2012 Second Joint 3DIM/3DPVT Conference: 3D Imaging, Modeling, Processing, Visualization & Transmission

An Evaluation Method for Multiview Surface Reconstruction Algorithms

Ran Song, Yonghuai Liu, Yitian Zhao
Department of Computer Science,
Aberystwyth University, UK
{res,yyl,yyz10}@aber.ac.uk

Ralph Martin, Paul Rosin
School of Computer Science and Informatics,
Cardiff University, UK
{ralph.martin,paul.rosin}@cs.cardiff.ac.uk

Abstract

We propose a new method, 3DGiC, for evaluating the performances of various multiview surface reconstruction methods. It does not need a complete ground truth model, providing a much wider range of applications. More importantly, most existing methods only measure the quality of reconstruction in a global *manner but* local *surface details are not involved. In contrast, 3DGiC depends on both* global *consistency and* local *accuracy of reconstruction in order to deliver a more comprehensive evaluation. The key idea is to compute a cumulative distribution of the joint probability of two local surface descriptors. We also designed experiments based on both synthetic and real data to demonstrate the advantages as well as the effectiveness of 3DGiC.*

1. Introduction

3D surface reconstruction from multiview datasets is a classical problem and remains active. Its goal is to reconstruct a complete surface model from multiple datasets captured from different viewpoints. Fig. 1 shows the reconstructions of the Bird, Teletubby and Frog datasets (multiview range images) using various algorithms. Through a visual comparison, it might be reasonable to claim that the reconstructions shown in the last three columns outperform other ones. However, for these three competing methods, it is not convincing to compare their reconstruction results but visually because they all look reasonable but have different flaws. In this case, a quantitative evaluation is needed.

1.1. Existing evaluation methods

We categorise existing methods into two groups according to whether a complete ground truth is required.

It is difficult to do evaluation without a complete ground truth model. In this case, the evaluation usually focuses on how much the reconstruction is consistent with the input datasets (viewed as a set of partial ground truth models). *Reconstruction error* [1] is computed as the average Euclidean distance from input points to the reconstructed surface. Similarly, *integration error* [19, 20] calculates the average of the Euclidean distances between the points in the final reconstructed surface and their closest points in the input data source. [18] quantifies the accuracy of the reconstruction by measuring an average per-point distance of the range data to the reconstruction.

Once a complete ground truth is available, a direct evaluation is readily implemented. [17] employs the mean square errors of different reconstructions against the known ground truth for comparisons. [5] measures the standard deviations of reconstructions to the ground truth under different levels of noise. *shape error* [6, 12] is calculated by the ratio between the volume of the symmetric difference between the estimated surface and the ground truth and the volume of the ground truth. To measure the *accuracy* of a reconstruction, [13] calculates the signed distances between the points in the reconstructed model and their closest points on the ground truth model. The output is a single distance value such that 90% (as suggested by the authors) of the reconstruction is within this distance threshold of the ground truth model. [2] proposed a three-step method to evaluate reconstruction methods where the ground truth models were produced via a commerical optical laser scanner.

A major weakness of the existing methods is that they mostly estimate *global accuracy* of reconstructions but neglect *local accuracy*. Global accuracy is used to give us a sense of whether a reconstruction tends to over- or underestimate the true shape. Local accuracy measures how consistent a local surface of the reconstruction is with its corresponding local surface on the true shape. For example, in Fig. 2, the reconstruction in (a) has a better *accuracy* according to [13]. But we believe that in most applications, the reconstruction in (b) is viewed more successful. Also, in [20], the reconstructed Bird model with a significant oversmoothing effect yielded a small *integration error* [20] while other better reconstructed models with more local surface details produce larger integration errors. The similar phenomenon also lies in the Frog model where the result of the quantitative evaluation via integration error was inconsistent with the qualitative comparison.

978-0-7695-4873-9/12 $26.00 © 2012 IEEE
DOI 10.1109/3DIMPVT.2012.24

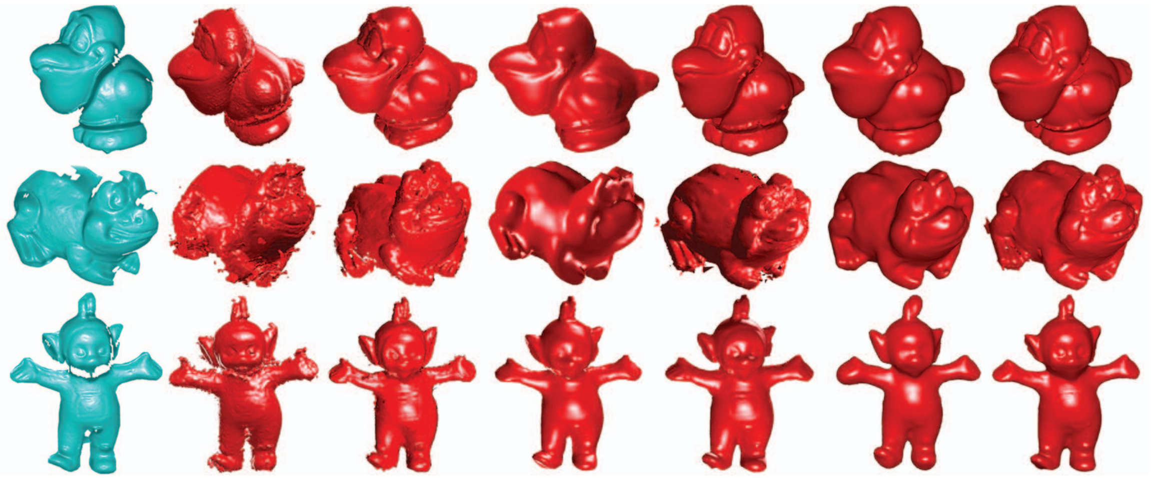


Figure 1. Rows: Reconstructions using 17 Bird range images, 17 Frog range images and 17 Teletubby range images respectively. From left to right: one single range image used as partial ground truth, volumetric method [3], mesh-based method [15], SFK method [20], k-means clustering-based method [19], pairwise MRF-based method [11], higher-order CRF-based method [14]

1.2. Requirements for a good measure

An evaluation method for multiview surface reconstruction is a specific measure of shape dissimilarity. Therefore, a good measure should satisfy some general requirements [10] desirable for a shape dissimilarity method and some specific properties beneficial to this particular application:

(1) Invariance: a good measure should be invariant to rigid transformation. Clearly, if the reconstructed surface is precisely superimposed with the ground truth after a certain rigid transformation, it should be viewed as a perfect reconstruction. The point here is that in such an evaluation, we always try to measure the surface error/difference rather than the pose error/difference.

(2) Robustness: a good measure should be robust to small perturbations. This property provides insensitivity (but not invariance) to noise and outliers. A simple method to achieve robustness is to use a strategy introducing a ratio between the measured difference and the ground truth or the input data as the output of the evaluation. Robustness is essentially a dynamic representation relying on the relative magnitude of the perturbation. Because we usually compare the magnitude of the perturbation to the size of the surface model to see whether it is 'small' or not, it is natural to use a ratio as the form of the assessment to achieve the robustness requirement.

(3) Generality: a good measure should be independent of the representation of the surface models. It is desired to be able to cope with input data in a variety of representations such as 3D unstructured point cloud, range scans, polygon soup, meshes, 2.5D slice and voxels, etc. We require that the measure can handle the input data of a surface model with or without connectivity information in both implicit and explicit representations.

(4) Applicability: a good measure should have a wide range of applications. Usually, an evaluation method which does not require a complete ground truth model is easy to use in most applications.

1.3. The proposed methodology

The proposed methodology for evaluating multiview surface reconstruction algorithms, namely 3D Gini Coefficient or 3DGiC for short, is inspired by Gini Coefficient, a well-known measure of statistical dispersion in economics and sociology. As illustrated in Fig. 3(a), the Gini Coefficient measures the inequality of a distribution by applying a ratio of the difference between such a distribution and the distribution representing sheer equality over the one representing sheer equality. The key idea to adapt Gini Coefficient into our work is to construct a parameterised probability distribution corresponding to an arbitrary surface model discriminatively. Then we can use the idea of Gini Coefficient to measure the difference between pairs of distributions.

We briefly summarise the advantages of our method in terms of the requirements listed in Section 1.2. Firstly, we develop two local surface descriptors invariant to rigid transformations and the following computation is completely based on them. Consequently, 3DGiC is invariant to rigid transformations. In contrast, most existing measures used for evaluating multiview surface reconstruction approaches [19, 1, 6, 13] do not have such invariance. Secondly, 3DGiC is robust to small perturbations while the existing methods relying only on distances do not have such robustness because the proposed local surface descriptors are not over-sensitive to noise, as demonstrated in Section 2.3. Thirdly, the proposed evaluation method can cope with different types of input data such as 3D unstructured point clouds and range images. However, the shape error [6] is

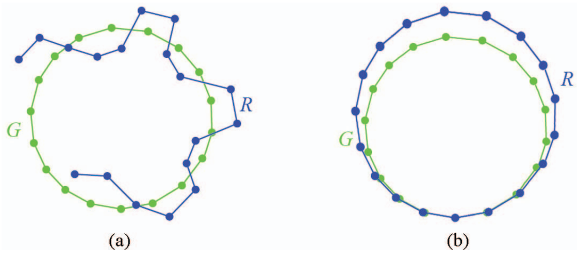


Figure 2. The *accuracy* methodology [13] neglects local surface details, leading to unfair comparison and evaluation. Here the blue circles are the reconstructed surface and the green circles represent the ground truth. Globally, the reconstruction in (a) is more accurate; locally, the reconstruction in (b) is more accurate. Please note that figure (a) is directly taken from Figure 3(a) of [13].

designed to compute the volume difference between two surfaces, which thus requires the connectivity information from the input data. Finally, a complete ground truth is not necessary for our method while its presence can make the evaluation more efficient.

In general, the key idea is to compute the cumulative distribution of a joint probability of local surface descriptors. The a cumulative distribution represents global statistics while the local surface descriptors are used for evaluating local accuracy. In this way, both global and local considerations are incorporated, which leads to a more comprehensive evaluation.

2. Algorithmic details

There are three stages to the estimation of 3DGiC: (1) detecting overlapping areas; (2) calculating 3D Lorenz surfaces; (3) computing the volume ratio.

2.1. Overlapping area detection

We assume that there is no complete ground truth model available. The only objective and reliable data that can be used for the evaluation are the input data. The first step is to calculate the overlapping area between an input point cloud (or range image, mesh, etc) and the reconstructed surface.

Let $I = \{I_m | m = 1, 2, ...n\}$ be the multiview input point clouds. After registration via [7], we obtain a collection of registered point clouds $\{I'_m | m = 1, 2, ...n\}$. These point clouds and R, the set of vertices on the reconstructed surface, are in the same coordinate system. We then find the 3 nearest neighbours for each point in I'_m from R. Next, we collect all the nearest neighbours as the overlapping point set R_m from R. The surface area covered by R_m is viewed as the overlapping area between R and the input point cloud I_m. This scheme is very simple but functional and fast when using a k-D tree speedup. A more advanced overlapping area detection method can be found in [19]. Usually, R_m and I_m contain different number of points. The process of overlapping area detection is illustrated by Fig. 4.

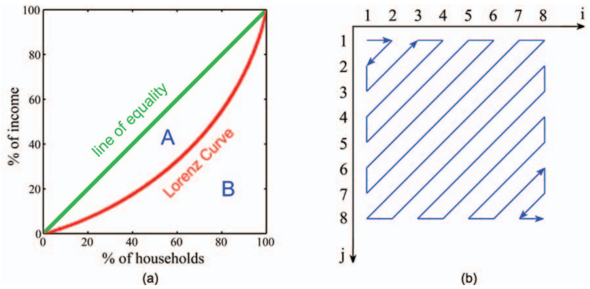


Figure 3. (a) The diagonal represents perfect equality of incomes. Gini Coefficient can then be thought of as the ratio of the area that lies between the line of equality and the Lorenz curve (marked 'A') over the total area under the line of equality (marked 'A' and 'B'); i.e., $G = A/(A + B)$. (b) The zigzag order of an 8×8 matrix

2.2. 3D Lorenz surface

In this paper, we propose the concept of 3D Lorenz surface. It is inspired by the Lorenz curve [9]. Lorenz curve is often used to describe the cumulative probability distribution of income graphically. By comparing one Lorenz curve with the line of equality (as shown in Fig. 3 (a)), we can calculate the Gini Coefficient measuring how dispersive the income of different groups of people in the society is.

The proposed 3D Lorenz surface is an extension of the idea of Lorenz curve in a 3D space. It is a cumulative distribution function of the joint probability distribution of two *transformed curvatures*. We develop a three-step scheme to compute the 3D Lorenz surface of a surface.

(1) **Transformed curvatures**. We use two transformed curvatures to calculate the Lorenz surface because both of them are invariant to rotation and translation. It means that ideally, after registration, the transformed curvatures should remain unchanged. They represent the *local* accuracy as it is only related to local surface details.

Given a surface, we employ a simple but fast method [16] to compute the principal curvatures k_1 and k_2 for each vertex. Then we compute the two *transformed curvatures*

$$d_1 = \frac{2}{\pi} \arctan k_1, \quad d_2 = \frac{2}{\pi} \arctan k_2. \tag{1}$$

The colour maps of d_1 and d_2 are shown in Fig. 5. The reason that we do not directly use k_1 and k_2 in this algorithm will be explained in Section 2.3.

Then we use N_1 and N_2 bins for computing the histograms of d_1 and d_2 over all the vertices on the surface respectively (as shown in Fig. 6). The quantization parameters N_1 and N_2 are important for the performance of 3DGiC. First, the histograms will probably remain the same if the transformed curvatures of every point slightly change. The quantization thus makes the measure not oversensitive to globally-distributed noise as it tolerates small errors on the transformed curvatures. The larger the quantization parameters, the more tolerant (insensitive) the mea-

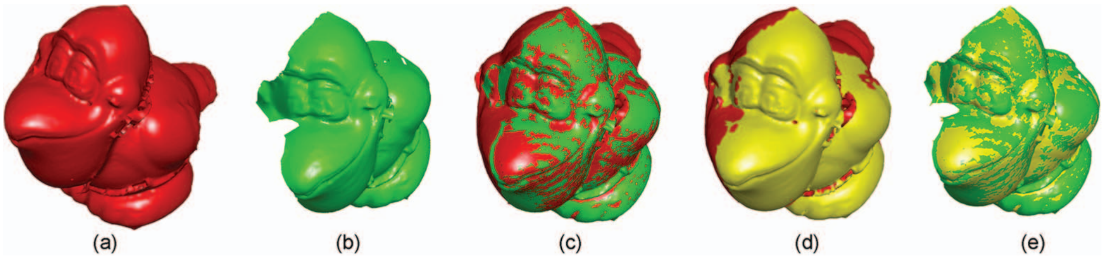


Figure 4. Overlapping area detection: (a) The reconstructed surface (b) One single view of the input (c) They are overlapped after registration (d) The surface area in yellow is detected as the overlapping area (e) They usually contain different number of points

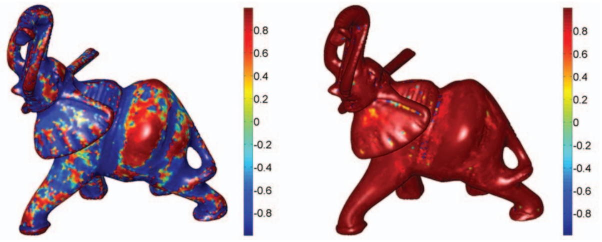


Figure 5. The two transformed curvatures of an elephant model. Left: Transformed curvature d_1; Right: Transformed curvature d_2

sure to errors and noise. Second, N_1 and N_2 largely decide whether the Lorenz surface is smooth or not, and further affect the final value of 3DGiC. Third, the computational time of 3DGiC is also related to the two parameters.

(2) **Lorenz surface estimation**. The joint probability distribution of d_1 and d_2 of all vertices on a surface can be calculated via:

$$p_{ij} = \frac{x_{ij}}{N}, \quad i = 1, 2, ..., N_1 \quad \text{and} \quad j = 1, 2, ..., N_2 \quad (2)$$

where x_{ij} indicates the number of the vertices whose transformed curvatures lie in the ith bin out of N_1 bins and the jth bin out of N_2 bins respectively. N is the number of vertices on the surface and

$$N = \sum_{i,j} x_{ij}, \quad i = 1, 2, ..., N_1 \text{ and } j = 1, 2, ..., N_2 \quad (3)$$

We then reorder $P = \{p_{ij} | i = 1, ..., N_1; j = 1, ..., N_2\}$ using the following procedure. We first calculate the zigzag order for the $N_1 \times N_2$ array of P as illustrated in Fig. 3 (b). Then we sort the $N_1 \times N_2$ p_{ij}s in P in ascending order. We do this reordering to make sure that the Lorenz surface is always under the diagonal plane, which complies with the nature of the Lorenz curve (as shown in Fig. 3 (a), the Lorenz curve is always under the line of equality). Fig. 7 shows the Lorenz surfaces produced with and without this reordering scheme for comparison. A Lorenz surface partitions the unit cube into two parts. One has a volume larger than 0.5 and the volume of the other one is smaller than 0.5. The reordering actually makes sure that the algorithm uses the smaller one to calculate the 3DGiC, leading to meaningful result. Once we obtained the reordered P, written as $P' = \{p'_{ij} | i = 1, ..., N_1; j = 1, .., N_2\}$, the cumulative distribution S of the joint probability can be calculated as:

$$S = \{s_{uv} | u = 1, ..., N_1; v = 1, ..., N_2\} \quad (4)$$

where

$$s_{uv} = \sum_{i=1,j=1}^{i=u,j=v} p'_{ij} \quad u = 1, ..., N_1 \text{ and } v = 1, ..., N_2. \quad (5)$$

Here, the order of summation is also the zigzag order.

The Lorenz surface L is the surface represented by the 3D point set $\{\mathbf{c}_{uv} | \mathbf{c}_{uv} = (\frac{u}{N_1}, \frac{v}{N_2}, s_{uv})\}$. It can be seen that any two Lorenz surfaces must intersect at point $(\frac{1}{N_1}, \frac{1}{N_2}, 0)$ (as the smallest p_{ij} is always equal to 0) and point $(1, 1, 1)$.

(3) **3DGiC computation**. Gini coefficient reflects how different a Lorenz curve is from the line of equality. In this paper, we extend this idea and propose the concept of 3DGiC. First, it measures how different one Lorenz surface is from the other one. Second, it is defined as the ratio of the volume that lies between the two Lorenz surfaces over the volume under the Lorenz surface on the top. Note that two Lorenz surfaces are possibly intersected. Thus, in Eq. 6, the denominator is chosen as the larger one between $V(L(R_m))$ and $V(L(I_m)$.

Let the Lorenz surfaces of R_m and I_m be $L(R_m)$ and $L(I_m)$ respectively. $V(L(R_m))$ and $V(L(I_m))$ denote the volumes under the two Lorenz surfaces respectively, the 3DGiC of R and I_m can be calculated as

$$G(R, I_m) = \frac{|V(L(R_m)) - V(L(I_m))|}{\max(V(L(R_m)), V(L(I_m)))}. \quad (6)$$

To measure the overall consistency between R and all of the multiple input point clouds I, we can calculate the mean 3DGiC $G(R, I) = \frac{1}{n} \sum_{m=1}^{m=n} G(R, I_m)$. The evaluation based on the 3DGiC is thus very intuitive: the smaller the mean 3DGiC, the better the reconstruction.

2.3. Transformed curvatures vs. curvatures

We use transformed curvatures instead of principal curvatures to compute the Lorenz surfaces. The reason is that

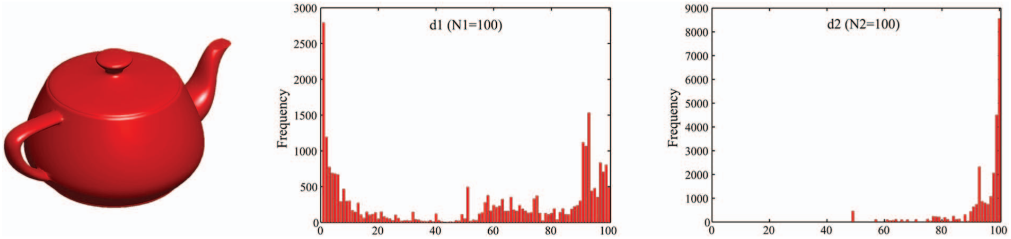


Figure 6. Left: The surface model of a teapot; Middle: The histogram of d_1; Right: The histogram of d_2. Please note that here the transformed curvatures d_1 and d_2 have been quantized with the quantization parameters $N_1 = N_2 = 100$.

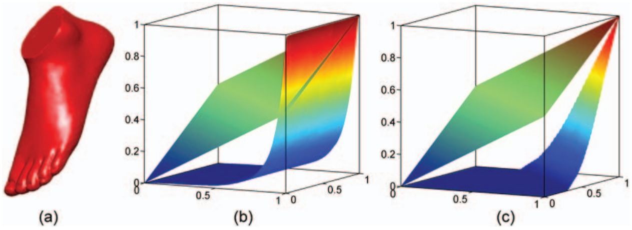


Figure 7. (a) The surface model of a foot; (b) the Lorenz surface generated without the reordering scheme intersects with the diagonal plane; (c) the Lorenz surface generated with the reordering scheme is always under the diagonal plane.

the principal curvatures tend to produce a Lorenz surface quite close to the bottom right corner as shown in Fig. 8 (a) and (b). If the volume under the Lorenz surface is excessively small, the 3DGiC will be over-sensitive to the change of the numerator in Eq. (6). In this case, even if the difference between two surfaces are small, their 3DGiC will still be close to 1. This is not desired because the 3DGiC will mostly distribute in a narrow interval.

In addition, if one surface contains plenty of noise, its Lorenz surface tends to bend towards the bottom right corner as shown in Fig. 8 (c) and (d). The noise makes the distributions of transformed curvatures more discontinuous because it usually produces a bumpy surface. Therefore the transformed curvatures fall into a smaller number of bins after quantisation. Eventually, the joint probability distribution p_{ij} becomes more concentrated.

3. Experiments

We conducted two experiments to examine the validity of the proposed method for objective assessment. In Experiment 1, noise was added to the test datasets. We employed the proposed 3DGiC and the *accuracy* [13] measure for evaluations respectively. In Experiment 2, we ran different multiview surface reconstruction methods and compute 3DGiCs for quantitatively comparing these reconstruction methods. We set $N_1 = N_2 = 100$ to balance the speed of implementation and the smoothness of the Lorenz surface. All experiments were implemented on a Pentium due core 2.4GHz, 3.25GB RAM computer.

3.1. Experiment 1

Test data and procedure. We used the Skull and Hippo models as shown in Fig. 9. We added zero-mean Gaussian noise to the ground truth models and generated two noisy models. Then we computed their 3DGiC and *accuracy* [13]. We also generated two test surface models by simply introducing a 0.3mm translation along the x direction to the ground truth surface models. We computed the 3DGiC and the *accuracy* [13] for the displaced surface models.

Results and discussion. The results are shown in Fig. 9 and Table. 1. With a value of 0.4665, the 3DGiC of the noisy skull sufficiently reflects how different it is from the ground truth. Although 3DGiC is not over-sensitive to small perturbations, it has the ability to recognise different levels of 'wrongness' as illustrated in Fig. 10 where we added different levels of Gaussian noise into the Stanford Bunny.

As shown in Table. 1, the 3DGiC of the displaced Skull and Hippo models is 0. This is desired as the displaced model is in fact a perfect reconstruction. In contrast, the accuracy of the test models is misleading. For the noisy Skull, with 90% of its points being within 0.1114*mm* of the ground truth model cannot give us a convincing sense of whether the reconstruction is accurate or not. For instance, the noisy Skull was clearly not better reconstructed than the one with displacement although that is what the accuracy measure indicated.

3.2. Experiment 2

Test data. In this experiment, we used three sets of real multiview range images from the Minolta database [4] to reconstruct complete 3D surface models. The thumbnails of these range images are shown in Table. 2-4.

Procedure. Since each range image can only cover partial surface of the object and is posed in their individual coordinate system, we employed an automatic registration method [7] to align them in a global coordinate system. Registration errors are shown in Table. 2-4. For each input set of range images, we used the three most successful multiview reconstruction methods [19, 11, 14] demonstrated in Fig. 1 to produce three different surface models (shown in the 4th, 5th and 6th columns in Fig. 1) respectively.

Results and discussion. The last three columns of each

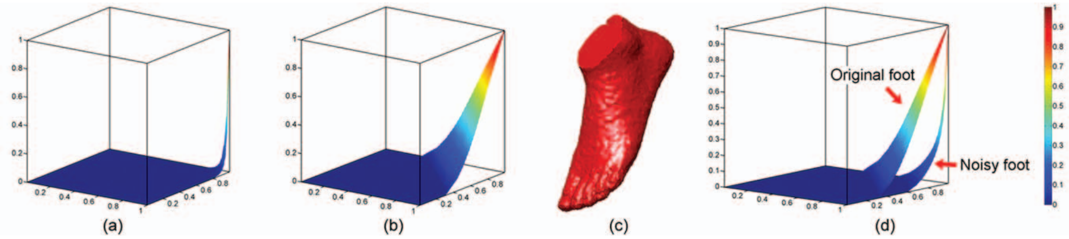


Figure 8. (a) The Lorenz surface of the foot produced by directly using principal curvatures for quantisation and the volume under the Lorenz surface is 0.0011 (b) The Lorenz surface of the foot produced by using transformed curvatures and the volume under the Lorenz surface is 0.0603 (c) The foot model with some zero-mean Gaussian noise (d) The Lorenz surface of the noisy foot tends to bend towards the bottom right corner where both Lorenz surfaces are produced using the proposed transformed curvatures.

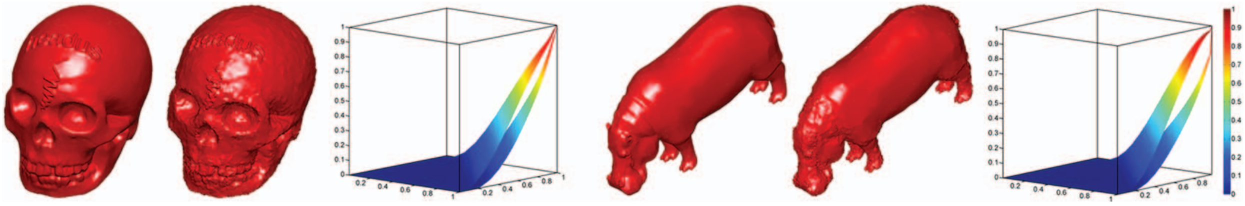
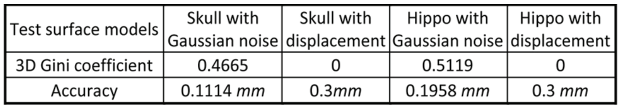


Figure 9. From left to right: The ground truth skull; The skull with Gaussian noise; The Lorenz surfaces of the the ground truth skull and the noisy skull; The ground truth hippo; The hippo with Gaussian noise; The Lorenz surfaces of the ground truth hippo and the noisy hippo



Test surface models	Skull with Gaussian noise	Skull with displacement	Hippo with Gaussian noise	Hippo with displacement
3D Gini coefficient	0.4665	0	0.5119	0
Accuracy	0.1114 *mm*	0.3*mm*	0.1958 *mm*	0.3 *mm*

Table 1. The 3DGiC and the accuracy [13] of different test models

of Table. 2-4 show the 3DGiCs of the three surface models produced by the three reconstruction methods. We can see that 3DGiC is very informative. First, according to the average 3DGiC, in general, the higher-order CRF-based method [14] (corresponding to Model 1 in Table. 2-4) produced the best reconstructions. The k-means clustering-based method [19] (corresponding to Model 3 in Table. 2-4) had a worse performance while the pairwise MRF-based method [11] (corresponding to Model 2 in Table. 2-4) was the worst. Second, it is possible that a reconstruction is generally poor but has a small patch of local surface region better reconstructed. For example, for the 7th input scan of the Frog (Table. 4), the pairwise MRF-based method achieved the lowest 3DGiC. Third, the 3DGiC of each individual input range image gives us a clue that whether some parts of the output complete surface model are well or poorly reconstructed. In Table. 2, for Model 1 of the Bird, the 3DGiC of the 8th range scan is as high as 0.2223. Considering that the average 3DGiC of the whole dataset is merely 0.0812, we know that some parts of the surface covered by the 8th range scan are not well reconstructed. Similarly, in Table. 4, for Model 1 of the Frog, from the 3DGiC corresponding to the 12th input range scan, we know that some parts of the surface covering the back of the Frog are not well reconstructed. This is very useful in practice. For example, the improvement of the surface quality can only focus on the partial surface poorly reconstructed, which makes the process more efficient.

If we compare the 3DGiCs calculated in Experiment 1 with those calculated in Experiment 2, we can have the sense that in most of the cases, some noise distributed throughout the surface tends to have more significant impact than registration error on 3DGiC. This is reasonable as such noise directly destroys local surface geometry and human perception is quite sensitive to it. For example, a bumpy surface is tended to be viewed as a poorer reconstruction than one surface under- or over-estimated to the ground true shape as we cannot easily distinguish the latter one from the true shape. Such kind of bumpy surface caused by a particular noise usually cannot be reflected and further recognised by existing evaluation methods such as *accuracy* (see Fig. 2 for a better understanding). In contrast, the proposed 3DGiC carries out a quantitative evaluation which complies more with human's habit of visual perception while sometimes it is indeed very difficult to judge which reconstruction is better visually.

Table. 5 shows the computational time of 3DGiC over different datasets. The algorithm needs more computational time when the quantisation parameters increase, but not significantly. This is easy to understand because we need more time to calculate the joint probability distribution of the quantised transformed curvatures as the joint probability $P = \{p_{ij}\}$ has more elements. In Table. 5, besides the 3 datasets of multiview range images, we also use four synthetic datasets with ground truth. We produce a new surface model by adding some Gaussian noise into the ground

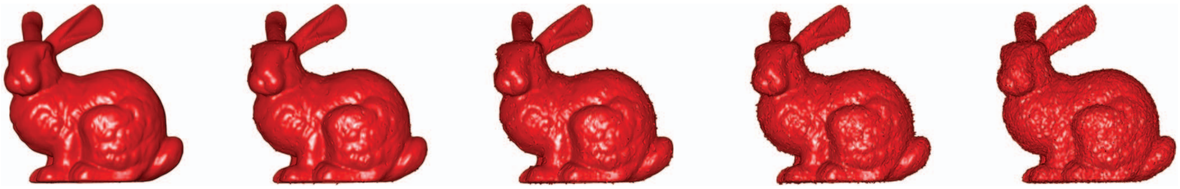
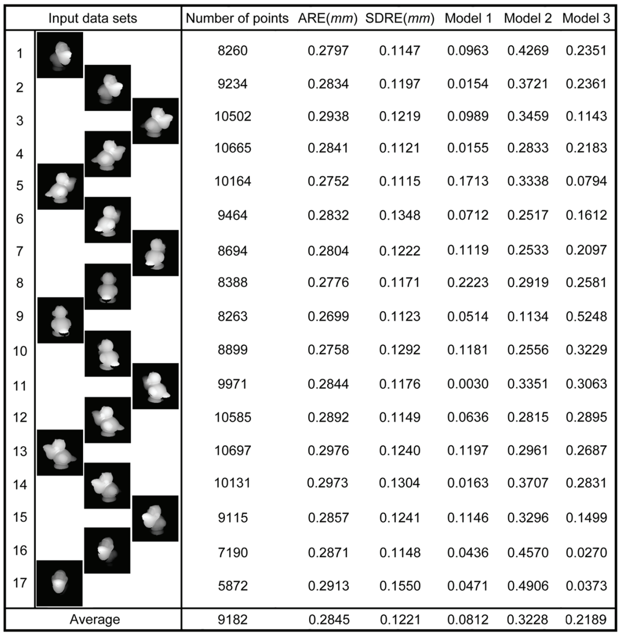
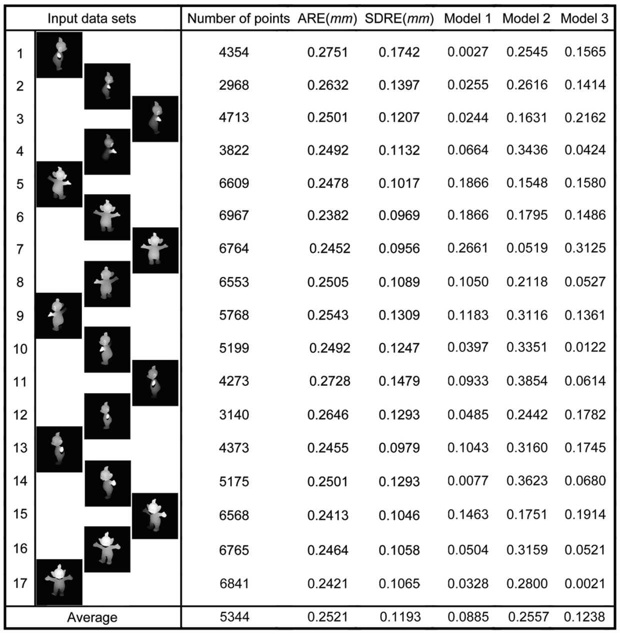


Figure 10. 3DGiCs of Bunny models with different levels of noise. From left to right: the first model is the original Bunny; The 3DGiCs of the 4 Bunny models with different levels of noise are 0.2093, 0.4370, 0.6457 and 0.8262 respectively.



Input data sets	Number of points	ARE(*mm*)	SDRE(*mm*)	Model 1	Model 2	Model 3
1	8260	0.2797	0.1147	0.0963	0.4269	0.2351
2	9234	0.2834	0.1197	0.0154	0.3721	0.2361
3	10502	0.2938	0.1219	0.0989	0.3459	0.1143
4	10665	0.2841	0.1121	0.0155	0.2833	0.2183
5	10164	0.2752	0.1115	0.1713	0.3338	0.0794
6	9464	0.2832	0.1348	0.0712	0.2517	0.1612
7	8694	0.2804	0.1222	0.1119	0.2533	0.2097
8	8388	0.2776	0.1171	0.2223	0.2919	0.2581
9	8263	0.2699	0.1123	0.0514	0.1134	0.5248
10	8899	0.2758	0.1292	0.1181	0.2556	0.3229
11	9971	0.2844	0.1176	0.0030	0.3351	0.3063
12	10585	0.2892	0.1149	0.0636	0.2815	0.2895
13	10697	0.2976	0.1240	0.1197	0.2961	0.2687
14	10131	0.2973	0.1304	0.0163	0.3707	0.2831
15	9115	0.2857	0.1241	0.1146	0.3296	0.1499
16	7190	0.2871	0.1148	0.0436	0.4570	0.0270
17	5872	0.2913	0.1550	0.0471	0.4906	0.0373
Average	9182	0.2845	0.1221	0.0812	0.3228	0.2189

Table 2. The Bird dataset in Experiment 2 and the resultant 3DGiCs of the three reconstructed surface models (listed in the columns of Model 1, Model 2 and Model 3). ARE: Average registration error [8]. SDRE: Standard deviation of registration errors [8].



Input data sets	Number of points	ARE(*mm*)	SDRE(*mm*)	Model 1	Model 2	Model 3
1	4354	0.2751	0.1742	0.0027	0.2545	0.1565
2	2968	0.2632	0.1397	0.0255	0.2616	0.1414
3	4713	0.2501	0.1207	0.0244	0.1631	0.2162
4	3822	0.2492	0.1132	0.0664	0.3436	0.0424
5	6609	0.2478	0.1017	0.1866	0.1548	0.1580
6	6967	0.2382	0.0969	0.1866	0.1795	0.1486
7	6764	0.2452	0.0956	0.2661	0.0519	0.3125
8	6553	0.2505	0.1089	0.1050	0.2118	0.0527
9	5768	0.2543	0.1309	0.1183	0.3116	0.1361
10	5199	0.2492	0.1247	0.0397	0.3351	0.0122
11	4273	0.2728	0.1479	0.0933	0.3854	0.0614
12	3140	0.2646	0.1293	0.0485	0.2442	0.1782
13	4373	0.2455	0.0979	0.1043	0.3160	0.1745
14	5175	0.2501	0.1293	0.0077	0.3623	0.0680
15	6568	0.2413	0.1046	0.1463	0.1751	0.1914
16	6765	0.2464	0.1058	0.0504	0.3159	0.0521
17	6841	0.2421	0.1065	0.0328	0.2800	0.0021
Average	5344	0.2521	0.1193	0.0885	0.2557	0.1238

Table 3. The Teletubby dataset used in Experiment 2 and the 3DGiCs of the three reconstructed surface models (listed in the columns of Model 1, Model 2 and Model 3). ARE: Average registration error [8]. SDRE: Standard deviation of registration errors [8].

truth model and the new surface model is used as the reconstructed surface model. The Foot, Skull and Hippo models have been shown in Fig. 8(a)(d) and Fig. 9 respectively. The Stanford Bunny is shown in Fig. 10. Please note that in Table. 5, the number of points denotes the number of all points involved in the estimation of 3DGiC. For the Foot, Skull, Hippo and Bunny models, it is the total number of points in the ground truth model as well as the reconstructed model. For the Bird, Frog and Teletubby datasets, it is equal to the total number of points in all of the input range images and the reconstructed model.

4. Conclusions

In this paper, a novel evaluation method, 3DGiC for assessing the quality of multiview surface reconstruction algorithms is proposed. We first develop the two transformed curvatures which describe the local surface geometry and then compute a joint probability distribution to obtain a global statistic of the surface. The global statistic is then converted into a Lorenz surface through reordering in order to generate an intuitive representation. Eventually, the 3DGiC is calculated based on the volumes under the Lorenz surfaces. For multiview surface reconstruction, we usually take the mean of a collection of 3DGiCs as the final evaluation outcome. Therefore, different from existing distance-based evaluation methods, 3DGiC incorporates local surface geometry into its global evaluation scheme. Experiments demonstrated that (1) the evaluation using 3DGiC does not require a complete ground truth model, (2) compared to existing evaluation methods, 3DGiC reflects both global and local accuracy of a reconstruction and (3) the assessment made by 3DGiC is intuitive and consistent with human perception.

We do not claim that 3DGiC should be employed exclusively for the evaluation of reconstruction methods. In practice, we can always use different measurements to more

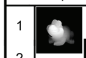
Input data sets		Number of points	ARE(*mm*)	SDRE(*mm*)	Model 1	Model 2	Model 3
1		9640	0.3036	0.1546	0.1398	0.3947	0.2888
2		8064	0.2921	0.2989	0.0932	0.3020	0.2018
3		7650	0.2837	0.1164	0.2156	0.1956	0.2030
4		8618	0.3139	0.1513	0.1207	0.4946	0.2253
5		10068	0.3138	0.1382	0.0035	0.3445	0.2102
6		10363	0.3076	0.1413	0.0923	0.2445	0.2430
7		10767	0.3128	0.1479	0.0224	0.1742	0.2126
8		10826	0.3025	0.1353	0.1440	0.3709	0.2586
9		10787	0.2883	0.1325	0.1092	0.0578	0.1997
10		10779	0.2831	0.1237	0.1236	0.0462	0.0345
11		10714	0.2814	0.1315	0.1050	0.0415	0.1448
12		10888	0.2781	0.1250	0.2678	0.0681	0.2671
13		10993	0.2813	0.1282	0.2221	0.0225	0.2653
14		11097	0.3263	0.1532	0.2051	0.2256	0.1932
15		11171	0.3270	0.1513	0.1300	0.2645	0.1867
16		10983	0.3097	0.1428	0.0375	0.4107	0.1110
17		10689	0.3108	0.1374	0.2222	0.2120	0.1896
Average		10241	0.3009	0.1476	0.1326	0.2276	0.2021

Table 4. The Frog dataset in Experiment 2 and the resultant 3DGiCs of the three reconstructed surface models (listed in the columns of Model 1, Model 2 and Model 3). ARE: Average registration error [8]. SDRE: Standard deviation of registration errors [8].

Test Data	Number of Points	Time (sec.) N_1=N_2=25	Time (sec.) N_1=N_2=50	Time (sec.) N_1=N_2=100	Time (sec.) N_1=N_2=200
Foot	20020	12	12	13	15
Skull	40004	36	37	39	42
Hippo	46210	49	50	53	56
Bunny	71894	112	116	119	128
Bird	172261	614	625	628	645
Frog	192518	717	718	721	726
Teletubby	101864	326	338	348	358

Table 5. The computational time of estimating the 3DGiCs using various datasets and quantisation parameters

comprehensively evaluate a reconstruction. Also, in this work, we show that the proposed transformed curvatures are better than principal curvatures. Nevertheless, in experiments, we still find its drawbacks. It seems over-sensitive to global noise. As a result, the output 3DGiC is somewhat unproportionally large if the reconstructed surface suffers from some global noise. Future work will thus focus on testing different descriptors of local surface geometry.

Acknowledgments

Ran Song is supported by HEFCW/WAG on the RIVIC project. This support is gratefully acknowledged.

References

[1] http://www.cgal.org/Manual/3.5/doc_html/cgal_manual/Surface_reconstruction_points_3/Chapter_main.html, CGAL.

[2] M. Berger, J. Levine, L. Nonato, G. Taubin, and C. Silva. An end-to-end framework for evaluating surface reconstruction. SCI Technical Report UUSCI-2011-001, SCI Institute, University of Utah, 2011.

[3] C. Dorai and G. Wang. Registration and integration of multiple object views for 3d model construction. *PAMI*, 20:83–89, 1998.

[4] P. Flynn and R. Campbell. A www-accessible 3d image and model database for computer vision research. *Empirical Evaluation Methods in Computer Vision*, pages 148–154, 1998.

[5] Q. Huang, B. Adams, and M. Wand. Bayesian surface reconstruction via iterative scan alignment to an optimized prototype. In *Proceedings of the fifth Eurographics symposium on Geometry processing, 2007*, 2007.

[6] H. Jin, S. Soatto, and A. Yezzi. Multi-view stereo reconstruction of dense shape and complex appearance. *IJCV*, 63(3):175–189, 2005.

[7] Y. Liu. Automatic 3d free form shape matching using the graduated assignment algorithm. *Pattern Recognition*, 38(10):1615–1631, 2005.

[8] Y. Liu. Automatic range image registration in the markov chain. *PAMI*, 32(1):12–29, 2010.

[9] M. Lorenz. Methods of measuring the concentration of wealth. *Publications of the American Statistical Association*, 9(70):209–219, 1905.

[10] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin. Shape distributions. *ACM Transactions on Graphics (TOG)*, 21(4):807–832, 2002.

[11] R. Paulsen, J. Bærentzen, and R. Larsen. Markov random field surface reconstruction. *IEEE Trans. Visual. Comput. Graph.*, 16(4):636–646, 2010.

[12] J. Pons, R. Keriven, and O. Faugeras. Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score. *IJCV*, 72(2):179–193, 2007.

[13] S. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proc. CVPR 2006*, 2006.

[14] R. Song, Y. Liu, R. R. Martin, and P. L. Rosin. Higher order crf for surface reconstruction from multi-view data sets. In *Proc. 3DIM/3DPVT 2011*, 2011.

[15] Y. Sun, J. Paik, A. Koschan, and M. Abidi. Surface modeling using multi-view range and color images. *Int. J. Comput. Aided Eng.*, 10:137–50, 2003.

[16] G. Taubin. Estimating the tensor of curvature of a surface from a polyhedral approximation. In *Proc. ICCV 1995*, 1995.

[17] G. Vogiatzis, P. Torr, S. Seitz, and R. Cipolla. Reconstructing relief surfaces. *Image and Vision Computing*, 26(3):397–404, 2008.

[18] Y. Yemez and C. Wetherilt. A volumetric fusion techinque for surface reconstruction from silhouette and range data. *Comput. Vision Image Understanding*, 105:30–41, 2007.

[19] H. Zhou and Y. Liu. Accurate integration of multi-view range images using k-means clustering. *Pattern Recognition*, 41(1):152–175, 2008.

[20] H. Zhou, Y. Liu, L. Li, and B. Wei. A clustering approach to free form surface reconstruction from multi-view range images. *Image and Vision Computing*, 27(6):725–747, 2009.