FAST WAY TO CREATE SEAM BOUNDARY FOR SQUARE PARAMETERIZATION WITH LOW-DISTORTION

Anuwat Dechvijankit, Hiroshi Nagahashi and Kota Aoki
Department of Computational Intelligence and Systems Science, Tokyo Institute of Technology, Tokyo, Japan

Keywords: Mesh Parameterization, Seam Cutting.

Abstract: In order to parameterize any three-dimensional surface like a closed boundary one, we need to convert its polygonal mesh into a disk topology surface. For the quality of texturing or re-meshing that uses a parameterization technique, it is more effective if the distortion of two-dimensional manifold planar domain map is as small as possible. We introduce a fast way to create seams along the three-dimensional surface for square boundary of a two-dimensional planar domain map. This paper describes a faster and less error technique than the existing seam cutting methods by simply observing the location of high distortion area in the planar domain. The distortion rate of our proposed method is low as the original one but with less time consuming.

1 INTRODUCTION

Mesh parameterization is defined as a mapping between a 3D manifold surface and a suitable target domain. To enable the mapping of a 3D surface into a 2D planar domain, mesh parameterization requires the 3D surface to be topologically equivalent to a disk without any hole, which can be easily mapped. Consequently, closed-boundary surfaces such as 3D models cannot be parameterized directly. Moreover, surfaces having any holes or being non-genus zero also cannot be parameterized directly too. To solve the above problem, we need to convert a 3D surface into disk topology by cutting seams into the surface to generate a boundary for parameterization. Parameterization between these two domains causes distortion errors such as stretch. Additionally, too short/long seam boundary or random seam without any strategy may give poor results of distortion easily.

We enhance the original method found in geometry images (Gu et al., 2002) by following the strategy of cutting seams through high curvature areas (e.g. fingers, tails, ear) and connecting them together with the shortest path. This strategy can improve distortion error when the parameterization is made onto a square planar domain. The difference from the original one is that we bypass unnecessary time consuming parameterization at the initial state and directly do unit square parameterization when all appropriate seams have been found at one time.

2 RELATED WORK

From a recent survey (Sheffer et al., 2006), there are several methods for seam cutting that have been proposed. (Gu et al., 2002), (Erickson and Har-Peled, 2002) and (Ni et al., 2004) dealt with genus-reducing, while (Lazarus et al., 2001) extracted canonical schema and (Sheffer and Hart, 2002) found high Gaussian curvature on the surface.

Dealing with high-genus models, (Erickson and Har-Peled, 2002) has proposed a cutting method which has some elegant theoretical guarantees but is complex to implement. They find the shortest loop path connecting a vertex to the vertex itself by using a front propagation technique, and then tests to see if the considering loop path reduces the surface genus or simply cut the surface into two pieces. The generation of minimal length cuts that convert a high genus surface into a topological disk is a NP-hard problem. The method used in (Erickson and Har-Peled, 2002) is a brute force approach which consumes a lot of time.

The Seamster algorithm (Sheffer and Hart, 2002) considers the differential geometry properties of the surface which are independent of a particular parameterization technique. It first finds regions of high Gaussian curvature on the mesh and then uses a minimal spanning tree of the mesh edges to connect them. Visibility of edges is used as the weight of the shortest path algorithms. When dealing with a high-genus model, (Erickson and Har-Peled, 2002) is need to create genus-reduce cutting first.

Dechvijankit A., Nagahashi H. and Aoki K..
FAST WAY TO CREATE SEAM BOUNDARY FOR SQUARE PARAMETERIZATION WITH LOW-DISTORTION.
DOI: 10.5220/0003811701850188
In *Proceedings of the International Conference on Computer Graphics Theory and Applications* (GRAPP-2012), pages 185-188
ISBN: 978-989-8565-02-0

An practical approach is proposed by (Gu et al., 2002), which traces a spanning graph of all the faces in the mesh and then prunes this graph to obtain a genus reducing cut. Then they parameterize the surface by using circular shape-preserving (Floater, 1997). They find high curvature position at the highest stretch on the mapping, and merge the boundary edges and that position with the shortest path between them. The process is terminated when the distortion of square parameterization increases.

3 OUR PROPOSED METHOD

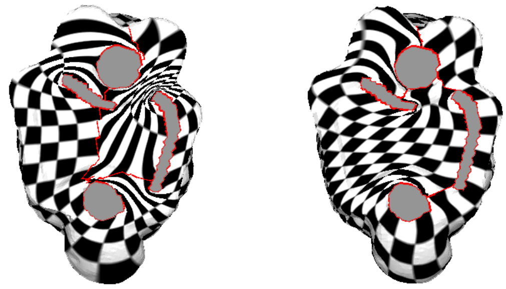
We have investigated the creation method of geometry image (Gu et al., 2002) from any kinds of meshes. A goal of the creation method is to deliver the lowest stretch of square parameterization. In practical implementation, it takes a lot of computational time during the square parameterizations to determine iteration's termination. Since geometric-stretch parameterization (Sander et al., 2001; Sander et al., 2002) has been used for stretch-minimizing method, it takes a lot of computational time. Even though we change it to a faster method, most of stretch-minimizing parameterization methods still require much solving time.

Besides the time consuming issue, we sometimes have a poor result when dealing with a mesh containing holes as shown in figure 1(a). From our investigation, it proved that the hole-connected paths generated in the phase of genus reduction cause unbalanced density of interior faces inside the boundary. This problem also affects to the iterated cut augmentation phase that fails to detect actual high curvature area because the highest stretch face will be located at boundary area instead.

We propose an enhancement of seam cutting method in geometry image (Gu et al., 2002), by improving the hole-connected paths and time consuming problems, which still maintains low stretch square parameterization result as original.

3.1 Hole-connected Paths

To avoid the problem of poor path connection among holes, we detect a boundary of the surface first in the phase of genus reducing. If one or more boundaries (holes) have been detected, we generate a cut-path by connecting these holes with shortest-path edges together. After that, we consider the hole-connected paths and existing holes as a single hole and then apply the genus reducing method in (Gu et al., 2002). See figure 1(b) for better quality stretch at boundary area after applying our approach.



(a) (b)

Figure 1: Comparison of stretch at bases of Stanford bunny. (a) is the result using original genus reducing algorithms directly; (b) is the result using our proposed algorithms by connecting holes with the shortest path first. Both cases are parameterized after selecting the best corner points that deliver lowest stretch already. The red lines indicate the boundary edges of that mesh (cut-path).

3.2 Termination Condition

The methods of geometry images (Gu et al., 2002) proposed that seam boundary should pass through various high curvature areas in order to obtain low stretch parameterization result. If we observe the properties of non-natural boundaries in state-of-art parameterizations, we can notice that too short or too long boundary edges will affect distortion in the same manner in terms of the location of the highest stretch face in planar domain.

Our approach is to ignore the comparison of L^2 stretch of square stretch-minimizing parameterization in each state as mentioned in geometry images. We directly keep detecting high-curvature areas by using shape-preserving circular parameterization iteratively. The iteration stops when highest stretch face locates very near to the boundary or boundary face itself. Also, the iteration can terminate when L^2 stretch of shape-preserving circular parameterization becomes lower than a specified threshold value.

The concept of this approach is to use the shape-preserving circular parameterization as a prediction of square parameterization. If we have too short boundary edges, we will have over-pack of interior faces that give high stretch values. Vice versa, if we have too long boundary edges, we will have over-pack of boundary faces that give high stretch at boundary faces or interior faces nearby boundary.

The stretch of circular parameterization is also important too. Since we keep extending cutting-paths during iterated cut augmentation, the number and length of boundary edges are being increased from the beginning. If the number and length of boundary edges reach the best condition, then the circular parameterization should give the lowest stretch.

Table 1: Comparison of computational time and L^2 stretch between ours and the original method. We set low stretch threshold value (for stop iteration) to 2.0. Unit square parameterization was computed by using the method (Yoshizawa et al., 2004) with random boundary positions. In the "Reason that stopped iteration in our approach" field, "A" means the highest stretch triangle was boundary area or boundary face itself; "B" means the circular stretch is lower than threshold value.

Model	genus open/closed	number of vertices/faces	time (seconds)		Stopped iteration reason in our approach	L^2 stretch	
			our	original		our	original
Mannequin head	0-open	(689/1355)	0.078	0.156	AB(1.84/2.0)	1.226	1.226
Hand	0-closed	(1002/2000)	0.34	0.982	A (2.10/2.0)	1.485	1.485
Max-Planck	0-closed	(49132/98260)	98	495	A (2.26/2.0)	1.257	1.257
Bunny	0-closed	(35947/69451)	26	36	A (2.19/2.0)	1.409	2336
Bunny(filled hole)	0-closed	(34823/69642)	51	252	B (1.88/2.0)	1.196	1.196
Cow	4-closed	(16612/33244)	20.55	21.53	A (29.7/2.0)	93009	153106
Dragon	1-closed	(50000/100000)	139	152	A (98.7/2.0)	10.94	21.68

If the shape-preserving circular parameterization can achieve a low stretch result, then stretch-minimizing square parameterization method with same boundary edges most likely delivers low stretch result too. In other words, if we stop iteration by only detecting the highest stretch at boundary area without considering stretch from circular parameterization, it might give worse square parameterization result due to the boundary constraint or too long boundary edges. Also if it gives a better result, the margin is slightly small.

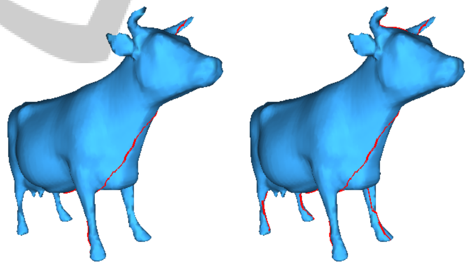
4 RESULTS

We apply our approach to various models, both open and closed meshes, genus zero and non-genus zero models. Our test system was on an Intel Core 2 Quad 2.67GHz with 4GB RAM computer.

We compare computational time between our and original approaches for finding seam-cutting path. The original approach returns seam-cutting path and square parameterization result, but our approach does not give square parameterization result. Therefore, we also do square parameterization after our approach is finished. To reduce calculation time, we use method (Yoshizawa et al., 2004) as stretch-minimizing square parameterization in both approaches instead of geometric-stretch method. Table 1 shows the result of the computational time.

In case of high genus models, the unit square parameterization results are poor because genus-reduce cutting paths are not the shortest loop as we expected. Same as hole-connected paths problem, the long boundary edges while covering small areas in 3D domain might cause unbalanced density of interior faces inside the boundary in square planar domain. If we manage to create the shortest genus-reduce cutting of these high genus models, square parameterization might give a better result. Even though we got poor genus-reduce cutting paths, our algorithm still could detect high curvature area more than the original one. See figure 2 as the results of cow model.

If seam-cutting edges pass through every high curvature areas, it is not true that it will give the best result. We extended seam-cutting edges to another high curvature area after our method was finished. The results show that they do not give a significant better result because the boundary edges are already too long since the beginning. A worse result was found in bunny model and a small margin difference was found in armadillo model. (see figures 3)

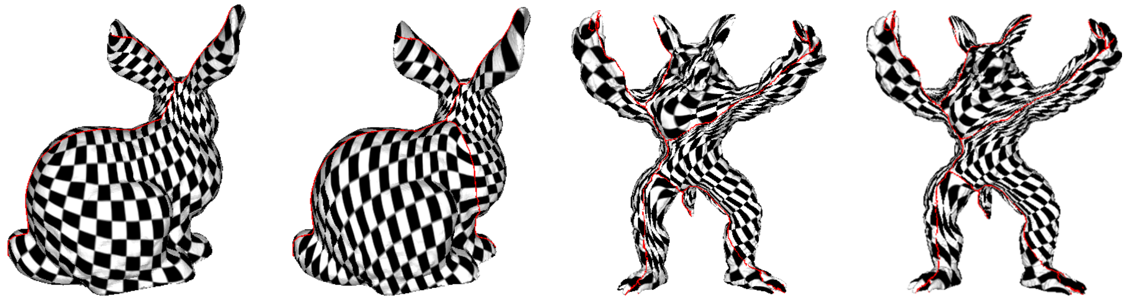


(a) Original approach. (b) Our approach.

Figure 2: Show cow models with seam cutting paths. (a) shows the result from original approach that enables to detect high curvature area only 2 areas. (b) shows our result that enables to detect high curvature area 6 areas.

5 CONCLUSIONS

We proposed an enhanced method of seam cutting method (Gu et al., 2002) in order to calculate appropriate seam boundary over a three-dimensional surface, which becomes a square boundary in planar domain. We try to convert any closed or open surface into the disk topological patch. By observing the highest stretch area and its stretch value in shape-preserving circular parameterization result, we use this information to predict the unit square parameterization result. Also the seam cutting path of the



(a) L^2 stretch: 1.194532. (b) L^2 stretch: 1.316625. (c) L^2 stretch: 1.450614. (d) L^2 stretch: 1.437329.

Figure 3: Models with seam-cutting paths and check-board texture mapping using square parameterization results. (a) shows the results using our approach which terminated cut augmentation process when stretch of circle parameterization becomes lower than threshold (2.0). (b) shows the results that manually extend seam-cutting paths to the highest stretch face in (a). (c) shows the results using our approach which terminated cut augmentation process when the highest stretch face locate at boundary area. (d) shows the results that manually extend seam-cutting paths to another high-curvature area (right-side ear). All cases are parameterized after selecting best corner points that deliver lowest stretch already. (a) has a better result than (b), (d) has slightly a better result than (c); judging from L^2 stretch value.

original method was depended on a starting face, and the obtained seam boundary that connected holes together might cause extremely under and over stretch around the boundary area. Hence, we avoided the problem by connecting them with the shortest path first before further analysis.

The issue that we are still interested in is how to find the perfect seam boundary length. When the highest stretch face is at boundary area, we think that we have over-pack faces at boundary area situation because boundary edges are too long already. We are also interested in how to assign positions in boundary that deliver the lowest stretch. Different positions give different distortion. Additionally, our method is still time consuming because of a brute force algorithm that checks almost possible positions.

ACKNOWLEDGEMENTS

We would like to gratefully thank Shin Yoshizawa for helpful advice and discussion including C++ code of (Yoshizawa et al., 2004), Hugues Hoppe for filled holes bunny and hand model data, Christian Rau who developed OpenGI library and all reviewers, The models are courtesy of the Stanford University (bunny, dragon and armadillo), the University of Washington (mannequin head), MPI für Informatik (Max-Planck), AIM@SHAPE(cow).

REFERENCES

Erickson, J. and Har-Peled, S. (2002). Optimally cutting a surface into a disk. In *Proceedings of the eighteenth annual symposium on Computational geometry*, SCG '02, pages 244–253, New York, NY, USA. ACM.

Floater, M. S. (1997). Parametrization and smooth approximation of surface triangulations. *Comput. Aided Geom. Des.*, 14:231–250.

Gu, X., Gortler, S. J., and Hoppe, H. (2002). Geometry images. *ACM Trans. Graph.*, 21:355–361.

Lazarus, F., Pocchiola, M., Vegter, G., and Verroust, A. (2001). Computing a canonical polygonal schema of an orientable triangulated surface. In *Proceedings of the seventeenth annual symposium on Computational geometry*, SCG '01, pages 80–89. ACM.

Ni, X., Garland, M., and Hart, J. C. (2004). Fair morse functions for extracting the topological structure of a surface mesh. In *ACM SIGGRAPH 2004 Papers*, SIGGRAPH '04, pages 613–622, New York, NY, USA. ACM.

Sander, P. V., Gortler, S. J., Snyder, J., and Hoppe, H. (2002). Signal-specialized parametrization. In *Proceedings of the 13th Eurographics workshop on Rendering*, EGRW '02, pages 87–98, Aire-la-Ville, Switzerland, Switzerland. Eurographics Association.

Sander, P. V., Snyder, J., Gortler, S. J., and Hoppe, H. (2001). Texture mapping progressive meshes. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '01, pages 409–416, New York, NY, USA. ACM.

Sheffer, A. and Hart, J. C. (2002). Seamster: inconspicuous low-distortion texture seam layout. In *Proceedings of the conference on Visualization '02*, VIS '02, pages 291–298, Washington, DC, USA. IEEE Computer Society.

Sheffer, A., Praun, E., and Rose, K. (2006). Mesh parameterization methods and their applications. In *Foundations and Trends in Computer Graphics and Vision*, page 2006. Now Publishers.

Yoshizawa, S., Belyaev, A., and Seidel, H.-P. (2004). A fast and simple stretch-minimizing mesh parameterization. In *SMI '04: Proceedings of the Shape Modeling International 2004*, pages 200–208, Washington, DC, USA. IEEE Computer Society.