Content-Aware Geometry Image Resizing

Shu-Fan Wang Yi-Ling Chen
Chen-Kuo Chiang Shang-Hong Lai
National Tsing Hua University
Taiwan, R.O.C.
Email: {shufan,yilin,ckciang,lai}@cs.nthu.edu.tw

Bing-Yu Chen
National Taiwan University
Taiwan, R.O.C.
Email: robin@ntu.edu.tw

Tomoyuki Nishita
The University of Tokyo
Email: nis@is.s.u-tokyo.ac.jp

***Abstract*—Polygonal meshes are widely used to represent the shape of 3D objects and the generation of multi-resolution models has been a significant research topic in computer graphics. In this paper, we demonstrate how to generate multi-resolution models through 2D image processing techniques. The goal of generating multi-resolution models is accomplished by resizing the corresponding geometry images of 3D models. By defining appropriate energy on 2D images reflecting the importance of 3D vertices, we propose a modified content-aware image resizing algorithm suitable for geometry images, which achieves the preservation of salient structures and features in 3D models as well. We evaluate various image resizing techniques and show experimental results to validate the effectiveness of the proposed algorithm.**

I. Introduction

Polygonal meshes have been the most common representation for 3D geometric models and widely used in various applications in computer graphics. With the advance in modern 3D scanning technology, it has been made possible to create and store massive polygonal models with relative ease (e.g. [8]). The rapidly increasing scale of data sets challenges the subsequent processing tasks, e.g. reconstruction or rendering. However, the over-sampled 3D data are not always necessary in every application. As a result, to create multi-resolution models have been an essential research topic for geometric modeling and computing.

Through the past few decades, we have seen significant advance in the development of surface or mesh simplification algorithms. The goal of surface simplification is to obtain a model of reduced complexity while maintaining a good approximation to the original model. The existing methods mainly exploits local operators, e.g. edge contraction or vertex clustering, to incrementally alter the current 3D models and local optimization is carried out to reduce the error introduced by the local operations, such as by minimizing the *quadric error metric* (QEM) [3], [4]. Global optimization technique has also been utilized to simplify 3D meshes [6] but consequently it also involves in solving a computationally difficult problem.

In [5], Gu et al. developed the *geometry images* which represent 3D surfaces by storing vertex coordinates as image pixel values. This research was firstly motivated by the intuition that geometry images of different resolutions actually correspond to 3D models of different sizes. As a result, we aim to explore the possibility of establishing a link between the traditional 3D problem, i.e. surface simplification, and 2D image processing

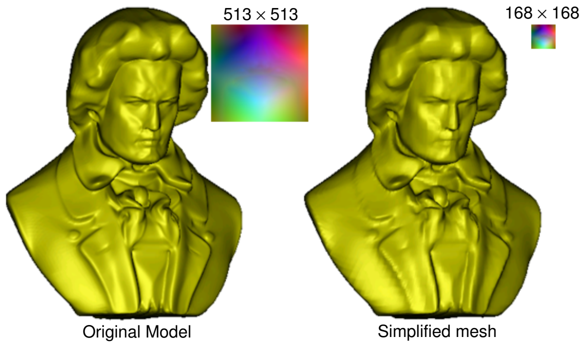

Fig. 1. Multi-resolution models by resizing geometry images. Left is the original 3D mesh with corresponding geometry image of size 513×513. The simplified 3D surface obtained by geometry image resizing to the size of 168×168 is shown in the right side.

techniques. On one hand, there are already a rich variety of methods readily to be applied to analyze and adjust 2D images. On the other hand, working in 2D domain instead of 3D may brings benefits, such as simplicity in computation or optimization. Similar ideas can be found in some previous works, shape matching [7] and surface completion [9]. In this paper, various image resizing techniques are examined to evaluate their applicability for resizing geometry images. Particularly, the *content-aware image resizing* (CAIR) methods [1], [14] are paid more attention for the purpose of preserving the important structures and features during surface simplification.

The rest of this paper is organized as follows. In Section II, we examine the surface simplification problem by resizing geometry images and introduce the proposed algorithm most suitable for this task. We evaluate various image resizing techniques and compare with the proposed method by showing experimental results on several 3D models in Section III. Section IV concludes this paper.

II. Surface Simplification by Resizing Geometry Images

Although this work was firstly motivated to solve surface simplification problem by image resizing techniques and the ideal scenario is to make as least modification on the 2D techniques as possible, the surface representation (i.e. geometry images) to be dealt with is intrinsically different from natural images after all. For example, it is important to adjust the

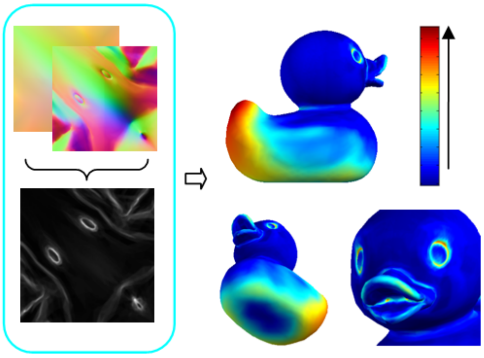

Fig. 2. High gradient energy in geometry image and normal map can well represent the features of a 3D object.

contents of a geometry images I_{gim} without destroying the topology of the corresponding 3D model, which is implicitly encoded in I_{gim}. As a result, in order to design an effective simplification method based on geometry images, several considerations are summarized as below.

*a) **Cost Definition**::* Typically, a simplification algorithm defines a certain measure of cost to guide the simplification process, e.g. QEM [3]. Similarly, to achieve feature preservation, it will be necessary to have some saliency measure reflecting the importance of 3D vertices.

*b) **Simplification**::* With appropriate cost definition, we still need a mechanism to drive the simplification process. In this work, it is attempted to resort to a certain optimization technique that determines which pixels (i.e. mesh vertices) to be retained or eliminated. Again, it is essential to maintain the structure and topology of the original 3D models.

*c) **Reconstruction**::* To alter the geometry of 3D models unavoidably introduces errors on the simplified models. It is thus significant to compute a good position for the vertices that survive during the simplification process. In the case of image resizing, to blend several pixels to form a new one may produce good results for real images, but is not necessarily good in terms of the quality of simplified meshes.

A. Saliency Measure

The key to CAIR methods [1], [14], [13], [10], [11] is to change image sizes by eliminating redundant pixels while retaining the important ones. However, it is not always easy to define an appropriate *saliency* measure for all types of natural images. Take Fig. 3(b) as an example, simple gradient energy does not always reflect the important regions in an image for the case of highly complex and textured background. When considering the simplification of 3D meshes, it is usually preferred that the redundant vertices (e.g. with higher sampling density) will be removed with higher priority and the feature points (e.g. with sharp orientation change) tend to be retained. Recall that geometry images are 2D arrays storing various surface properties, such as 3D coordinates and normal vectors, as the RGB values. Therefore, the gradients of a geometry image and a normal map actually indicate the local sampling

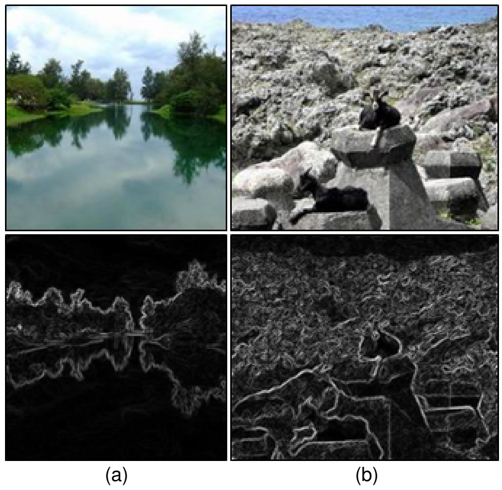

Fig. 3. High gradient energy does not always correspond to the important area in a natural image.

density of 3D vertices and shape variation of the corresponding 3D model. Despite the simplicity, the weighted combination of gradient energy derived from the geometry image and the normal map well represents the saliency map of a 3D model:

$$E = \alpha E_{gim} + (1-\alpha) E_{nmp}, \tag{1}$$

$$\begin{aligned} E_{gim} &= \sum_{c\in\{R,G,B\}} \sqrt{(\tfrac{\partial}{\partial x} I^c_{gim})^2 + (\tfrac{\partial}{\partial y} I^c_{gim})^2}, \\ E_{nmp} &= \sum_{c\in\{R,G,B\}} \sqrt{(\tfrac{\partial}{\partial x} I^c_{nmp})^2 + (\tfrac{\partial}{\partial y} I^c_{nmp})^2}, \end{aligned} \tag{2}$$

where E_{gim} and E_{nmp} denote the accumulated gradients over the R, G, B channels from the geometry image I_{gim} and the normal map I_{nmp}, respectively. Fig. 2 demonstrates an example of the saliency map corresponding to a 3D model, where the importance features can also be visually inspected.

B. Proposed Resizing Algorithm

In this paper, we propose to adopt the warping-based image retargeting algorithm originally developed in [14] for geometry image resizing. The problem can be formulated as solving a constrained linear system so as to recover the new position $(x_{i,j}, y_{i,j})$ of each pixel (i,j) under three types of constraints. Take the calculation of horizontal displacement for instance. First, each pixel is assumed to be at a fixed distance from its left and right neighbors: $x_{i,j} - x_{i-1,j} = 1$ and $x_{i+1,j} - x_{i,j} = 1$. The second constraint is to map each pixel to a location similar to the one of its upper and lower neighbors: $x_{i,j} - x_{i,j+1} = 0$. The third constraint fits the warped image to the dimensions of the target image size: $x_{1,j} = 1$ and $x_{W,j} = W_{target}$, where W and W_{target} denote the width of the original and resized image, respectively.

Given the saliency map of I_{gim} and I_{nmp}, an important pixel is preferred to be warped to a new position occupied by only itself while less important ones can be safely blended

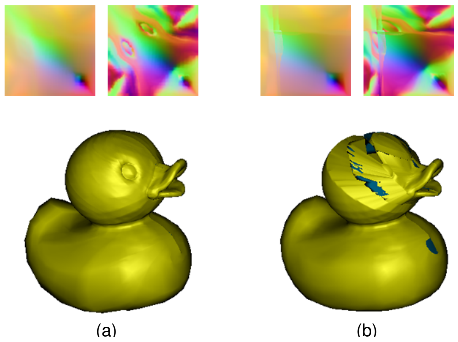

Fig. 4. Resize a geometry image from 500×500 to 105×105 by using (a) progressive resizing and (b) direct resizing [14].

with other unimportant neighbors. Therefore, the constraints should be weighted by the corresponding energy value:

$$\begin{array}{l} E_{i,j}(x_{i,j} - x_{i-1,j}) = E_{i,j}, \\ E_{i,j}(x_{i+1,j} - x_{i,j}) = E_{i,j}, \\ E_{i,j}(x_{i,j} - x_{i,j+1}) = 0. \end{array} \quad (3)$$

All the equations form an over-determined constrained sparse linear system. The optimal positions of the warped pixels can be obtained by minimizing the error of the above equations, which is equivalent to finding the least-squares solution of the sparse linear system:

$$Ax \approx b \Rightarrow x = (A^T A)^{-1} A^T b. \quad (4)$$

Similarly, the coordinate variables $y_{i,j}$ of pixels (i,j) can also be obtained from the least-square solution of following equations:

$$\begin{array}{l} E_{i,j}(y_{i,j} - y_{i,j-1}) = E_{i,j}, \\ E_{i,j}(y_{i,j+1} - y_{i,j}) = E_{i,j}, \\ E_{i,j}(y_{i,j} - y_{i+1,j}) = 0, \\ y_{i,1} = 1, \\ y_{i,H} = H_{target}. \end{array} \quad (5)$$

There are two considerations for applying this warping-based method to geometry image resizing. Firstly, we take a progressive strategy to gradually resize the geometry image to the target size instead of direct resizing. Progressive resizing avoids a mass of pixels of low energy to be over-decimated. Besides, progressive resizing also alleviates the *self-intersection* phenomenon produced by [14], which means the warped pixels are not guaranteed to be positioned in the same order horizontally and vertically as in the original image. Since geometry images implicitly encode the connectivity of 3D vertices, this phenomenon results in noticeable artifacts, such as the kinks of triangular faces as shown in Fig. 4(b). Secondly, we perform bi-cubic interpolation to reconstruct appropriate pixel values after warping. In [14], the pixels mapped to the same position are averaged to obtain the new pixel value. Empirically, we found this blending strategy causes the 3D model to shrink, which becomes serious in our progressive resizing scheme.

In each resizing step, we adaptively decide the image size of the next step by the energy distribution. Note that I_{gim} is square and the saliency map ranges in $[0,1]$. The width and height in the k-th step can be adaptively decided by:

$$W_k = H_k = \max(\sqrt{|\{(i,j)|E^{k-1}(i,j) > \lambda\}|}, W_{target}), \quad (6)$$

where the E^{k-1} is the saliency map obtained from the intermediate images of the $(k-1)$-th step and λ is a thresholding parameter controlling the step size, which is 0.1 in our default setting. Briefly speaking, Equation (6) counts the number of pixels with energy higher than λ to decide the dimensions of resized images. To summarize, the main steps of the proposed simplification algorithm are given as below:

1) Compute the gradient energy map. (Section II-A)
2) Determine the target image size $(W_i,\ H_i)$. (Eq. 6)
3) Warp I_{gim} and I_{nmp} to the size of $(W_i,\ H_i)$.
4) If $W_i = W_{target}$ and $H_i = H_{target}$, stop. Else go to Step 1.

III. Experimental Results

In this section, we evaluate the performance of various image resizing techniques and compare the results with the proposed method. For validation, we have adopted the METRO geometric comparison tool [2] to evaluate the deviation of 3D meshes corresponding to the original and resized geometry images. The energy maps of different methods [1], [14] and the proposed method are computed by Equation (1).

A. Evaluation of Various Image Resizing Methods

We applied the proposed image resizing method to obtained simplified 3D models of various resolutions, as shown in Fig. 5 and **??**. It is interesting to note that in [5] Gu et al. applied JPEG encoding to *compress* geometry images, which is essentially different from resizing geometry images. Compression on geometry images does not alter the complexity of the corresponding 3D models, but inevitably introduces loss in accuracy of the mesh vertices. In Fig. 5, the number of triangles of the original 3D meshes (524288) were reduced to roughly one half of the original size (278258) without causing any noticeable visual differences. As discussed in Section II, an appropriate method which best reconstructs the pixel values is desirable when performing image resizing. We use three commonly used data interpolation methods (say, *bi-cubic*, *bi-linear* and *nearest-neighbor* interpolation) to generate multi-resolution 3D models of BUNNY and evaluate the deviations with the original model by using METRO. As shown in Fig. 6, bi-cubic interpolation produces best results in terms of least errors measured on the models of lower resolutions. Consistent results were obtained by resizing other 3D models and thus we adopt bi-cubic interpolation in the rest experiments when needed.

For comparison, we applied different image resizing methods, include regular down-sampling (RDS), seam carving (SC) and the technique propose in [14] (VR), to generate

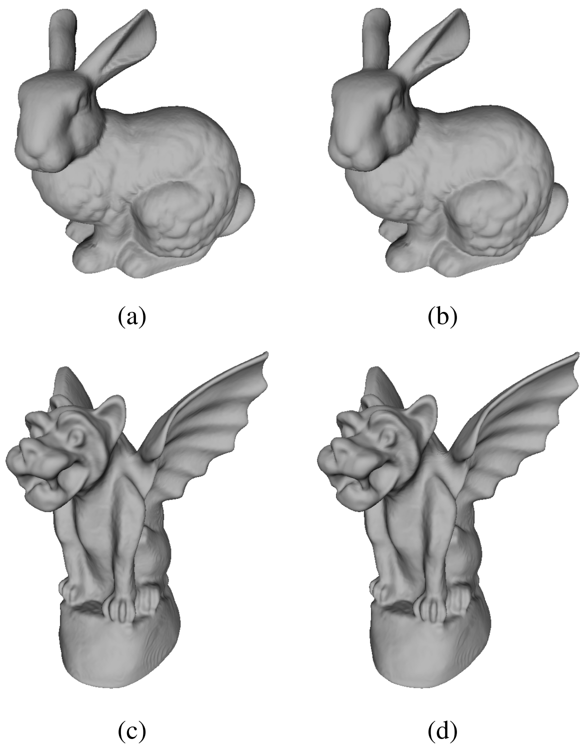

Fig. 5. Multi-resolution models generated by geometry image resizing. (a)(c) 512×512, (b)(d) 374×374.

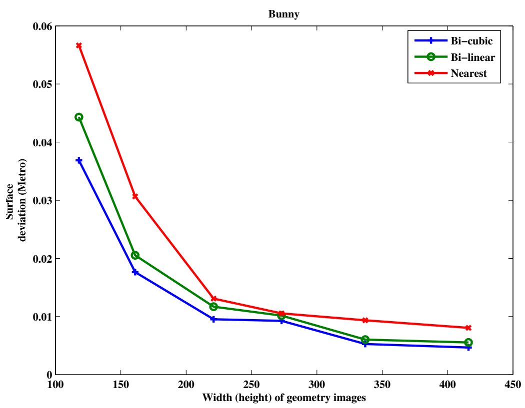

Fig. 6. Comparison of various interpolation methods.

multi-resolution models to evaluate their performance. Fig. 7 demonstrates an example of aggressively resizing geometry images of original size 500×500 to 105×105. As shown in Fig. 7, although RDS can roughly preserve the global structures of the original models, the important features were not well preserved. Directly resizing an geometry image (VR) produces undesirable results because the pixels of low energy are over-decimated, which greatly alters the structure of the original model. Seam carving works by iteratively removing seams of least cost horizontally and vertically. Although important pixels tend to be retained, there is no mechanism to prevent the regions of low energy from being repeatedly carved out. Moreover, the connectivity between pixels (3D vertices)

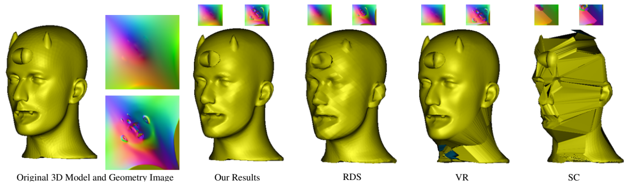

Fig. 7. Surface simplification of two example models by resizing the corresponding geometry images to the target sizes of 105×105.

becomes more and more irregular after a large number of seams are removed. The proposed method outperforms other methods in content-aware geometry image resizing.

IV. Conclusion

In this paper, we presented an improved content-aware image resizing method suitable for geometry images. Generating multi-resolution 3D models by 2D representation and processing techniques brings some benefits because geometry images are more compact representations for 3D models, and easier to render, transmit and store than traditional polygonal meshes. In addition, most 2D image processing techniques are easy to be accelerated by GPUs, which will be part of our future work. It will also be interesting to explore the feasibility of applying the proposed method to *multi-chart* geometry images [12] which improves the uneven regular resampling on surfaces.

References

[1] Avidan, S., Shamir, A.: Seam carving for content-aware image resizing. ACM SIGGRAPH **26**(3) (2007)
[2] Cignoni, P., Rocchini, C., Scopigno, R.: Metro: measuring error on simplified surfaces. Computer Graphics Forum **17**(2), 167–174 (1998)
[3] Garland, M., Heckbert, P.S.: Surface simplification using quadric error metrics. In: ACM SIGGRAPH, pp. 209–216 (1997)
[4] Garland, M., Heckbert, P.S.: Simplifying surfaces with color and texture using quadric error metrics. In: Proceedings of IEEE Visualization, pp. 263–269 (1998)
[5] Gu, X., Gortler, S., Hoppe, H.: Geometry images. In: ACM SIGGRAPH, pp. 355–361 (2002)
[6] Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., Stuetzle, W.: Mesh optimization. In: ACM SIGGRAPH, pp. 19–26 (1993)
[7] Laga, H., Takahashi, H., Nakajima, M.: Spherical parameterization and geometry image-based 3d shape similarity estimation. The Visual Computer **22**(5), 324–331 (2006)
[8] Levoy, M., Pulli, K., Curless, B., Rusinkiewicz, S., Koller, D., Pereira, L., Ginzton, M., Anderson, S., Davis, J., Ginsberg, J., Shade, J., Fulk, D.: The digital michelangelo project: 3d scanning of large statues. In: ACM SIGGRAPH, pp. 131–144 (2000)
[9] Nguyen, M.X., Yuan, X., Chen, B.: Geometry completion and detail generation by texture synthesis. The Visual Computer (Proceedings of Pacific Graphics '05) **21**(9–10), 669–678 (2005)
[10] Rubinstein, M., Shamir, A., Avidan, S.: Improved seam carving for video retargeting. ACM SIGGRAPH **27**(3) (2008)
[11] Rubinstein, M., Shamir, A., Avidan, S.: Multi-operator media retargeting. ACM SIGGRAPH **28**(3) (2009)
[12] Sander, P.V., Wood, Z.J., Gortler, S.J., Snyder, J., Hoppe, H.: Multi-chart geometry images. In: ACM SIGGRAPH symposium on Geometry processing, pp. 146–155 (2003)
[13] Wang, Y.S., Tai, C.L., Sorkine, O., Lee, T.Y.: Optimized scale-and-stretch for image resizing. ACM SIGGRAPH Asia **27**(5) (2008)
[14] Wolf, L., Guttmann, M., Cohen-Or, D.: Non-homogeneous content-driven video-retargeting. In: ICCV, pp. 1–6 (2007)