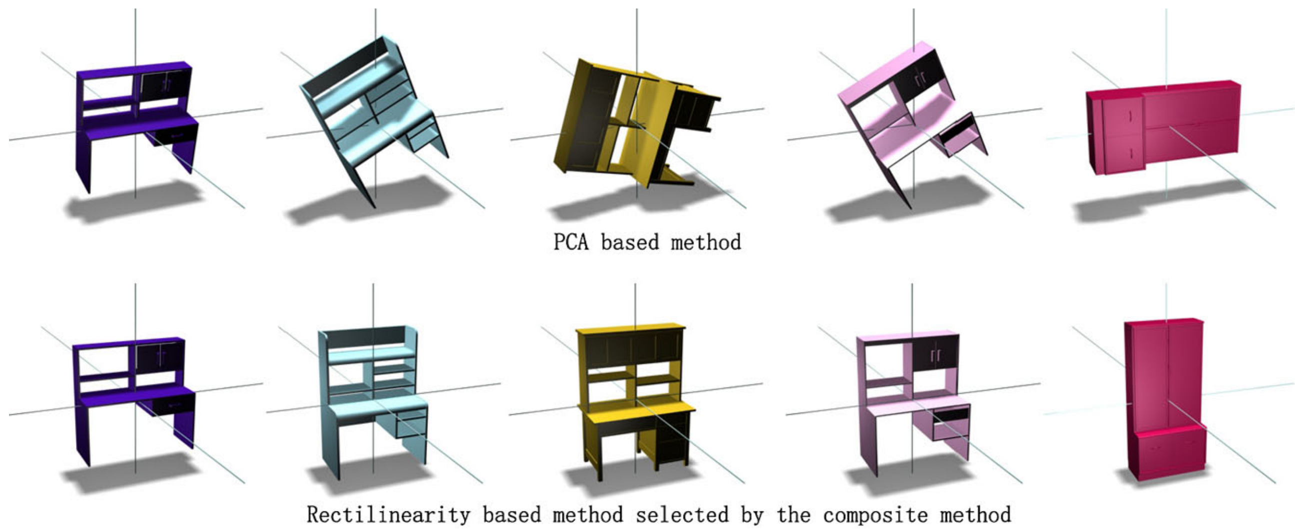

Fig. 16 Cabinets normalized by two methods. The results of the PCA based method are demonstrated in the *first row*, while final results, the same as the rectilinearity based method, are displayed in the *second row*

appear perceptually identical but has a very different rectilinearity value (0.4270). Such effects can often occur with scanned objects and voxel models. However, a simple solution would be to first smooth the mesh sufficiently to eliminate the effects of quantization errors. We can also analyse an object at multiple scales obtained by smoothing and get a set of rectilinearity values which are suitable for different applications.

6 Pose Normalization Application

In this section, we describe how to normalize a mesh by its rectilinearity and how to combine this new method with PCA to generate a reliable result which corresponds well with human intuition.

We find out that the calculation of rectilinearity can be used for pose estimation of 3D meshes. The basic idea is that the value of α, β, γ maximizing $\frac{S(M)}{P(M,\alpha,\beta,\gamma)}$ of a mesh M specifies a standard pose for this object. Figure 16 demonstrates several normalized cabinets applying the rectilinearity based and the PCA based normalization methods, respectively. Referring to the figure, we can see that, in this case, PCA-alignment not only corresponds poorly to our intuitive perception but also results in quite dissimilar orientations for different models within the same class. On the contrary, the rectilinearity based method provides much better results that exactly coincide with our common sense. In fact, through the experiments conducted on the PSB database, we observe that, usually, from intuitive human perception, the rectilinearity based method performs better than the PCA based method, especially when processing artificial objects such as cabinets, tables, chairs, houses, etc. (see some examples in Fig. 17).

However, the rectilinearity based method does not guarantee good performance for all shapes. For instance, when the properties of an object are dominated by its principal axes (i.e. elongated objects) or symmetry, better alignment results can be obtained by applying the PCA based method rather than rectilinearity. Examples are illustrated in Fig. 18.

Consequently, we suggest using a combination of these two methods. A selection criterion will be defined such that the correct alignments would be automatically chosen from the two methods. The steps of our composite pose normalization (CPN) method are listed as follows.

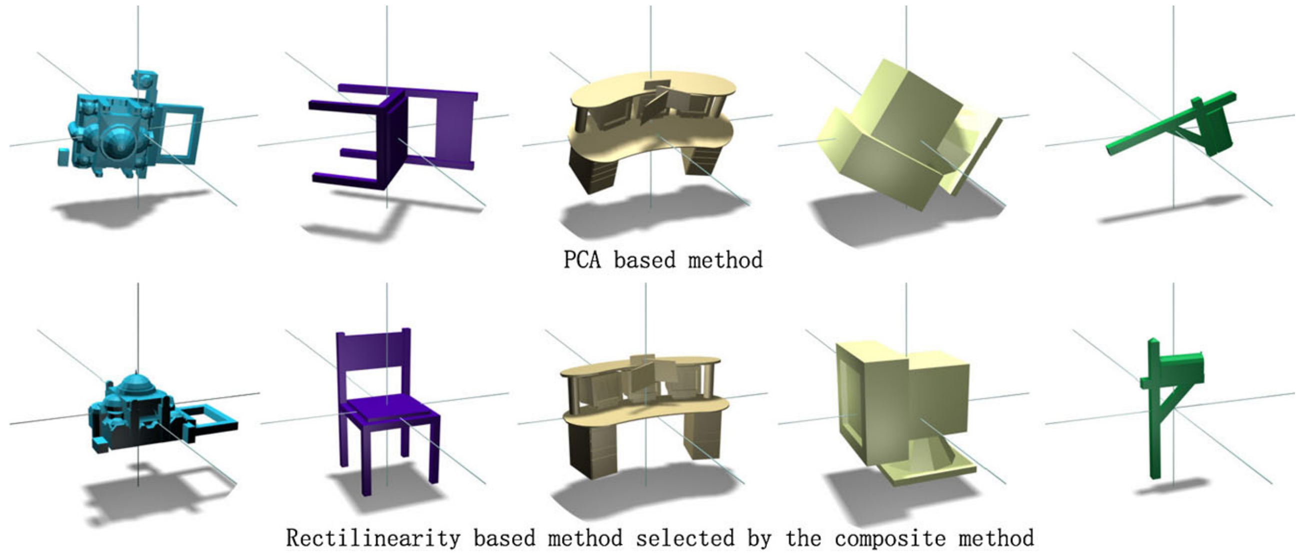

Fig. 17 Several models normalized by two methods. The results of the PCA based method are demonstrated in the *first row*, while final results, chosen to apply the rectilinearity based method, are displayed in the *second row*

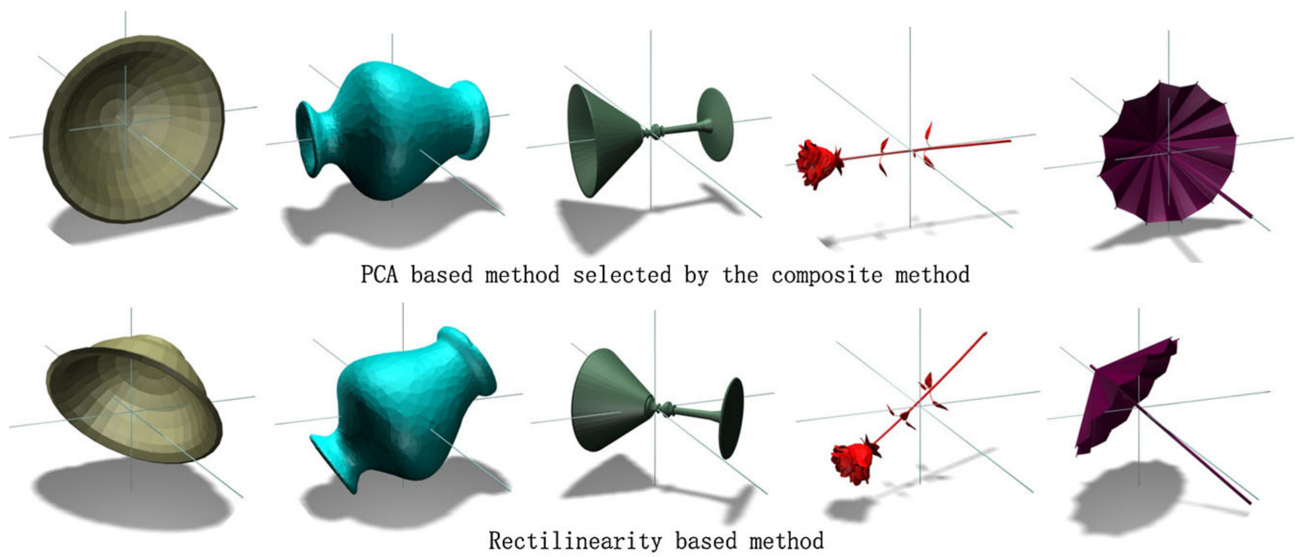

Fig. 18 Several models normalized by two methods. Final results, chosen to adopt the PCA based method, are displayed in the *first row*, while the results of the rectilinearity based method are demonstrated in the *second row*

1. *Translation and scaling*. For a given 3D mesh, translate the center of its mass to the origin and then scale the maximum polar distance of the points on its surface to one.
2. *Rotation by two methods*. Apply PCA and the rectilinearity based method, respectively, to rotate the original model to the canonical coordinate frame and then store these two normalized meshes in memory;
3. *Selection*. Calculate the number of valid pixels of three silhouettes, projected on the planes *YOZ*, *ZOX*, *XOY*, for the two normalized meshes generated in the previous step. And then select the model which yields the smaller value, as the final normalization result.

Here we make the selection just according to the summed area of the projected silhouettes which we have observed to be effective for almost all models. Take Fig. 19 for an example, it is obvious that the well normalized cabinet yields a smaller value of the summed area of three projected silhouettes than the one incorrectly normalized. Using our composite pose normalization method, almost all models in the PSB database can be successfully normalized. By manual inspection (an experiment with 39 participants has been carefully conducted), we find that more than 98.2% models in the database have been correctly normalized. The experiment was carried out as follows. First, all models normalized by the new method were provided to the participants with a user-friendly interface. Afterwards, they were asked to judge whether the given models had been correctly normalized according to their intuitive perception. Finally, the average percentage of correctly normalized models on the PSB can be calculated. Moreover, we also validate the effectiveness through a more objective way. Using the GSMD_66 descriptor to evaluate retrieval performance (the same signature and evaluation method as described in Sect. 7.1), the Discounted

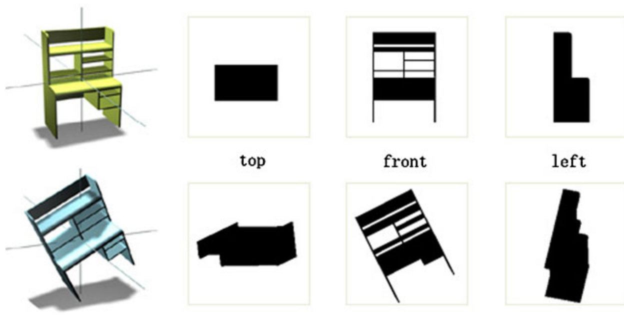

Fig. 19 Demonstration of our selection criterion. A correctly normalized model and its three projected silhouettes in top, front, and left views are shown in the *first row*, while the *second row* displays an example of incorrectly aligned model which yields a greater value of summed area of these three silhouettes

Cumulative Gain (well-known as the most stable retrieval measure (Shilane and Funkhouser 2007)) is 0.660 for the database normalized using PCA, while the value is improved to 0.664 by our method. Several successful normalization examples are displayed in the second row of Figs. 16 and 17, where the rectilinearity based method is chosen. In contrast, PCA is selected to correctly align the models in the first row of Fig. 18.

Nevertheless, there are still some models (about 1.8%) which can not be perfectly normalized by our method. Inspecting the models on which it failed, we could classify them into following two categories and suggest several possible reasons for failure. The first class consists of the models which can be correctly normalized by one of these two original methods separately but not the composite approach. About 80% of the failed cases belong to this category. For instance, the hand gun and the flowerpot, shown in the first column and the second column of Fig. 20 are correctly normalized by the rectilinearity based method, but finally the composite method choose PCA to incorrectly align the meshes. On the other hand, the hot-air balloon in the middle can be successfully normalized by PCA instead of the rectilinearity method. In this case, the summed areas of three projected silhouettes of the models normalized by two methods are

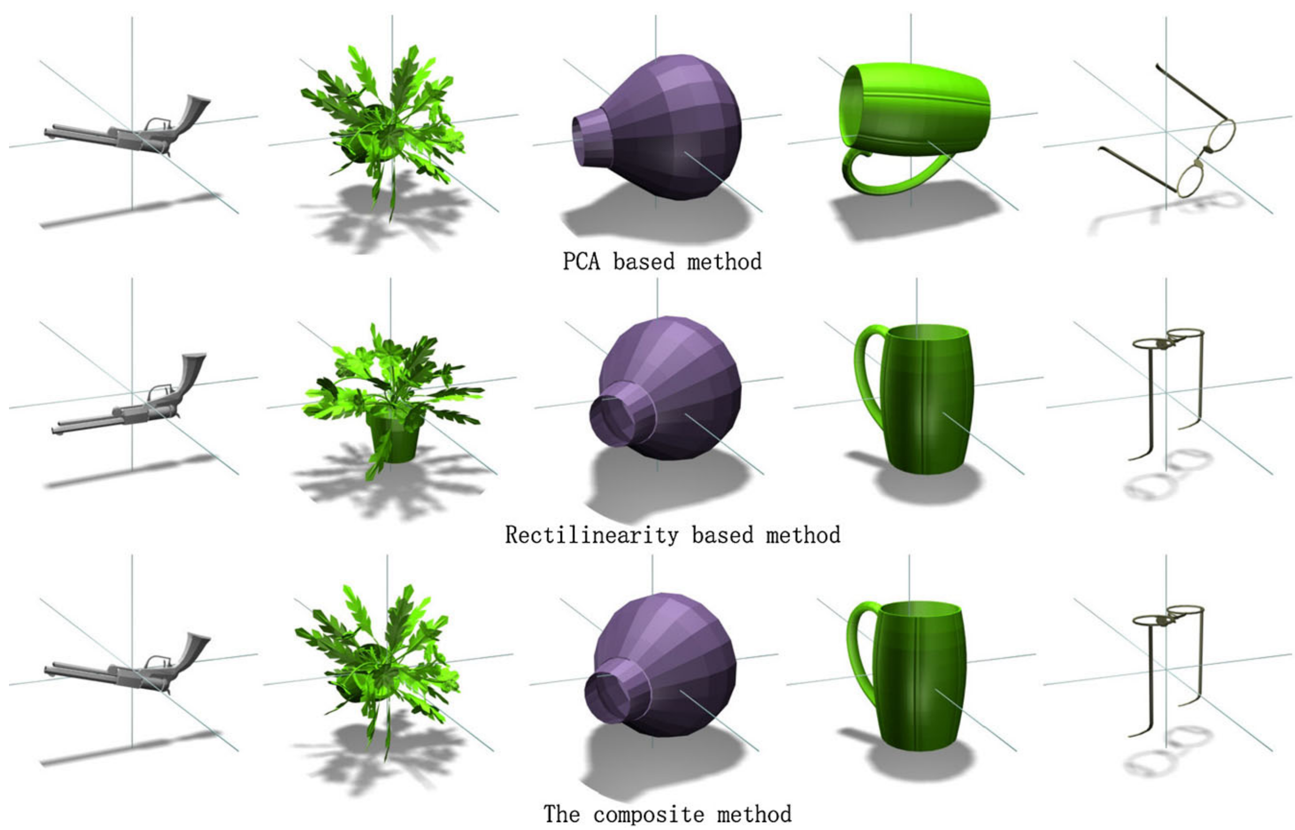

Fig. 20 Examples of incorrect normalization by our method. In the *first row*, models are normalized by the PCA based method, while the rectilinearity based method is adopted to the *second-row* objects. Final normalized poses are illustrated in the *third row*

almost the same. We speculate that the area of silhouettes alone might not be sufficient for basing our decisions. The second class consists of the models on which both PCA and the rectilinearity based method failed. Just about 20% of the failed cases belong to this category. Examples can be found in the last two columns of Fig. 20, from which we can see that the cup and the eyeglasses have been incorrectly normalized by both two methods, thus the composite method inevitably results in unsatisfactory alignments.

The potential solution may be either introducing more high-level information of the depth buffers to the selection criterion or developing a learning mechanism to integrate as many significant properties (e.g. rectilinearity, principal axes, and symmetry et al.) as possible.

7 3D Shape Retrieval Application

In this section, we describe how to design a discriminative view based shape matching mechanism for the models normalized by the composite method and demonstrate that our rectilinearity measure can be used as an important element of a composite descriptor to efficiently improve the retrieval performance.

7.1 Searching for Normalized Models

After pose normalization, 3D meshes have been well aligned to a canonical coordinate frame. However, only the positions of three axes are fixed for the mesh normalized by our composite method, namely, the direction of each axis is still undecided and the x-axis, y-axis, z-axis of the canonical coordinate system can be located in all three axes. That means 24 different orientations are still plausible for the aligned models, or rather, 24 matching operations should be carried out when comparing two normalized objects. Shih et al. (2007) described a similar matching approach, but they just used the relative position between every two depth buffers and did not take the right-hand rule into account, so that their method resulted in two times more matching operations than ours. Moreover, they just investigated the situation involving six images.

Without pose normalization, due to the $O(N^3)$ complexity of shape matching (Chen et al. 2003), it is almost impractical to utilize a large number of images to extract view-based descriptors for further comparison. Applying an offline pose normalization preprocessing, the method utilized in this section reduces the matching complexity to $O(N)$. Furthermore, it can be executed at different levels according to the desired retrieval speed and discrimination. Although the idea of matching views for normalized objects is considered as common sense and it has been used by many researchers (Chen et al. 2003; Shilane et al. 2004; Chaouch and Blondet 2007; Shih et al. 2007), so far no one has systematically described how to efficiently match huge (even infinite) numbers of views captured from the models with incomplete pose alignment.

For the sake of convenience, we denote $x+$, $x-$, $y+$, $y-$, $z+$, and $z-$ axis as 0, 1, 2, 3, 4, and 5 respectively. When comparing two models, one of them will be placed in the original orientation denoted as a permutation $p_0 = \{p_0(k)|k = 0, 1, 2, 3, 4, 5\}$ while the other one may appear in 24 different poses denoted as permutations $p_i = \{p_i(k)|k = 0, 1, 2, 3, 4, 5\}$, $0 \le i \le 23$. Table 1 lists these 24 permutations from which all possible matching pairs ((p_0, p_i), $0 \le i \le 23$) between two models can be obtained. More specifically, we can capture six silhouettes or depth buffers from the vertices of a unit regular octahedron and then extract 2D shape descriptors for these images to construct a view-based 3D feature vector. The vertices in the corresponding axes are also denoted as 0, 1, 2, 3, 4, and 5, respectively. Then we compare all 24 matching pairs for two models and the minimum distance is chosen as their dissimilarity.

The geodesic spheres generated from the unit regular octahedron are suitable for multi-view based feature extraction and shape matching. Examples of this kind of geodesic spheres are displayed in Fig. 21. For more details, we refer

Table 1 Twenty-four permutations for shape matching between a query model and a matching model, both of them have been normalized before matching

k	0	1	2	3	4	5	k	0	1	2	3	4	5	k	0	1	2	3	4	5
$p_0(k)$	0	1	2	3	4	5	$p_8(k)$	4	5	2	3	1	0	$p_{16}(k)$	2	3	1	0	4	5
$p_1(k)$	0	1	4	5	3	2	$p_9(k)$	4	5	1	0	3	2	$p_{17}(k)$	2	3	4	5	0	1
$p_2(k)$	0	1	3	2	5	4	$p_{10}(k)$	4	5	3	2	0	1	$p_{18}(k)$	2	3	0	1	5	4
$p_3(k)$	0	1	5	4	2	3	$p_{11}(k)$	4	5	0	1	2	3	$p_{19}(k)$	2	3	5	4	1	0
$p_4(k)$	1	0	3	2	4	5	$p_{12}(k)$	5	4	2	3	0	1	$p_{20}(k)$	3	2	0	1	4	5
$p_5(k)$	1	0	4	5	2	3	$p_{13}(k)$	5	4	0	1	3	2	$p_{21}(k)$	3	2	4	5	1	0
$p_6(k)$	1	0	2	3	5	4	$p_{14}(k)$	5	4	3	2	1	0	$p_{22}(k)$	3	2	1	0	5	4
$p_7(k)$	1	0	5	4	3	2	$p_{15}(k)$	5	4	1	0	2	3	$p_{23}(k)$	3	2	5	4	0	1

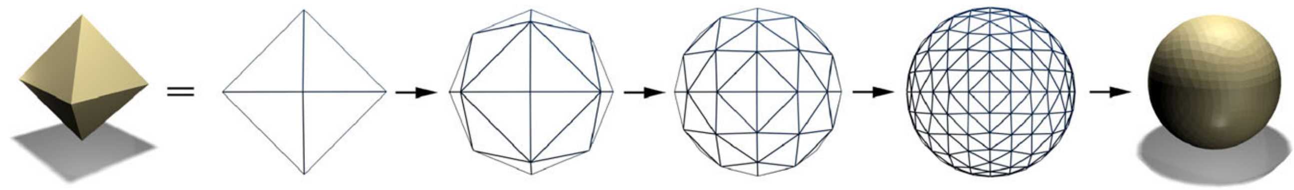

Fig. 21 Geodesic spheres generated from a regular octahedron

the reader to paper (Laga et al. 2006). These kinds of geodesic spheres can be used for our multi-view based shape retrieval mechanism, mainly because of following three reasons. First, the vertices are distributed evenly in all directions. While unless low discrepancy sequences are used, random sampling will produce an uneven distribution of viewpoints. Second, these geodesic spheres enable different level resolutions in a natural manner. The coarsest (level-0) one is obtained using a unit regular octahedron with 6 vertices and 8 faces. Higher levels can be generated by recursive subdivisions. Then we can capture different number of silhouettes or depth buffers from the vertices of the geodesic spheres and then extract 2D shape descriptors for these images to construct different level 3D feature vectors. Third, since all these spheres derive from an octahedron, given the position of six vertices for the original octahedron, other vertices can be specified automatically. Moreover, all vertices are symmetrically distributed with respect to the coordinate frame axes. That means, when comparing two models, only 24 matching pairs need to be considered for the feature vector in an arbitrary level.

Generally speaking, our method performs in three steps:

1. *Initialization*. Recursively subdividing the original unit octahedron n_d times, we get a geodesic sphere with the required resolution and the coordinates of its vertices should be recorded to a table (called the *vertex table*) consecutively according to the time they emerge. The number of its vertices is calculated as follows,

$$N_v(n_d) = N_v(n_d - 1) + N_e(n_d - 1), \tag{29}$$

$$N_e(n_d) = (N_f(n_d) \times 3)/2, \tag{30}$$

$$N_f(n_d) = 4 \times N_f(n_d - 1), \tag{31}$$

and

$$N_v(0) = 6, N_f(0) = 8, \tag{32}$$

where $N_e(n_d)$ and $N_f(n_d)$ stand for the number of edges and faces, respectively. During the process of subdivision, a table (named the *edge table*) which records the relationship between old and new vertices is also obtained. Note that we only need to process this step once.
2. *Feature extraction*. The vertices are selected as a model's viewpoints, from which $N_v(n_d)$ silhouettes or depth buffers are then captured. Next, we extract several 2D descriptors for each image before arranging them in a vector in the order that coincides with the vertex table. We denote the feature vector as $FV_i = \{FV_i(k)|0 \le k < N_v(n_d)\}$, where $FV_i(k)$ is the signature of image k.
3. *Shape matching*. As mentioned above, when comparing two models represented by level-0 descriptors, we will calculate the minimum distance among 24 matching pairs $((p_0, p_i),\ 0 \le i \le 23)$ which can be derived using the permutations listed in Table 1. If higher-level shape descriptors are applied, we should use the edge table and p_i, $0 \le i \le 23$ to build new permutations $p'_i = \{p'_i(k)|0 \le k < N_v\}$, $0 \le i \le 23$ describing all possible matching pairs (p'_0, p'_i), $0 \le i \le 23$ for two models represented by N_v views. Finally, using the L_1 norm, the dissimilarity between the query model q and the source model s is defined as,

$$Dis_{q,s} = \min_{0\le i\le 23} \sum_{k=0}^{N_v-1} \|FV_q(p'_0(k)) - FV_s(p'_i(k))\|. \tag{33}$$

In order to demonstrate the effectiveness and the efficiency of our method, we extract 2D shape descriptors, the same as the well-known LFD proposed by Chen et al. (2003), to describe the silhouettes captured on the N_v-vertex geodesic spheres. More specifically, we use the feature vector with 47 members including 35 Zernike moments, 10 Fourier coefficients, eccentricity and compactness to describe a silhouette and then the vector is normalized to its unit L_1 norm. For feature extraction, we use the source code from their web site without modification. However, for shape matching, we develop our own platform to compare different methods under the same condition without any extra optimizations, because the original LFD method applies a storage optimization and a hierarchical matching technology whose impacts on the retrieval performance are difficult to evaluate.

Unless otherwise specified, in this paper we use the PSB (Shilane et al. 2004) test set with base classification to evaluate the 3D shape retrieval performance that is quantified by the following evaluation measures:

- Nearest neighbor (1-NN): The percentage of the closest matches that belong to the same class as the query.

Table 2 Retrieval performance of our GSMD signatures and LFD descriptor

Methods	Length	Compare time (s)	1-NN	1-tier	2-tier	DCG
GSMD_258	258×47	0.000609	65.8%	40.6%	50.9%	65.9%
GSMD_66	66×47	0.000154	67.9%	40.8%	51.3%	66.4%
GSMD_18	18×47	0.000041	65.8%	40.1%	50.3%	65.6%
GSMD_6	6×47	0.000013	62.0%	35.2%	45.8%	62.2%
LFD	100×47	0.011642	65.0%	38.3%	48.7%	64.4%

- First-tier (1-Tier) and Second-tier (2-Tier): The percentage of models in the query's class that appear within the top K matches, where K depends on the size of the query's class. Specifically, for a class with $|C|$ members, $K = |C| - 1$ for the first tier, and $K = 2(|C| - 1)$ for the second tier.
- Discounted Cumulative Gain (DCG): A statistic that weights correct results near the front of the list more than correct results later in the ranked list under the assumption that a user is less likely to consider elements near the end of the list. For details, we refer the reader to the paper (Shilane et al. 2004).

Here, we test four Geodesic Sphere based Multi-view Descriptors(GSMD), denoted as GSMD_6, GSMD_18, GSMD_66, and GSMD_258, with respect to different resolution geodesic spheres having 6, 18, 66, and 258 vertices, respectively. Table 2 shows the storage requirements, comparison times, and retrieval statistics for LFD and our descriptors. Since the number of view pairs need to be compared is $(60 \times N_{rot}^2 \times 2 \times N_{cam})$ for LFD method, which uses $N_{cam} = 10$ viewpoints with $N_{rot} = 10$ random rotations, and $(24 \times N_{cam})$ for our GSMD method corresponding to a geodesic sphere with N_{cam} vertices, we can see that the retrieval speed as well as the discrimination has been significantly improved by our method. For example, GSMD_66 requires 44% less storage and performs about 70 times quicker, but provides better classification than LFD. We also observe that higher-level descriptor does not guarantee better discrimination. This is mainly because comparing too many details is not suitable for coarse shape retrieval tasks such as the base classification of PSB. It is reasonable to infer that high resolution GSMD descriptors could perform well if precise discrimination is necessary.

The above experiment demonstrates the advantage of the retrieval method with pose normalization against the method without pose normalization. In order to show the effect of our pose alignment method, using the GSMD_66 descriptor, we also compare the retrieval performance under different pose normalization and shape matching methods which are denoted as follows:

Table 3 Retrieval performance of the GSMD_66 descriptors with different pose normalization and shape matching methods

	1-NN	1-Tier	2-Tier	DCG
GSMD_rect24	67.9%	40.8%	51.3%	66.4%
GSMD_pca24	65.9%	40.3%	50.9%	66.0%
GSMD_pca4	65.7%	39.0%	49.5%	65.1%
GSMD_pca	65.0%	38.2%	48.5%	64.4%

– GSMD_rect24: GSMD_66 descriptor with our composite pose normalization algorithm and our original GSMD shape matching method.
– GSMD_pca24: GSMD_66 descriptor with PCA pose normalization algorithm and our original GSMD shape matching method.
– GSMD_pca4: GSMD_66 descriptor with PCA pose normalization algorithm using eigenvalues to resolve for axis switch. During shape matching, only 4 matching pairs (p'_0, p'_0), (p'_0, p'_2), (p'_0, p'_4), (p'_0, p'_6) are compared.
– GSMD_pca: GSMD_66 descriptor with PCA pose normalization algorithm using eigenvalues and area distribution (i.e. the positive direction of an axis points to the half part with larger surface area) (Shilane et al. 2004) to resolve for axis switch and flip, respectively. During shape matching, only one matching pair (p'_0, p'_0) is compared.

As we can see from Table 3, the approach using the composite pose normalization algorithm and our shape matching method obtain the best retrieval performance.

7.2 Composite Descriptors with Rectilinearity Measure

In this section, we discuss the composite shape descriptors that integrate several features via the linear combination of the distance values they produce, using fixed weights. The process of building a composite descriptor can be achieved in the following two steps:

1. *Feature selection*. Select several complementary descriptors which represent different aspects of a shape.
2. *Weight tuning*. Tune the weights of features in an independent training database to maximize the retrieval performance. For instance, we use a GA to automatically

Table 4 Retrieval performance of several descriptors

	1-NN	1-Tier	2-Tier	DCG
D1	25.7%	13.1%	19.3%	40.6%
D2	33.7%	16.4%	24.5%	44.4%
SHD	56.9%	28.8%	38.5%	57.2%
LFD	65.0%	38.3%	48.7%	64.4%
GSMD	67.1%	41.8%	52.0%	67.0%
GSMD + SHD	71.8%	44.5%	56.2%	69.8%

find the optimal weights to achieve an approximately maximum DCG with respect to base classification in the train set of the PSB database.

The goal of this section is to evaluate the retrieval performance of the composite features consisting of the rectilinearity measure and the following shape descriptors:

- D1: A histogram of distances from the center of mass to points on the surface (Ankerst et al. 1999). The number of histogram bins is selected as 64.
- D2: A histogram of distances between pairs of points on the surface (Osada et al. 2002). The number of histogram bins is chosen as 64.
- SHD: A vector consisting of spherical harmonic coefficients which are calculated from three spherical functions giving the maximal distance from center of mass as a function of spherical angle. The spherical functions differ due to their polar axes which are located at three axes of the canonical coordinate frame, respectively. The spherical harmonic descriptors are computed on a 64×64 spherical grid and then represented by its harmonic coefficients less than order 16. The dissimilarity between two objects is the minimum L_1 distance of six matching pairs. We use SpharmonicKit2.7 (2004) to calculate the spherical harmonic coefficients and the feature vector of each spherical function is normalized to its unit L_1 norm.
- LFD: See Sect. 7.1 for details. The length of the feature vector is 4700.
- GSMD: See Sect. 7.1 for details. Note that depth buffers instead of silhouettes are utilized here and the number of the viewpoints is selected as 66.

First, we separately test the descriptors without rectilinearity. Results are shown in Table 4.

Next, the rectilinearity values, with well tuned weights (using the train set of PSB), are added to the original signatures to form new features. Thus we obtain six new composite descriptors which are denoted as D1 + R, D2 + R, SHD + R, LFD + R, GSMD + R, and GSMD + SHD + R, respectively. Their retrieval results are demonstrated in Table 5. The distances between every pair of shape descriptors are calculated using their L_1 difference. Precision-recall plots for the original descriptors and corresponding composite descriptors are also compared in Fig. 22. We can see that considerable improvements have been achieved, mainly because the rectilinearity measure provides extra effective information with respect to the original shape descriptors.

Table 5 Retrieval performance of the descriptors combined with rectilinearity

	1-NN	1-Tier	2-Tier	DCG
D1 + R	34.2%	18.7%	28.2%	46.8%
D2 + R	43.0%	22.2%	32.0%	50.1%
SHD + R	62.2%	33.2%	45.2%	61.5%
LFD + R	67.5%	41.3%	53.6%	67.3%
GSMD + R	69.2%	44.2%	56.3%	69.4%
GSMD + SHD + R	73.1%	47.2%	60.2%	72.1%

Table 6 Comparing retrieval results of our method (first row) with state-of-the-art descriptors

	1-NN	1-Tier	2-Tier	DCG
GSMD + SHD + R	73.1%	47.2%	60.2%	72.1%
MDLA-DPD	68.8%	43.6%	54.2%	67.8%
LFD + AAD + SPRH	–	42.7%	52.7%	–
LFD(PSB)	65.7%	38.0%	48.7%	64.3%
SWC	46.9%	31.4%	39.7%	65.4%

Such kinds of combined signatures are fairly simple, but the result is encouraging. As shown in Table 6, compared to other state-of-the-art methods, such as, MDLA-DPD (Chaouch and Blondet 2007), LFD + AAD + SPRH (Ohbuchi and Hata 2006), LFD(PSB) (Shilane et al. 2004) and SWC (Laga et al. 2006), our method outperforms them in discrimination, occupies medium size of memory, and can be computed sufficiently quickly without special optimization.

8 Conclusion and Future Work

In this paper, we have proposed a novel rectilinearity measure, describing the extent to which a 3D mesh is rectilinear, and we also explicitly proved several corresponding theorems. The measure presented here has many desirable properties including simplicity, stability, robustness, and invariance to similarity transformation. We demonstrated how to compute it efficiently by a genetic algorithm. Afterwards, a series of experiments were carried out to validate the robustness as well as the effectiveness of our shape measurement in practice.

Furthermore, applications of 3D rectilinearity to *pose normalization* and *shape retrieval* were also investigated. First, we demonstrated that the calculation of rectilinearity

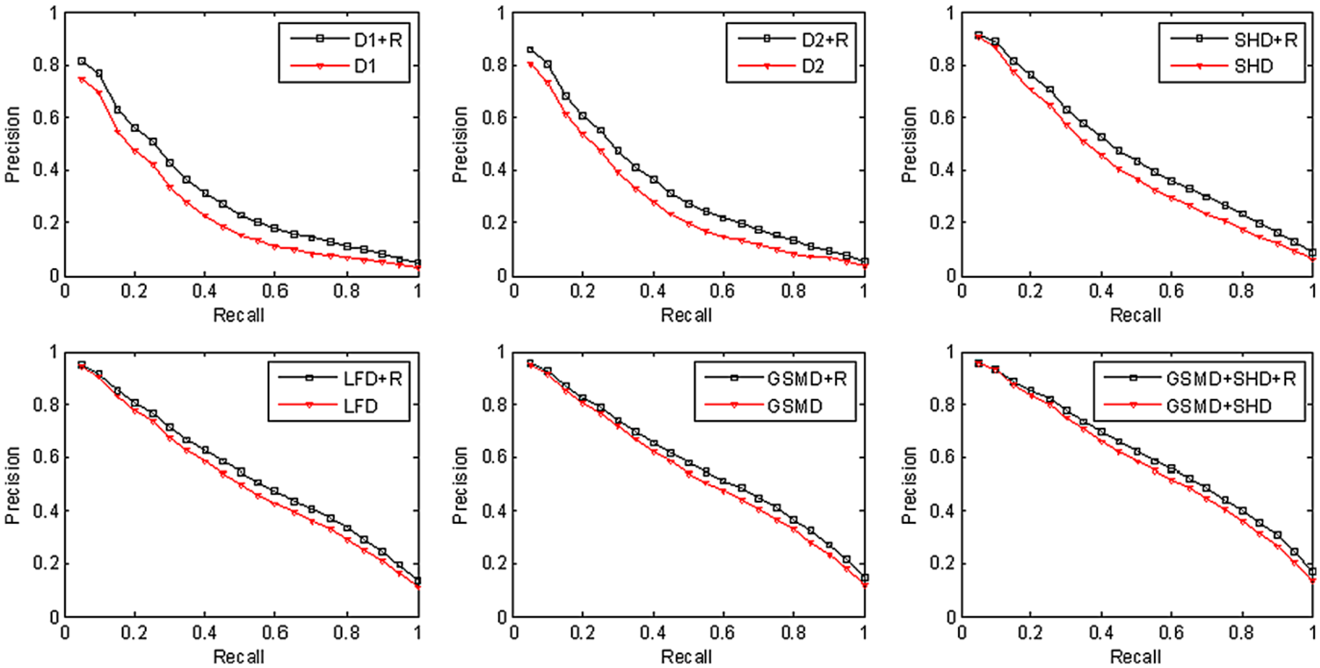

Fig. 22 Precision-recall curves calculated for 12 descriptors in the PSB test set with base classification. The precision-recall curves of a descriptor and its corresponding composite descriptor with rectilinearity measure are drawn together in a small figure

provided a new powerful tool to improve the pose normalization result of 3D meshes. Second, we applied a multi-view based shape matching mechanism for the normalized models and combined rectilinearity measure with other signatures to markedly enhance the performance of 3D shape retrieval.

Four directions for future investigation are listed as follows:

1. Is it possible to calculate the optimal value of the rectilinearity analytically, and how can this be done?
2. Can we derive other 3D shape measures, such as convexity, rectangularity, and compactness, using the relation between area and projected areas instead of the perimeter in 2D field in the same manner as we have described in this paper?
3. Can other selection criteria and properties be better integrated to build a stable and effective pose normalization method that can cope with all (or almost all) shapes?
4. Can other more discriminative 2D shape descriptors be introduced into our multi-view based shape retrieval mechanism to further improve the retrieval performance?

Acknowledgements This work was supported by China Scholarship Council and NSFC Grant 60674030.

Appendix

Lemma 1 $\frac{P(M,\alpha,\beta,\gamma)}{S(M)}$ *is a nonconstant function defined on* $\alpha, \beta, \gamma \in [0, 2\pi]$.

Proof We prove the lemma by a contradiction: $\frac{P(M,\alpha,\beta,\gamma)}{S(M)}$ is a constant function defined on $\alpha, \beta, \gamma \in [0, 2\pi]$. Assume that the unit normal of the triangle T_i is $n_i = [a_i, b_i, c_i]^T$. After rotation, we obtain a new normal

$$n_i' = R(\alpha, \beta, \gamma) n_i = \begin{bmatrix} a_i' \\ b_i' \\ c_i' \end{bmatrix} \tag{34}$$

where

$$a_i' = a_i \cos\gamma \cos\beta + b_i(\sin\gamma \cos\alpha + \cos\gamma \sin\beta \sin\alpha) + c_i(\sin\gamma \sin\alpha - \cos\gamma \sin\beta \cos\alpha) \tag{35}$$

$$b_i' = -a_i \sin\gamma \cos\beta + b_i(\cos\gamma \cos\alpha - \sin\gamma \sin\beta \sin\alpha) + c_i(\cos\gamma \sin\alpha + \sin\gamma \sin\beta \cos\alpha) \tag{36}$$

$$c_i' = a_i \sin\beta - b_i \cos\beta \sin\alpha + c_i \cos\beta \cos\alpha. \tag{37}$$

And we have

$$\frac{P(M, \alpha, \beta, \gamma)}{S(M)} = \frac{\sum_{i=1}^{N}(|a_i'|S_i + |b_i'|S_i + |c_i'|S_i)}{\sum_{i=1}^{N} S_i}. \tag{38}$$

Let α, β be constants and ensure that not all faces of the mesh are parallel to *XOY* plane. Then we can find an interval $\gamma \in [\gamma_0, \gamma_0 + \varepsilon]$, $\varepsilon > 0$ such that

$$P(M, \alpha, \beta, \gamma) = \sum_{i=1}^{N} \left(f a_i \cdot a_i' \cdot S_i + f b_i \cdot b_i' \cdot S_i + f c_i \cdot c_i' \cdot S_i \right) \tag{39}$$

and

$$P_{YOZ}(M,\alpha,\beta,\gamma)+P_{ZOX}(M,\alpha,\beta,\gamma)$$
$$=\sum_{i=1}^{N}\left(fa_i\cdot a_i'\cdot S_i+fb_i\cdot b_i'\cdot S_i\right)>0 \tag{40}$$

where $\{fa_i, fb_i, fc_i | i=1,2,\dots,N\}\subset\{+1,-1\}$.

Since $\frac{P(M,\alpha,\beta,\gamma)}{S(M)}$ and $S(M)$ both are constant functions, $P(M,\alpha,\beta,\gamma)$ should also be a constant function defined on $\alpha,\beta,\gamma\in[0,2\pi]$. Then we obtain

$$0=\frac{d(P(M,\alpha,\beta,\gamma))}{d\gamma}=\frac{d^2(P(M,\alpha,\beta,\gamma))}{d\gamma^2}$$
$$=\sum_{i=1}^{N}\left(-fa_i\cdot a_i'\cdot S_i-fb_i\cdot b_i'\cdot S_i\right)<0 \tag{41}$$

when α,β are the given constants and $\gamma\in[\gamma_0,\gamma_0+\varepsilon]$, $\varepsilon>0$. The obtained contradiction $0<0$ proves the lemma. □

References

Ankerst, M., Kastenmuller, G., Kriegel, H., & Seidl, T. (1999). Nearest neighbor classification in 3D protein databases. In *Proc. the seventh international conference on intelligent systems for molecular biology* (pp. 34–43).

Bribiesca, E. (2008). An easy measure of compactness for 2D and 3D shapes. *Pattern Recognition*, *41*(2), 543–554.

Bustos, B., Keim, D., Saupe, D., Schreck, T., & Vranić, D. (2005). An experimental effectiveness comparison of methods for 3D similarity search. *International Journal on Digital Libraries 6*(1), 39–54.

Chaouch, M., & Blondet, A. (2006). Enhanced 2D/3D approaches based on relevance index for 3D-shape retrieval. In *Proc. IEEE international conference on shape modeling and applications (SMI 2006)* (pp. 36–36).

Chaouch, M., & Blondet, A. (2007). A new descriptor for 2D depth image indexing and 3D model retrieval. In *Proc. IEEE international conference on image processing (ICIP 2007)* (Vol. 6, pp. 373–376).

Chaouch, M., & Blondet, A. (2008). A novel method for alignment of 3D models. In *Proc. IEEE international conference on shape modeling and applications (SMI 2008)* (pp. 187–195).

Chen, D. Y., Tian, X. P., Shen, Y. T., & Ouhyoung, M. (2003). On visual similarity based 3D model retrieval. In *Proc. Eurographics 2003* (Vol. 22, pp. 223–232).

Corney, J., Rea, H., Clark, D., Pritchard, J., Breaks, M., & MacLeod, R. (2002). Coarse filters for shape matching. *IEEE Computer Graphics and Applications*, *22*(3), 65–74.

Fink, E., & Wood, D. (1996). Fundamentals of restricted-orientation convexity. *Information Sciences*, *92*(1), 175–196.

Fu, H., Cohen-Or, D., Dror, G., & Sheffer, A. (2008). Upright orientation of man-made objects. In *Proc. international conference on computer graphics and interactive techniques (ACM SIGGRAPH 2008)*.

Gal, R., Shamir, A., & Cohen-Or, D. (2007). Pose-oblivious shape signature. *IEEE Transactions Visualization and Computer Graphics*, *13*(2), 261–271.

Haralick, R. M. (1974). A measure for circularity of digital figures. *IEEE Trans. Systems, Man, and Cybernetics*, *4*, 394–396.

Holland, J. H. (1992). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control and artificial intelligence*. Cambridge: MIT.

Kazhdan, M. (2007). An approximate and efficient method for optimal rotation alignment of 3D models. IEEE Transactions Pattern Analysis and. *Machine Intelligence*, *29*(7), 1221–1229.

Kazhdan, M., Chazelle, B., Dobkin, D., Funkhouser, T., & Rusinkiewicz, S. (2003). A reflective symmetry descriptor for 3D models. *Algorithmica*, *38*(1), 201–225.

Kazhdan, M., Funkhouser, T., & Rusinkiewicz, S. (2003). Rotation invariant spherical harmonic representation of 3D shape descriptors. In *Proc. 2003 Eurographics/ACM SIGGRAPH symposium on geometry processing* (Vol. 43, pp. 156–164).

Krinidis, S., & Chatzis, V. (2008). Principal axes estimation using the vibration modes of physics-based deformable models. *IEEE Trans. Image Processing*, *17*(6), 1007–1019.

Laga, H., Takahashi, H., & Nakajima, M. (2006). Spherical wavelet descriptors for content-based 3D model retrieval. In *Proc. IEEE international conference on shape modeling and applications (SMI 2006)* (pp. 15–15).

Lee, C. H., Varshney, A., & Jacobs, D. W. (2005). Mesh saliency. *ACM Transactions on Graphics (ACM SIGGRAPH 2005)*, *24*(3), 659–666.

Leou, J., & Tsai, W. (1987). Automatic rotational symmetry determination for shape analysis. *Pattern Recognition*, *20*(6), 571–582.

Levin, D. T., Takarae, Y., Miner, A., & Keil, F. C. (2001). Efficient visual search by category: specifying the features that mark the difference between artifacts and animals in preattentive vision. *Perception and Psychophysics*, *63*(4), 676–697.

Lian, Z., Rosin, P. L., & Sun, X. (2008). A rectilinearity measurement for 3D meshes. In *Proc. ACM international conference on multimedia information retrieval (MIR'08)* (pp. 395–402).

Loncaric, S. (1998). A survey of shape analysis techniques. *Pattern Recognition*, *31*(8), 983–1001.

MeshLab1.1.0 (2008). http://meshlab.sourceforge.net/.

Ohbuchi, R., & Hata, Y. (2006). Combining multiresolution shape descriptors for 3D model retrieval. In *Proc. WSCG 2006*.

Osada, R., Funkhouser, T., Chazelle, B., & Dobkin, D. (2002). Shape distributions. *ACM Transactions on Graphics*, *21*(4), 807–832.

Paquet, E., & Rioux, M. (1999). Nefertiti: a query by content system for three-dimensional model and image databases management. *Image and Vision Computing*, *17*(2), 157–166.

Paquet, E., Rioux, M., Murching, A., Naveen, T., & Tabatabai, A. (2000). Description of shape information for 2-D and 3-D objects. *Signal Processing: Image Communication*, *16*(1–2), 103–122.

Petitjean, M. (2003). Chirality and symmetry measures: A transdisciplinary review. *Entropy*, *5*, 271–312.

Podolak, J., Shilane, P., Golovinskiy, A., Rusinkiewicz, S., & Funkhouser, T. (2006). A planar-reflective symmetry transform for 3D shapes. *ACM Transactions on Graphics (ACM SIGGRAPH 2006)*, 549–559.

Proffitt, D. (1982). The measurement of circularity and ellipticity on a digital grid. *Pattern Recognition*, *15*(5), 383–387.

Rosin, P. L. (1999). Measuring rectangularity. *Machine Vision and Applications*, *11*(4), 191–196.

Rosin, P. L. (2003). Measuring shape: ellipticity, rectangularity, and triangularity. *Machine Vision and Applications*, *14*(3), 172–184.

Rosin, P. L. (2008). A two-component rectilinearity measure. *Computer Vision and Image Understanding*, *109*(2), 176–185.

Ruggeri, M. R., & Saupe, D. (2008, in press). Isometry-invariant matching of point set surfaces. In *Proc. Eurographics workshop on 3D object retrieval*.

Shih, J., Hsing, C., & Wang, J. (2007). A new 3D model retrieval approach based on the elevation descriptor. *Pattern Recognition*, *40*(1), 283–295.

Shilane, P., & Funkhouser, T. (2007). Distinctive regions of 3D surfaces. *ACM Transactions on Graphics*, *26*(2), 659–666.

Shilane, P., Min, P., Kazhdan, M., & Funkhouser, T. (2004). The Princeton shape benchmark. In *Proc. shape modeling applications (2004)* (pp. 167–178).

SpharmonicKit2.7 (2004). http://www.cs.dartmouth.edu/~geelong/sphere/.

Sundar, H., Silver, D., Gavani, N., & Dickinson, S. (2003). Skeleton based shape matching and retrieval. In *Proc. shape modeling international 2003 (SMI 2003)* (pp. 130–139).

Tangelder, J. W., & Veltkamp, R. C. (2008). A survey of content based 3D shape retrieval methods. *Multimedia Tools and Applications*, *39*(3), 441–471.

Tangelder, J. W. H., & Veltkamp, R. C. (2003). Polyhedral model retrieval using weighted point sets. In *Proc. shape modeling international 2003 (SMI 2003)* (pp. 119–129).

The Stanford 3D Scanning Repository (2008). http://graphics.stanford.edu/data/3dscanrep/.

Vazquez, P., Feixas, M., Sbert, M., & Heidrich, W. (2003). Automatic view selection using viewpoint entropy and its application to image-based modelling. *Computer Graphics Forum*, *22*(4), 689–700.

Vranić, D. V. (2005). Desire: a composite 3D-shape descriptor. In *Proc. IEEE international conference on multimedia and expo (ICME 2005)*.

Vranić, D. V., Saupe, D., & Richter, J. (2001). Tools for 3D-object retrieval: Karhunen-Loeve transform and spherical harmonics. In *Proc. 2001 IEEE fourth workshop on multimedia signal processing* (pp. 293–298).

Yamauchi, H., Saleem, W., Yoshizawa, S., Karni, Z., Belyaev, A., & Seidel, H. P. (2006). Towards stable and salient multi-view representation of 3d shapes. In *Proc. IEEE international conference on shape modeling and applications (SMI 2006)* (pp. 40–40).

Yang, Y., Lin, H., & Zhang, Y. (2007). Content-based 3-D model retrieval: a survey. *IEEE Trans. Systems, Man, and Cybernetics*, *37*(6), 1081–1098.

Zhang, C., & Chen, T. (2001a). Efficient feature extraction for 2D/3D objects in mesh representation. In *Proc. international conference on image processing (ICIP 2001)* (Vol. 3, pp. 935–938).

Zhang, C., & Chen, T. (2001b). Indexing and retrieval of 3D models aided by active learning. In *Proc. the ninth ACM international conference on multimedia* (pp. 615–616).

Zhang, D., & Lu, G. (2004). Review of shape representation and description techniques. *Pattern Recognition*, *37*(1), 1–19.

Žunić, J., & Rosin, P. L. (2003). Rectilinearity measurements for polygons. *IEEE Transactions Pattern Analysis and Machine Intelligence*, *25*(9), 1193–1200.

Žunić, J., & Rosin, P. L. (2004). A new convexity measure for polygons. *IEEE Transactions Pattern Analysis and Machine Intelligence*, *26*(7), 923–934.