Shape Isophotic Error Metric Controllable Re-Sampling for Point-sampled Surfaces

Yongwei Miao[1,2,3], Pablo Diaz-Gutierrez[4], Renato Pajarola[3], M. Gopi[4], Jieqing Feng[1]
[1]State Key Laboratory of CAD & CG, Zhejiang University, Zhejiang 310027, China
[2]College of Science, Zhejiang University of Technology, Zhejiang 310032, China
[3]Department of Informatics, University of Zürich, Zürich 8050, Switzerland
[4]Department of Computer Science, University of California, Irvine, CA 92697-3425, USA

***Abstract*— Shape simplification and re-sampling of underlying point-sampled surfaces under user-defined error bounds is an important and challenging issue. Based on the regular triangulation of the Gaussian sphere and the surface normals mapping onto the Gaussian sphere, a Gaussian sphere based re-sampling scheme is presented that generates a non-uniformly curvature-aware simplification of the given point-sampled model. Owing to the theoretical analysis of shape isophotic error metric for did that Gaussian sphere based sampling, the proposed simplification scheme provides a convenient way to control the re-sampling results under a user-specified error metric bound. The novel algorithm has been implemented and demonstrated on several examples.**

***Keywords*— point-sampled surfaces; isophotic error metric; Gaussian sphere; error metric controllable; re-sampling**

1. INTRODUCTION

Due to large memory requirements and high time complexity, efficiently processing of large-scale point-sampled surfaces is still facing great challenges such as efficient storage, editing, transmission, and rendering, etc [1], [2]. Surface simplification provides an efficient solution to mitigate storage and time complexities [3], [4]. For point-sampled surface simplification, one important and challenging issue is to re-sample the original geometry for faithful approximation of the underlying shape given a user-defined geometric error bound [4], [5].

According to the theory of sampling, the sampling size should be proportional to the total absolute Gaussian curvature of the surface [6], [7], which can directly be approximated by a regular sampling of the Gaussian sphere and mapping these Gaussian samples back onto the underlying surface [8]. Moreover, in view of shape error metric for surface re-sampling, compared with the pure geometric L^2 metric, a shape isophotic metric (also called $L^{2,1}$ metric and defined as the L^2 metric of the surface normal field) [8], [9], [10] can capture the anisotropy of the underlying geometry efficiently and is appropriate to shape re-sampling. Owing to these two aspects, based on the theoretical analysis of shape isophotic error metric for regular triangulation of the Gaussian sphere, a shape error metric controllable re-sampling scheme is presented in this paper to simplify the underlying point-sampled geometry.

The contributions are summarized as follows:

- Theoretical analysis of the shape isophotic error metric for the regular triangulation of a Gaussian sphere;
- A simplification scheme that provides a convenient way to control the re-sampling results under a user-specified isophotic error metric bound;
- An efficient algorithm to generate non-uniform curvature-aware adaptive point sample distributions.

The paper is organized as follows: The related work about error bound simplification methods for 3D models is reviewed in Section 2. In Section 3, the shape isophotic error is analyzed under the Gaussian sphere based sampling method. Using an adaptive neighborhood and Gaussian sphere sampling, an adaptive re-sampling scheme is proposed in Section 4. In Section 5, experimental results and a discussion are presented. Finally, conclusions are drawn and future research directions are given in Section 6.

2. RELATED WORK

Many researchers have developed techniques for 3D model simplification and decimation [3], [11]. However, guaranteeing an approximation tolerance during a re-sampling operation is an important requirement for many computer graphics applications [4], [9], [12]. Here, only some important techniques related to shape simplification with global approximation tolerance bounds are discussed.

For simplifying polygon meshes with an approximation error bound, the incremental mesh decimation scheme is most commonly adopted. Garland and Heckbert [12] associated to each original vertex the set of planes spanned by its adjacent faces, and defined the quadric error metric (QEM) as the sum of squared distances from the vertex to its associated planes. Their algorithm uses iterative contractions of vertex pairs to simplify models and also adopts the QEM to track

978-1-4244-4070-2/09/$25.00 © 2009 IEEE

the global approximate error. Furthermore, Garland and Heckbert [13] extended their earlier QEM framework to deal with both vertex position and vertex attributes in $\mathbb{R}^{3+m}$. The error metric is generalized to the sum of squared distances form the vertex to its associated hyper-planes. By extending these methods, Hoppe [14] introduced an improved quadric error metric for simplifying meshes with attributes, which is defined by both geometric and attribute errors based on geometric correspondence in 3D space. Extending directly from the mesh simplification scheme [12], Pauly et al. [4] presented an iterative point-simplification technique. It needs to maintain a dynamic topology for unstructured point clouds in order to perform point-pair contraction.

With regard to shape approximation error metrics, Cohen-Steiner et al. [9] proposed a k-means clustering algorithm for geometric approximation of surfaces under the two definitions of error metrics—L^2 metric and $L^{2,1}$ metric, which measure a generalized distance of a region to its respective proxy. The algorithm iteratively alternates between a geometry partitioning phase and a proxy fitting phase, and the approximation error is decreased by clustering faces into best-fitting regions iteratively. Recently, by attaching the field of unit normal vectors to the surface vertices as a vector-valued image, Lai et al. [15] presented the so-called image manifold, which using 3D surface maps associates each surface point $\mathbf{x} \in \mathbb{R}^3$ to a point $\mathbf{x}_f = (\mathbf{x}, \omega\mathbf{n}(x)) \in \mathbb{R}^6$. Applying the isotropic surface remeshing and sampling scheme to the so-called image manifold, they can achieve feature sensitive remeshing and sampling results which can generate more vertices in highly curved than in flat areas. Moreover, based on the mapping of regular sampling and triangulation of the Gaussian sphere onto a manifold surface, Diaz-Gutierrez and Gopi [8] applied results from quantization and surface approximation theory to propose a robust and output sensitive algorithm for re-sampling meshes with the purpose of surface approximation. Their Gaussian sphere based sampling scheme is related to our re-sampling algorithm for point-sampled geometry. Their sampling algorithm [8] depends heavily on the selection of feature edges, and partitions the input triangular mesh into featureless regions.

However, these simplification and re-sampling schemes are applicable to polygonal meshes, which possess globally consistent topological information, relying heavily on mesh connectivity information, edges between vertices. In contrast, our method is purely point-based, requiring only vertex positions and associated normals. This allows direct processing of scanned data without the need to construct polygonal meshes beforehand, making it particularly suitable for processing very large raw point models obtained with modern range scanners.

For point-sampled surfaces, Alexa et al. [16], [17] introduced a down-sampling algorithm, which creates a sub-sampled point cloud by ordering iterative point removal operations according to a surface error metric. The pure sub-sampling unnecessarily restricts potential sampling positions and can lead to aliasing artifacts. Pauly et al. [4] adopted uniform incremental and hierarchical clustering methods to simplify a given point-sampled model. The uniform incremental clustering approach is computationally efficient but is reported to cause high approximation errors. Similarly, the hierarchical clustering approach, which is efficient in memory and execution cost, even in its adaptive version has a high approximation error in general. The user also should carefully specify the variation threshold which is not an intuitive parameter. Moreover, these simplification schemes cannot take an apriori approximation error tolerance into account.

3. Shape Isophotic $L^{2,1}$ Error Metric Analysis

In order to adaptively re-sample the underlying 3D surface according to the Gaussian sampling algorithm outlined in the following section, the Gaussian sphere is regularly sampled and triangulated. This can be achieved by approximating the sphere by a recursive subdivision of an inscribed octahedron. In our terminology, the vertices and triangles of Gaussian sphere are called *Gaussian vertices* and *Gaussian triangles*. For simplifying point-sampled surfaces under an approximation error bound, the shape error metric should first be analyzed. As we know, the shape isophotic $L^{2,1}$ metric measures the distance between two sample points by the Euclidean distance of their corresponding Gaussian vertices projected on the Gaussian sphere. Thus in order to estimate the isophotic shape approximation error under a certain subdivision level of the Gaussian sphere, we must analyze the maximum Euclidean distance between Gaussian triangles on the sphere.

In general, if providing a subdivision level n of the Gaussian sphere, there are two types of spherical triangles on the Gaussian sphere (see Figure 1(a)). One is the spherical triangle $\triangle_1 = \{V_{i,j}, V_{i,j+1}, V_{i+1,j+1}\}$ for $i = 1, 2, \cdots, n-2, j = 0, 1, 2 \cdots, i-1$, and the other is the spherical triangle $\triangle_2 = \{V_{i,j}, V_{i+1,j}, V_{i+1,j+1}\}$ for $i = 0, 1, 2, \cdots, n-2, j = 0, 1, 2 \cdots, i$.

Now, without loss of generality, we only estimate the Euclidean distance between Gaussian vertices $V_{i,j}$ and $V_{i+1,j+1}$. The Cartesian coordinates of vertices $V_{i,j}$ and $V_{i+1,j+1}$ are

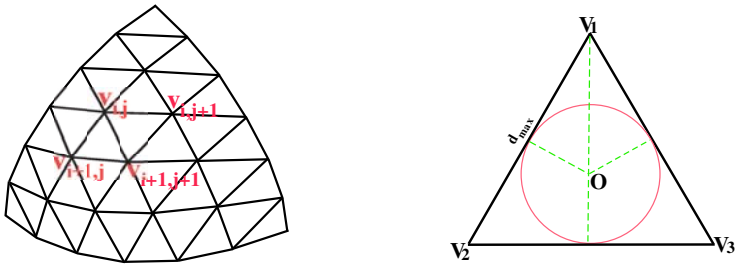
$$V_{i,j} = \Big(sin(\tfrac{i*\pi}{2(n-1)})cos(\tfrac{j*\pi}{2i}), sin(\tfrac{i*\pi}{2(n-1)})sin(\tfrac{j*\pi}{2i}), cos(\tfrac{i*\pi}{2(n-1)})\Big)$$

and

$$V_{i+1,j+1} = \big(sin(\frac{(i+1)*\pi}{2(n-1)})cos(\frac{(j+1)*\pi}{2(i+1)}), sin(\frac{(i+1)*\pi}{2(n-1)})sin(\frac{(j+1)*\pi}{2(i+1)}), cos(\frac{(i+1)*\pi}{2(n-1)})\big)$$

respectively. So, the square of the Euclidean distance between them can be estimated as follows:

$$d^2 = \big(sin(\frac{i\pi}{2(n-1)})cos(\frac{j\pi}{2i}) - sin(\frac{(i+1)\pi}{2(n-1)})cos(\frac{(j+1)\pi}{2(i+1)})\big)^2$$



(a) (b)

Fig. 1. *The triangulation of Gaussian sphere (a) and the distance analysis for equilateral triangle (b).*

$$+\big(sin(\frac{i\pi}{2(n-1)})sin(\frac{j\pi}{2i})-sin(\frac{(i+1)\pi}{2(n-1)})sin(\frac{(j+1)\pi}{2(i+1)})\big)^2$$

$$+\big(cos(\frac{i\pi}{2(n-1)})-cos(\frac{(i+1)\pi}{2(n-1)})\big)^2$$

$$=2-2cos(\frac{i\pi}{2(n-1)})cos(\frac{(i+1)\pi}{2(n-1)})$$

$$-2sin(\frac{i\pi}{2(n-1)})sin(\frac{(i+1)\pi}{2(n-1)})cos(\frac{j\pi}{2i}-\frac{(j+1)\pi}{2(i+1)})$$

$$\leq 2-2cos(\frac{i\pi}{2(n-1)})cos(\frac{(i+1)\pi}{2(n-1)})$$

$$-2sin(\frac{i\pi}{2(n-1)})sin(\frac{(i+1)\pi}{2(n-1)})\big(1-\frac{1}{2}(\frac{j\pi}{2i}-\frac{(j+1)\pi}{2(i+1)})^2\big)$$

$$=4\big(sin(\frac{\pi}{4(n-1)})\big)^2$$

$$+sin(\frac{i\pi}{2(n-1)})sin(\frac{(i+1)\pi}{2(n-1)})(\frac{j-i}{i(i+1)})^2\frac{\pi^2}{4}$$

$$\leq 4(\frac{\pi}{4(n-1)})^2+\frac{i*\pi}{2(n-1)}*\frac{(i+1)*\pi}{2(n-1)}*(\frac{j-i}{i(i+1)})^2*\frac{\pi^2}{4}$$

$$\leq \frac{\pi^2}{4(n-1)^2}(1+\frac{\pi^2}{4})$$

$$<\frac{7\pi^2}{8(n-1)^2}$$

In the above derivation of our estimation, we use the trigonometric in\equalities, and the simple relation $\frac{(j-i)^2}{i(i+1)} \leq 1$ for $0 \leq j \leq i$ and $i > 0$. Specifically, the distance between $V_{0,0} = (0,0,1)$ and $V_{1,1} = (0, sin(\frac{\pi}{2(n-1)}), cos(\frac{\pi}{2(n-1)}))$ can also be estimated easily, that is

$$d^2 = 2-2cos(\frac{\pi}{2(n-1)})$$

$$=4\big(sin(\frac{\pi}{4(n-1)})\big)^2$$

$$\leq \frac{\pi^2}{4(n-1)^2}$$

$$<\frac{7\pi^2}{8(n-1)^2}$$

and the other distances can be estimated similarly.

According to the above Euclidean distance estimation, the maximum Euclidean distance d_{max} between Gaussian vertices in both of these two types of spherical triangles is $\sqrt{\frac{7\pi^2}{8(n-1)^2}}$, given n-levels for the Gaussian sphere subdivision. Moreover, we can easily find that the distance between a triangle vertex and its barycenter is less or equal to $\frac{1}{\sqrt{3}}d_{max}$, which only happens for equilateral triangles (see Figure 1(b)). Hence we can control the level of subdivision according to a user-specified isophotic $L^{2,1}$ error metric threshold. In detail, for the error threshold ε, the subdivision level n should be greater than $\sqrt{\frac{7\pi^2}{24\varepsilon^2}}+1$, or $n = \lceil\sqrt{\frac{7\pi^2}{24\varepsilon^2}}+1\rceil$. For example, if the isophotic $L^{2,1}$ error threshold is 0.1, the subdivision level can be set to 18; while if the error threshold is 0.2, the level can be set to 10; while if the error threshold is 0.3, the level can be set to 7.

Similar to the geometric error analysis tool Metro [18], we measure the isophotic $L^{2,1}$ surface approximation error with respect to both the root mean squared (RMS) and the maximum error, i.e.,

$$\Delta_{RMS}(\mathcal{S}',\mathcal{S}) = \sqrt{\frac{1}{\|\mathcal{S}'\|}\sum_{\mathbf{p}\in\mathcal{S}'}\frac{\sum_{\mathbf{p}_i\in\Re_{\mathbf{p}}}\|\mathbf{n}_i-\mathbf{n}\|^2\omega(\mathbf{p}_i,\mathbf{p})}{\sum_{\mathbf{p}_i\in\Re_{\mathbf{p}}\subseteq\mathcal{S}}\omega(\mathbf{p}_i,\mathbf{p})}}$$

$$\Delta_{max}(\mathcal{S}',\mathcal{S}) = max_{\mathbf{p}\in\mathcal{S}'}\sqrt{\frac{\sum_{\mathbf{p}_i\in\Re_{\mathbf{p}}}\|\mathbf{n}_i-\mathbf{n}\|^2\omega(\mathbf{p}_i,\mathbf{p})}{\sum_{\mathbf{p}_i\in\Re_{\mathbf{p}}\subseteq\mathcal{S}}\omega(\mathbf{p}_i,\mathbf{p})}}$$

respectively, where $\Re_{\mathbf{p}} \subseteq \mathcal{S}$ represents the sample points in the original model whose representative point is $\mathbf{p}$ and the weight is given by $\omega(\mathbf{p}_i,\mathbf{p}) = \theta(\|\mathbf{p}_i-\mathbf{p}\|)$, where θ is a smooth, positive and monotonically decreasing function that specifies the influence of a sample point $\mathbf{p}_i$, such as the Gaussian weight $\theta(r) = e^{-r^2/h^2}$ as in [16], [17].

4. Adaptive Gaussian Sampling

4.1 Overview of Algorithm

Owing to the theoretical analysis of the shape isophotic $L^{2,1}$ error metric, we can re-sample a given point-sampled surface under a specific isophotic approximation error bound ε. For this, neighboring sample points are assigned to different clusters, unless the images of their normals on the Gaussian sphere are located in the same Gaussian triangle. That is, the isophotic $L^{2,1}$ distance within each cluster is bound by ε and the metric error can be controlled by the level n of subdivision of the Gaussian sphere as derived above.

Our algorithm takes as input a set of unstructured surfels rather than a discrete point cloud. Surface normals are generally available from 3D scanning systems or are

calculated as part of a standard point cloud preprocessing step [19]. The input surfels for our algorithm can be represented as clouds of point-normal pairs $\{(\mathbf{p}_i, \mathbf{n}_i)\}$. The basic steps of our algorithm for adaptive Gaussian sampling can be summarized as follows:

- Adaptive neighborhood generation for each sample point determined by normal deviation;
- Clustering sample points into regions by an index propagation scheme;
- Index rectification to eliminate singleton points which locate near corners of Gaussian triangles;
- Representation of each cluster region by a surfel as a representative point.

4.2 Adaptive Neighborhood for Sample Points

For each sample point in the original model, the traditional k-nearest neighborhood $\mathcal{N}_k$ has some inevitable drawbacks. The traditional k-nearest neighbors cannot reflect the normal variance and the requirement of sampling density. One potential solution to this issue is an adaptive neighborhood selection, which reflects the normal variance around sample points. According to the analysis of sampling criteria by Gopi et al. [20], the following adaptive selection for nearest neighbor points is adopted. For each sample point $\mathbf{p}$, the traditional neighborhood $\mathcal{N}_k$ is defined by k-nearest neighbors. Then, depending on the local sampling density and local feature size at point $\mathbf{p}$, the normals of the neighboring points $\mathbf{q} \in \mathcal{N}_k$ can span quite a wide range, that is, the bounding cone-of-normals of all normals of the points $\mathbf{q} \in \mathcal{N}_k$ is not constant for a fixed k-nearest neighborhood. Our adaptive nearest neighbors of a sample point $\mathbf{p}$ can be defined as all points $\mathbf{q}$ in $\mathcal{N}_k$ within some given angle of normal deviation. These neighbors span a (bounding) normal cone of a pre-defined opening angle. Due to the relation between the angle and the distance of surface normal we have $< \mathbf{n}_1, \mathbf{n}_2 >= 1 - \frac{1}{2}\|\mathbf{n}_1 - \mathbf{n}_2\|^2$. In our implementation, the bounded normal cone neighbors $\mathcal{N}(\mathbf{p})$ are defined as

$$\mathcal{N}(\mathbf{p}) = \{\mathbf{q} \in \mathcal{N}_k \| < \mathbf{n}_p, \mathbf{n}_q > \geq 1 - \frac{1}{2}d_{max}^2\}$$

where the d_{max} denotes the maximum length of Gaussian edge on the sphere.

4.3 Adaptive Re-sampling by Gaussian Sphere

Now, for each sample point $\mathbf{p}$ in the given point-sampled surface $\mathcal{S}$, the adaptive neighbors $\mathcal{N}(\mathbf{p})$ are obtained as discussed above. Then, according to the position information and neighborhood relation, an undirect non-symmetric abstract neighbor graph $\aleph = (P, E)$ can be constructed, which represents the non-symmetric neighbor information. In this abstract neighbor graph, the edge (i, j) belongs to E if and only if sample point $\mathbf{p}_j \in \mathcal{N}(\mathbf{p}_i)$.

The basic idea of our adaptive re-sampling algorithm for point-sampled surfaces is that neighboring sample points whose normal vectors lie inside the same Gauss triangle are considered to be located in a featureless region of the underlying surface, and therefore, should belong to one common cluster. On the other hand, two neighboring sample points whose normal vectors locate in different Gauss triangles should belong to two different clusters. Owing to the abstract graph associated with the adaptive neighborhood relation for the given point-sampled surface, the goal of our algorithm is to find a solution for labelling each graph vertex with an index and then cluster and replace the sample points with the same index by one representative surfel. Therefore, our index labelling procedure should satisfy the following criteria: for each graph edge, if the two end points have normals located in the same Gauss triangle then they should share the same index, while if two end points have normals located in different Gaussian triangles then they should be labelled differently. Our index propagation procedure for point-sampled surface models is summarized in Algorithm 1.

Algorithm 1 Index Propagation Procedure

INPUT: A given point-sampled surface model $\mathcal{S} = \{(\mathbf{p}_k, \mathbf{n}_k)\}_{k=1,2,3,\cdots,n}$
INPUT: The adaptive neighbors $\mathcal{N}$ for every sample point in $\mathcal{S}$
while *There is a new un-indexed sample point* $\mathbf{p}_l$ **do**
  Create new index label for $\mathbf{p}_l$
  Push $\mathbf{p}_l$ *onto a stack as a root point*
  while *Stack is not empty* **do**
    Pop the top point $\mathbf{p}_i$ *from the stack*
    for *Each sample point* $\mathbf{p}_j \in \mathcal{N}(\mathbf{p}_i)$ **do**
      if *Sample point* $\mathbf{p}_j$ *is a direct neighbor of the root point* $\mathbf{p}_l$ **and**
      The normal directions $\mathbf{n}_j$ *and* $\mathbf{n}_l$ *locate in the same Gauss triangle* **then**
        The label of $\mathbf{p}_j$ *is the same as* $\mathbf{p}_l$
        Push $\mathbf{p}_j$ *onto the stack*
      end if
    end for
  end while
end while

However, in order to avoid introducing badly formed clusters, some extra constraints for index propagation are introduced. If the normals of two neighboring points are located near two corners of one Gaussian triangle after mapping to the Gaussian sphere, then they most likely belong to two different clusters rather than falling into the same. Hence a distance constraint for normal directions is introduced to avoid this exceptional situation. That is, the Euclidean distance between two normal directions should not exceed the radius of the inscribed circle in the Gaussian triangle. Only if the two normal directions of two neighboring sample points belong to one Gaussian triangle and the Euclidean distance of normal directions is less than $\frac{1}{2\sqrt{3}}d_{max}$, then the index of the current sample point is propagated to the neighboring point. On the other hand, in the case of an almost planar big region, our sampling scheme will also generate a bad cluster. Hence a distance constraint for point positions is adopted, that is, a depth constraint of graph traversal in our index propagation procedure. We limit this depth constraint as 6, which works well in all of our experiments.

4.4 Eliminating Singleton Points

After our index propagation procedure, almost all sample points are given an appropriate index. However, in order to eliminate possible singleton points, an index rectification procedure should be performed to group these singleton points with neighboring clusters. For each singleton sample point $\mathbf{p}$ which locates near the corner of a Gaussian triangle, its neighbor clusters $\mathcal{C}_i$ are first determined. The normal of each neighbor cluster is then taken as the weighted average of the normals of its attached points (see Section 4.5), and the combining score of the cluster $\mathcal{C}_i$ can be assigned as $1- < \mathbf{n}, \mathbf{n}_i >$, where $\mathbf{n}$ and $\mathbf{n}_i$ denote the normals of the singleton sample point and the cluster respectively. Then, each singleton point can be grouped with the neighbor cluster with minimal score.

4.5 Generating Representative Surfels

For each cluster $\mathcal{C}_i$ obtained by the index labelling scheme, the representative surfel can be obtained by minimizing the $L^{2,1}$ metric [9], [14]. That is, the total $L^{2,1}$ error metric for the representative surfel $(\mathbf{p}, \mathbf{n})$ can be described as:

$$L(\mathbf{p}, \mathbf{n}) = \sum_i \omega_i L^{(\mathbf{p}_i, \mathbf{n}_i)}(\mathbf{p}, \mathbf{n}) = \sum_i \omega_i (\mathbf{n} - \mathbf{n}_i)^T (\mathbf{n} - \mathbf{n}_i)$$

For the sake of simplicity, if the weight ω_i is taken as irrelevant to the surface normal, the minimization can be determined easily by the following equation:

$$\mathbf{n} = \frac{\sum_i \omega_i \mathbf{n}_i}{\sum_i \omega_i}$$

which gives the normal of the representative surfel after normalization. Coincidentally, it is similar to the average normal definition in [21]. The weight is given by $\omega_i = \theta(\|\mathbf{p}_i - \mathbf{p}\|)$, where θ is taken as the same weight function as surface approximation error measurement (see Section 3). On the other hand, the position information is irrelevant to the $L^{2,1}$ minimization. In our implementation, the position of the proxy surfel is chosen as the mean of the sample points in the cluster, that is $\mathbf{p} = \frac{1}{k} \sum_i \mathbf{p}_i$.

5. Results and Discussion

The algorithm presented in this paper has been implemented and tested on a PC with a Pentium IV 3.0 GHz CPU and 1GB memory. In our implementation, the Gaussian sphere based sampling scheme can adaptively reflect the intrinsic property of the underlying 3D model and can account for the local geometric features. In particular, it provides a convenient way for the user to control the shape error metric, that is, it can adjust the regular sampling and triangulation of the associated Gaussian sphere to cater for the required isophotic $L^{2,1}$ error.

5.1 Adaptive Re-sampling of Different Point Models

In the Gaussian sphere based re-sampling method, the essential issue is to cluster sample points using their normal direction distribution. In general, surface normals give us first order information of the underlying surface variation around the sample point, and the variation of normal directions can reflect the intrinsic curvature distribution to some extent. Hence the proposed algorithm can generate non-uniform adaptively distributed discrete sample points. Figure 2 shows the adaptive re-sampling results for different point-base surface models. For the sake of unification and comparison between different models, the subdivision level of the Gaussian sphere is fixed for that experiment to $n = 7$ and the size of the adaptive sample point neighborhood in each model is $6 - 16$. The experimental results show that the proposed re-sampling scheme adapts excellently to the different curvature distributions of the underlying models.

Fig 3 demonstrates the re-sampling results of applying our Gaussian sampling algorithm to raw scanned point clouds. The experiments show that our re-sampling results can also reflect the curvature distribution of the underlying models even for non-uniformly sampled Buddha model and for raw scanned noisy Woman model.

Table 1 shows the data and timing statistics for the test models and the Gaussian sampling algorithm introduced in this paper. For example, the Dragon model originally contains 437645 vertices and is simplified to 40390 points satisfying root mean squared (RMS) and maximum isophotic errors of $\Delta_{RMS} = 0.0564$ and $\Delta_{max} = 0.1432$ respectively. The timing of subsampling, index propagation and representative surfel generation is 9.17s. Table 1 also clearly demonstrates that at the same RMS error, more feature-rich surfaces such as the Stanford Armadillo or Buddha models are less reduced in percentage of points than smoother and feature-less models like the Stanford Bunny or Dragon.

Figure 4 shows a comparison to previous clustering methods for point cloud simplification [4]. Note that the number of sample points for the different simplified models by different clustering methods is similar. It shows the quantitative error estimates for the Max-Planck model that has been simplified as about 11.5% of original points. Our Gaussian sampling scheme always leads to the lowest shape isophotic $L^{2,1}$ error in terms of the RMS error and the maximum error. The experimental results indicate that our Gaussian sampling scheme can generate adaptive and non-uniformly distributed discrete sample points in a curvature-aware manner, and the intrinsic geometric features can be preserved well in the reconstructed version of the simplified model.

5.2 Isophotic $L^{2,1}$ Error Controlled Re-sampling via Different Gaussian Sphere Sampling

In our proposed re-sampling framework, the shape isophotic $L^{2,1}$ error can be controlled easily by the

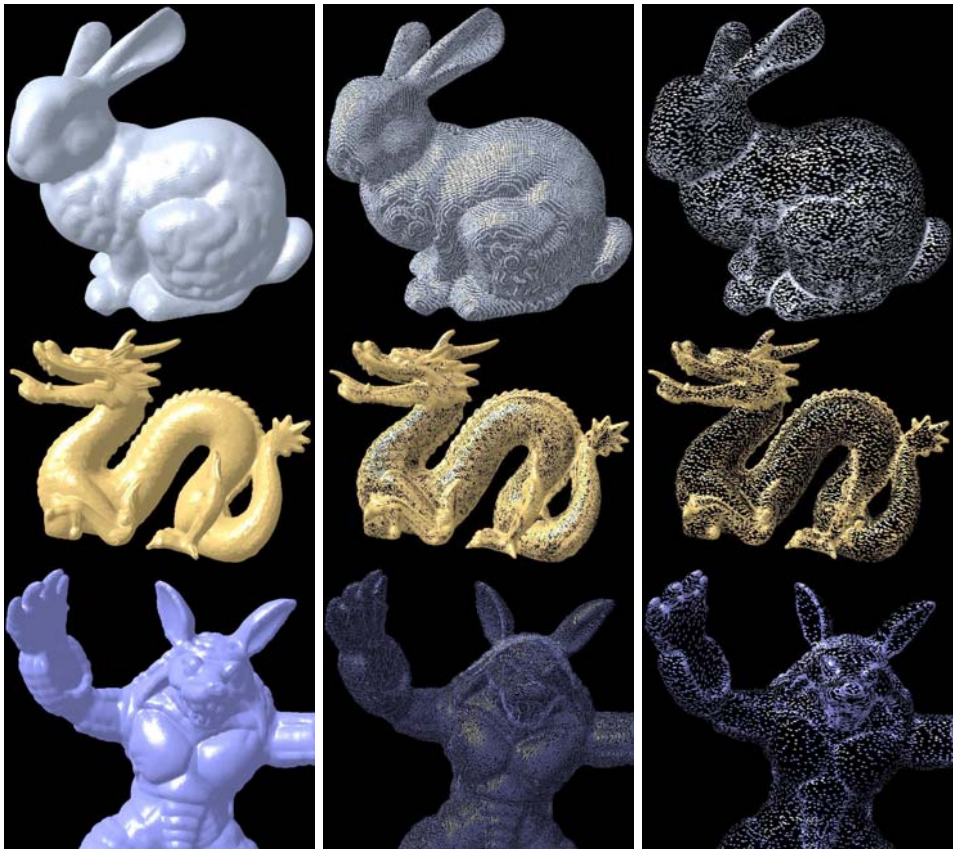


Fig. 2. *Adaptive re-sampling results for different point-sampled models by our Gaussian sampling scheme. First column: Original surface models. Second column: Original sample points distribution. Third column: Adaptive sampling results based on our Gaussian sampling method.*

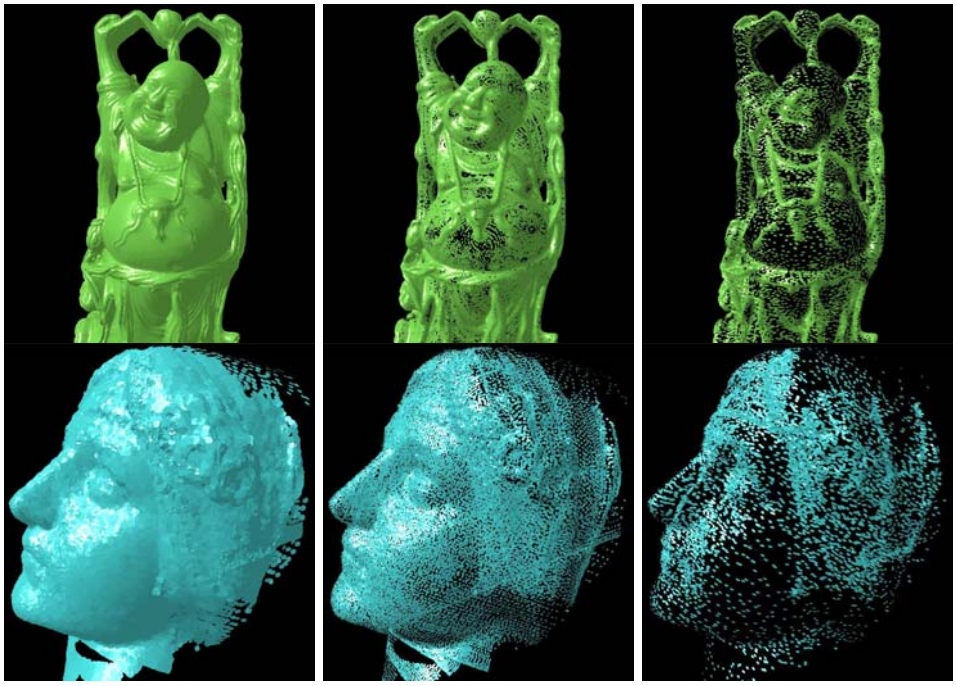
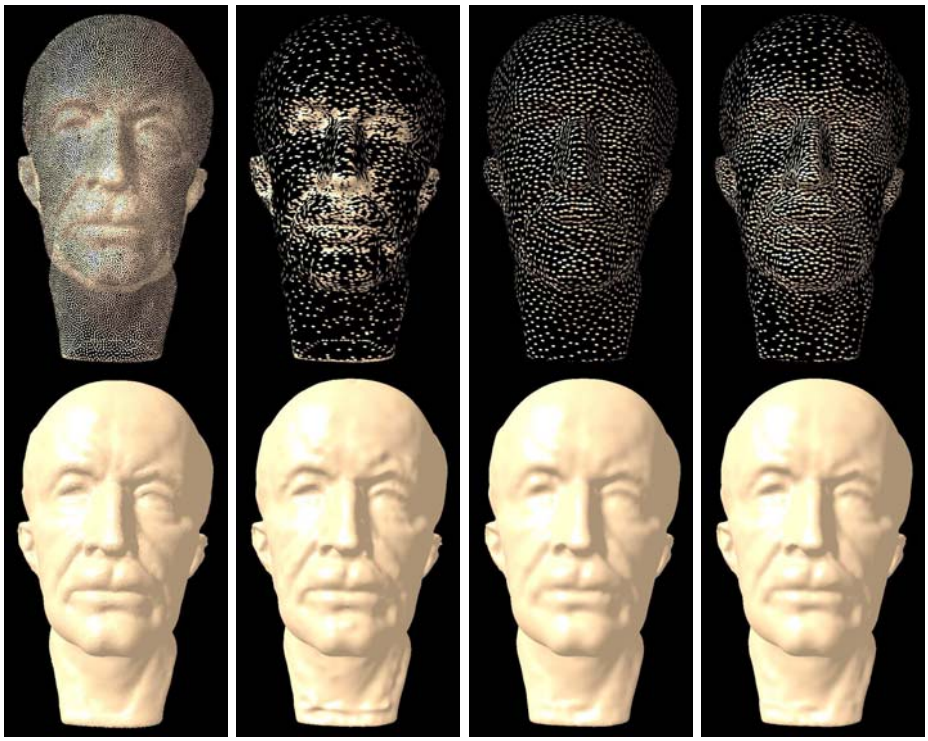


Fig. 3. *Adaptive re-sampling results by our Gaussian sampling scheme. First row: The original sampling and our adaptive re-sampling result for non-uniformly sampled Buddha model. Second row: The original sampling and our adaptive re-sampling result for raw scanned noisy Woman model.*

Model	#Points	Timing	#Simplified Points (% of original)	Isophotic Error	
				RMS error	Max Error
Buddha	543652	13.35s	69613 (12.8%)	0.0523	0.1468
Dragon	437645	9.17s	40390 (9.2%)	0.0564	0.1432
Bunny	280792	5.69s	19883 (7.1%)	0.0508	0.1296
Armadillo	172974	4.28s	26554 (15.4%)	0.0565	0.1334
Balljoint	137062	3.19s	15133 (11.0%)	0.0536	0.1274
Santa	75781	1.87s	10565 (13.9%)	0.0599	0.1303
Noisy Woman	55368	1.56s	17308 (31.3%)	0.0397	0.1245
Max-Planck	52809	1.18s	6048 (11.5%)	0.0521	0.1189

TABLE 1

Statistics and Timings for our Gaussian sampling approach for different point-sampled models. Note: In our implementation, the subdivision level of Gaussian sphere is chosen as $n = 7$, and the number of adaptive neighbors per sample points in each model is $6 - 16$.



(a) (b) (c) (d)

Fig. 4. *Re-sampling results of different clustering methods for the Max-Planck model. (a) Original surface model. (b),(c),(d) Re-sampling results and surface reconstructions by our Gaussian sampling scheme ($\Delta_{RMS} = 0.0521$ and $\Delta_{max} = 0.1189$), uniform incremental clustering [4] ($\Delta_{RMS} = 0.1420$ and $\Delta_{max} = 1.0453$) and adaptive hierarchical clustering [4] ($\Delta_{RMS} = 0.1437$ and $\Delta_{max} = 1.2640$), respectively.*

subdivision level of the Gaussian sphere. According to the user-specified isophotic error bound ε, the system can determine the subdivision level n automatically as described in Section 3. Figure 5 shows the experimental re-sampling results via different subdivision levels for the Stanford bunny model (the number of initial samples is 280792). If the subdivision level is chosen as $n = 7$, the RMS and maximum isophotic errors are $\Delta_{RMS} = 0.0508$ and $\Delta_{max} = 0.1296$, and the number of points in the simplified model is 19883. However, if the subdivision level is chosen as $n = 18$, the RMS and maximum isophotic errors decrease to $\Delta_{RMS} = 0.0164$ and $\Delta_{max} = 0.0496$, and the number of points in the simplified model is 62401. The experimental results show that increased levels of Gaussian sphere subdivision will lead to larger number of sample points in the final simplified model, as expected.

6. Conclusions and Future Work

Based on Gaussian sphere sampling, a novel error controllable re-sampling approach for point-sampled surfaces is presented. Our simplification scheme can generate non-uniformly distributed discrete sample points, adaptively reflecting the intrinsic geometric features of the underlying 3D surface model, and it also provides a convenient way to control the re-sampling results under a user-specified isophotic error metric bound. According to the user defined shape simplification error, the algorithm can easily adjust the regular sampling and triangulation of the associated Gaussian sphere to achieve the required error bound.

However, one limitation of our Gaussian sampling scheme is that the point clouds with large noise or highly non-uniform sampling cannot be treated well. For these raw scanner data, some pre-processing steps [19]

(a) (b) (c) (d)

Fig. 5. *Adaptive re-sampling results via different levels of Gaussian sphere subdivision. (a) Original sampling of the Stanford bunny model. (b), (c), (d) Show different adaptive sampling results for the Stanford bunny using* $n = 7$, 10 *and* 18 *levels of Gaussian sphere subdivision, respectively.*

should be performed for subsequent simplification and re-sampling task.

Based on the proposed sampling approach, future research should focus on geometry processing tasks for point-sampled surfaces, such as model compression, real-time rendering, streaming processing, etc.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their valuable comments. This research work is supported by the State Scholarship Fund of China. This work is also partially supported by the National Natural Science Foundation of China under Grant No. 60743002, the Natural Science Foundation of Zhejiang Province under Grant Nos. Y107311, R106449, the China Postdoctoral Science Foundation under Grant No. 20070421184, and the Swiss National Science Foundation under Grant No. 200021-111746/1.

REFERENCES

[1] M. Gross and H. Pfister (editors). Point-based graphics. Series in computer graphics, Morgan Kaufmann Publishers, 2007.

[2] M. Zwicker, M. Pauly, O. Knoll, and M. Gross. Pointshop 3d: An interactive system for point-based surface editing. ACM Transactions on Graphics, 21(3): 322-329, 2002.

[3] P. Heckbert and M. Garland. Survey of polygonal surface simplification algorithms. In Proc. of ACM SIGGRAPH 1997 Course Notes.

[4] M. Pauly, M. Gross, and L. P. Kobbelt. Efficient simplification of point-sampled surfaces. In Proc. of IEEE Visualization, pages 163-170, 2002.

[5] J. Wu and L. P. Kobbelt. Optimized sub-sampling of point sets for surface splatting. Computer Graphics Forum, 23(3): 643-652, 2004.

[6] M. Gopi. Theory and Practice of Sampling and Reconstruction for Manifolds with Boundaries. PhD thesis, Department of Computer Science, UNC Chapel Hill, 2001.

[7] K. Clarkson. Building triangulations using ε-net. In Proc. of the Thirty-Eighth Annual ACM Symposium on theory of Computing, pages 326-335, 2006.

[8] D.-G. Pablo and M. Gopi. Gaussian sphere sampling based surface approxiamtion. Technical Report, University of California, Irvine, 2007.

[9] D. Cohen-steiner, P. Alliez, and M. Desbrun. Variational shape approximation. ACM Transaction on Graphics, 23(3): 905-914, 2004.

[10] H. Pottmann, T. Steiner, M. Hofer, C. Haider, and A. Hanbury. The isophotic metric and its application to feature sensitive morphology on surfaces. In Proc. of ECCV, Part IV, pages 560-572, 2004.

[11] M. Botsch, M. Pauly, L. P. Kobbelt, P. Alliez, B. Levy, S. Bischoff, and C. Rossl. Geometric modeling based on polygonal meshes. In Proc. of ACM SIGGRAPH 2007 Course Notess.

[12] M. Garland and P. Heckbert. Surface simplification using quadric error metrics. In Proc. of ACM SIGGRAPH, pages 209-216, 1997.

[13] M. Garland and P. Heckbert. Simplifying surfaces with color and texture using quadric error metrics. In Proc. of IEEE Visualization, pages 263-269, 1998.

[14] H. Hoppe. New quadric metric for simplifying meshes with appearance attributes. In Proc. of IEEE Visualization, pages 59-66, 1999.

[15] Y.-K. Lai, Q.-Y. Zhou, S.-M. Hu, J. Wallner, and H. Pottmann. Robust feature classification and editing. IEEE Transactions on Visualization and Computer Graphics, 13(1): 34-45, 2007.

[16] M. Alexa, J. Behr, D. Cohen-or, S. Fleishman, D. Levin, and C. Silva. Point set surfaces. In Proc. of IEEE Visualization, pages 21-28, 2001.

[17] M. Alexa, J. Behr, D. Cohen-or, S. Fleishman, D. Levin, and C. Silva. Computing and rendering point set surfaces. IEEE Transactions on Visualization and Computer Graphics, 9(1): 3-15, 2003.

[18] P. Cignoni, C. Rocchini, and R. Scopigno. Metro: Measuring error on simplified surfaces. Computer Graphics Forum, 17(2): 167-174, 1998.

[19] T. Weyrich, M. Pauly, R. Keiser, S. Heinzle, S. Scandella, and M. Gross. Post-processing of scanned 3D surface data. In Proc. of Eurographics Symposium on Point-Based Graphics, pages 85-94, 2004.

[20] M. Gopi, S. Krishnan, and C. Silva. Surface reconstruction based on lower dimensional localized delaunay triangulation. In Proc. of EUROGRAPHICS, pages 467-478, 2000.

[21] M. Alexa and A. Adamson. On normals and projection operators for surfaces defined by point sets. In Proc. of Eurographics Symposium on Point-Based Graphics, pages 149-155, 2004.