COMPUTER-AIDED DESIGN & APPLICATIONS, 2016
VOL. 13, NO. 6, 855–862
http://dx.doi.org/10.1080/16864360.2016.1168233

Retaining circular features on deforming subdivision surface

Yiu Wai Tsoi and Kin Chuen Hui

The Chinese University of Hong Kong

ABSTRACT

In this paper, a method is proposed for retaining size and shape of circular feature on a deforming subdivision surface. An effective alternating local/global optimization method with projections is used to retain features. The multi-resolution property and the ability to generate smooth surface from arbitrary topology of subdivision surface allows user to model freeform objects. The multi-resolution editing of subdivision surface is especially useful for obtaining a smooth mesh by editing a low-complexity mesh. Constrained deformation is applied to the low complexity mesh. This speed up the computation involved, and allows the constrained deformation of complex mesh to be performed in a real-time manner. The method proposed illustrates that constrained deformation can be applied to multi-resolution model constructed with subdivision surface such that essential functional features can be retained in a shape editing process.

KEYWORDS

Subdivision surface; constrained deformation; object modeling

1. Introduction

In engineering design, designers are always facing the problem of modify existing design to fulfill new application requirements. In this process, existing object shapes may have to be deformed while special shape features are retained by including constraints in the model based on the application requirements. Retaining features while modifying regular objects to freeform objects is always a challenging problem for existing engineering CAD software. Most of the CAD software use CSG-tree and NURBS surface representation, which is effective in modeling regular engineering objects and is accurate in describing the dimension of features. However CSG tree is inefficient in representing freeform objects. NURBS surfaces are essentially four-sided, which limits user to model shapes with general topology.

Subdivision surface is a better candidate than NURBS when it comes to representing freeform objects. It is capable of generating smooth surface with irregular topology, allowing user to model freeform object efficiently. Multi-resolution editing is also supported when using subdivision surface. Editing can be performed on coarse initial mesh and propagate to fine mesh, saving time and complexity for mesh editing. With the above advantages, subdivision surface is widely used in the animation industry. However, in engineering design, subdivision surface is usually not adopted as modeling primitives. This is mainly due to two reasons: 1. Subdivision surface cannot be expressed explicitly and 2. The instability caused by extraordinary vertices. This makes subdivision surfaces difficult for modeling high precision features, limiting them to be used in conceptual design in existing CAD systems.

In this paper, we combine mesh deformation technique and subdivision surface for the purpose of retaining features while editing the general shape of the object. A local/global optimization scheme is used to solve the constrained deformation problem in order to retain the user-selected engineering feature in a deformation. On the other hand, smoothness of model and multi-resolution editing are achieved by using subdivision surface, which facilitates user to model freeform object. We also need to compensate the possible shrinkage due to the application of the approximating subdivision scheme to the model. Both the shape and dimension of the engineering features need to be maintained after deformation. The initial mesh is scaled up to compensate the shrinkage. Given the mesh of a regular object, our method can facilitate user in redesigning of general shape of the object to make it more attractive. At the same time, circular engineering features can be maintained, saving time and effort to rebuild the engineering features or redesign component to fit in the deformed shape.

CONTACT Kin Chuen Hui kchui@mae.cuhk.edu.hk

2. Background and related work

There are a large amount of techniques developed for deformation and manipulation of mesh model. We review related work on constrained deformation and provide some background information on subdivision surface.

2.1. Related work

Constrained deformation always involves non-linear optimization as many geometric properties themselves are non-linear in nature, like angle, curvature, area and volume. PriMo[2] formulated surface deformation problem based on elastically coupled rigid prisms and the problem is solved by using non-linear optimization. Eigensatz [6] preserved original shape metric by mapping the mesh from spatial domain to curvature domain. The constraints proposed by this method can be non-linear, and a non-linear optimization solver is employed. These methods took longer time to converge than linear methods and hard to guarantee if a large number of constraints are applied. A more efficient method to handle nonlinear constraints is by using local/global optimization method [4, 11, 13]. This class of method determines the correspondence of each feature between initial mesh and deformed mesh. Each vertex is projected to its corresponding constraints locally and then globally blends the projected vertices by using least square method. We use this approach on mesh deformation. The optimum transformation of each feature is determined. The transformations are then applied to features on the initial mesh correspondingly to obtain the projected vertices. After that, the optimum vertex position is determined by least square method. The other type of mesh deformation technique is based on differential coordinates [1,3,7,8]. This method preserves local properties (e.g., surface smoothness, details) in the deformation. There is also a class of method to limit the transformation of the handle vertices to explore the local modification possibility based on constraint [5,14]. However, constrained deformation is still a challenging problem. In particular, the method to enforce hard constraints is not well addressed.

2.2. Subdivision surface

Subdivision surface is a method to model a smooth surface by successively subdividing an initial mesh following the subdivision rules. The subdivision rules include vertex insertion and corner-cutting operation. Subdivision surface can be classified into two groups-approximating and interpolating schemes. In the interpolating scheme, the limit surface pass through the initial mesh. In the approximating scheme, the limit surface only approximates the initial mesh. Though both schemes can generate smooth surface, approximating scheme generate smooth surface with higher quality. Regular vertices are those vertices with valence of 4 for quadrilateral mesh or valence of 6 for triangular mesh. Otherwise, it is an extraordinary vertex. Extraordinary vertex has different surface connectivity with regular vertex [10,15]. For example, the extraordinary vertices of Loop subdivision surface are C^1-continuous, while the regular vertices are C^2-continous. We employ Loop subdivision surface [9], which is an approximating subdivision scheme applied on triangular mesh. To achieve multi-resolution editing, we subdivide the user input mesh into a fine mesh instead of getting a reduced mesh from the user input mesh. We use the original subdivision rules proposed in [9] to apply Loop subdivision surface to refine user input mesh.

3. Method overview

There are three stages in our proposed method of model editing. They are Shape editing, Features retention and Initial mesh adjustment. Our method requires user to provide a 3D mesh model as input. The target outcome is the limit surface of a subdivision surface which is a smooth surface with the desired freeform shape and retained engineering features. Fig. 1 shows the procedures of our method.

In the first stage, user need to provide a 3D mesh model and specify the engineering features to be retained under deformation. The vertex position of these features is stored. The input mesh is taken as the initial mesh of the subdivision surface. All the vertices of the input mesh are the control vertices of the subdivision surface. User edits the model by moving the control vertices. The deformation is propagated to the limit mesh. This will give a deformed initial mesh with desired shape as shown in Fig. 1b.

In the second stage, we need to retain the circular engineering features deformed during the editing process. We use an alternating local/global optimization method to retain the feature. In this stage, the isometric transformations of the features from input mesh to deformed mesh are computed and applied to input feature vertices. These vertices are the constraints in the optimization procedure. The retaining features procedure is performed on the initial mesh, shown in Fig. 1c. We obtain a smooth mesh with desired shape and retained engineering features after subdividing the feature retained mesh.

The final stage is to compensate the shrinkage of the model due to the Loop subdivision process. Radius of the circular features is compared before and after the subdivision to determine the scaling factor of each feature. The

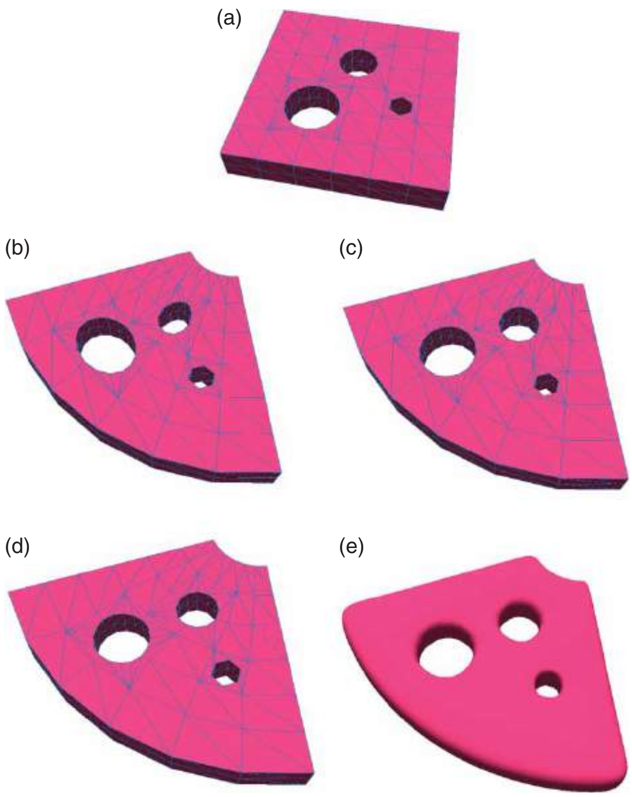

Figure 1. (a) User input mesh, (b) Deformation without constraints, (c) Deformation with constraints: Retained feature shape from deformation, (d) Scaled up of (c) to compensate shrinkage by Loop subdivision surface, (e) Limit surface with retained shape and size of the features.

scaling factors are applied to the corresponding features vertices and the initial mesh. Each scaled feature and scaled mesh is treated as the constraint in the local/global optimization problem to obtain the initial engineering feature size on the smooth limit mesh as shown in Fig. 1e.

4. Constraint deformation

We adopted a local/ global optimization framework to perform the constrained deformation. Bouaziz and colleagues [4] developed the Shape-Up method to model 3D objects with shape constraint. A projection operator is applied to the current point set involved in the constraint. A shape proximity function is formulated as the squared distance between the current point set to the projected point set. The optimal vertices position are located by minimizing the shape proximity function. In this paper, we only highlight the critical procedure of this method that we adopted. For the details of Shape-Up method, please refer to [4].

Let the column vector of all vertices to be denoted by $\mathbf{X}$. Let the set of features be denoted as $\{\mathbf{C_1}, \mathbf{C_2}, \ldots, \mathbf{C_k}\}$. For a feature $\mathbf{C_m}$, there are n vertices in the current surface involved in $\mathbf{C_m}$, denoted as $\mathbf{X_m}$. The shape proximity function for this constraint is defined as,

$$\Phi(X) = \sum_{m=1}^{k} w_m \|N_m X_m - P_m(N_m X_m)\|_2^2 \quad (1)$$

where $\mathrm{P_m}$ project $\mathbf{X_m}$ onto constraints $\mathbf{C_m}$, $\mathrm{w_m}$ is the weight of $\mathbf{C_m}$. Users are required to input the value of the weights $\mathrm{w_m}$, according to the importance of the constraint $\mathbf{C}_\mathrm{m}$. $\mathrm{N_m}$ is the mean center matrix.

$\mathrm{N_m}$ $\mathbf{X_m}$ gives the locations of the constrained vertices relative to the center of $\mathbf{X_m}$ as reference. Results from [4] show that using $\mathrm{N_m}$ can increase the convergence rate. It is defined as,

$$N_m = I_{n\times n} - (1/n)_{n\times n} \quad (2)$$

$\mathrm{P_m}(\mathbf{X_m})$ are the vertices that have the minimum squared distance to the constraint domain $\mathbf{X_c}$, and is defined as

$$P(X_m) = \mathrm{argmin}||\mathrm{Xc} - Xm||_2^2 \quad (3)$$

Combining all the projected vertices for all the constraints in the problem, equations (1) can be reformulated as,

$$E_{shape} = \|QX - P(X)\|_2^2 \quad (4)$$

The constraint relations have no changes throughout the optimization process. $\mathrm{Q^TQ}$ can be pre-factorized by Cholesky factorization. Therefore, Q is the matrix combining all weighed mean center matrix, P(X) is all the projected vertices.

We can first compute P(X) from current X; after that, we update X by using the normal equation, $\mathrm{Q^TQX = Q^TP(X)}$. Since there is no changes on the constraints relation throughout the optimization process, the following iterations are linear systems that are solved by back substitution of the updated vertices with the pre-factorized matrix.

Detail preservation can be achieved by only allowing the desired feature to perform isometric transformation while the non-preserving elements can be deformed freely. The following discussion introduces the formulation of the projective operator we applied in the above optimization framework. As we stated in Section 3, the target outcome is the smooth limit surface of subdivision surface with retained features. Subdivision surface cannot be represented explicitly such that subdivision surface cannot be formulated as projective operator. Moreover, for the purpose of multi-resolution editing, we decided to preserve the details on the initial mesh. The specified feature vertices on the initial mesh are stored. We applied the relative shape constraints in [4] on the initial mesh. The transformation parameters between the initial feature vertices and the deformed feature vertices are obtained by least squares estimation proposed by [12]. Let V' be the mean-centered initial feature and V^* be the mean-centered deformed matrix. We compute the covariance matrix C,

$$C = (V')^T (V^*) \tag{5}$$

We then perform singular value decomposition (SVD) on the covariance matrix C,

$$C = ADB^T \tag{6}$$

The optimum rotation R between V' and V^* is computed by,

$$R = ASB^T \tag{7}$$

where

$$S = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \text{ if } \det(A)\det(B) = 1 \text{ or}$$
$$S = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \text{ if } \det(A)\det(B) = -1 \tag{8}$$

The projected vertex is defined as,

$$P(V^*) = V'R \tag{9}$$

Vertices V on the current mesh is updated by solving Eq. (4) with the above P(V). The dimension of Q is the number of projected vertices times the total number of vertices on the mesh. The dimension of PV is the number of projected vertices times 3.

It is too restrictive to adjust feature vertices position if we formulate the constrained deformation problem with shape energy E_{shape} only. Edges near the feature vertices on the deformed mesh M' are easily to get overlapped. We included closeness energy E_{close} and Laplacian energy E_{smooth} for setting boundary and smoothing the deformation. These two energies are used to penalize the shape deformation of the mesh.

$$E_{close} = \sum_{i=1}^{n} \| v_i - c(v_i) \|_2^2 \tag{10}$$

where $c(v_i)$ is the closest point of v_i on the original surface.

$$E_{smooth} = \sum_{i=1}^{n} \| \sum_{\{i,j\} \in E} w_{ij}(I_j - I_i) \|_2^2 \tag{11}$$

where E denotes the edges of the mesh, $I_i = v*_i - v'_i$ and w_{ij} is the standard cotangent weight for triangular mesh[11].The total energy for constrained deformation is the sum of the above energies,

$$E_{total} = \lambda_s E_{shape} + \lambda_c E_{close} + \lambda_m E_{smooth} \tag{12}$$

where λ_s, λ_c, λ_m are the weights of the corresponding energy. We used the weight of (10,5,1) throughout the experiment.

The above summarizes the procedure of the Shape-Up method. This method is capable of setting constraints with explicit representation. Combining this method with loop subdivision surface, circular features on the initial mesh can be retained. This features retained initial mesh is then subdivided to obtain a smooth mesh.

5. Initial mesh adjustment

We assume the user input model has the desired feature size and shape. We apply subdivision surface to provide the smoothness and multi-resolution ability of the model. We retain the desired feature shape on the limit surface of the deformed mesh by using the Shape-Up method as summarized in Section 4. If we use interpolating subdivision surface scheme, the feature size can also be maintained on the limit surface. However, the smooth surface generated by interpolating subdivision scheme is less desirable than that generated by approximating subdivision scheme. We decided to use Loop subdivision surface, one of the approximating subdivision schemes, to provide the surface smoothness. However, the inherent problem that comes with the Loop subdivision surface is that the model shrinks on the subsequent subdivision

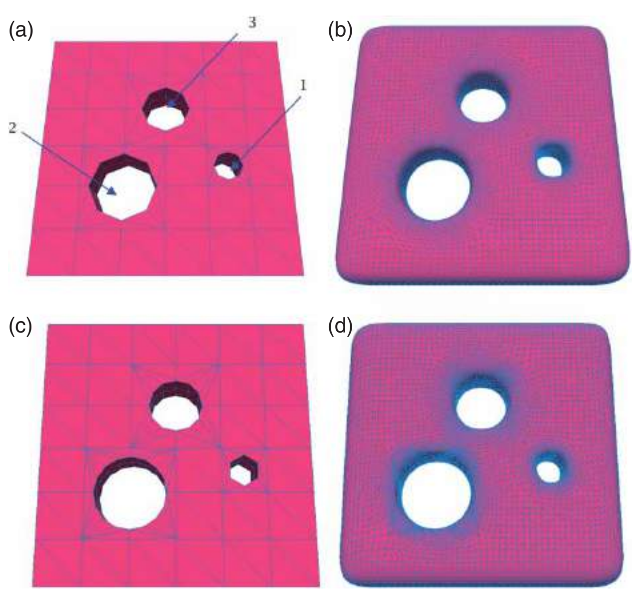

Figure 2. (a) Model A initial mesh, (b) Model A limit mesh, (c) Model B initial mesh, (d) Model B limit mesh.

levels. There is no general formulation for the shrinkage of the subdivision surface. The shrinkage depends on the initial mesh quality, the number and the position of extraordinary vertex. Hence, each model has its own shrinkage rate on the subsequent subdivision levels.

In order to retain the size of engineering features on the limit surface, the most direct way is to evaluate the shrinkage of the feature and then determine a suitable scaling factor to scale up the limit surface. However, the shrinkage of each feature is different. This depends on the extraordinary vertices near the feature. Fig. 2 and Tab. 1 shows the shrinkage difference between two models. They represent the same features while the numbers of vertices representing the feature are different. We measure the radius of the circular features on the limit surface of the two models and compare their difference.

Table 1. Comparison of shrinkage of features due to Loop subdivision surface on different meshes, (Feature 1,2,3 are indicated on F2.a).

	Initial radius	Model A radius	Change (%)	Model B radius	Change (%)
Feature 1	0.35	0.316363	9.610571	0.292578	16.40629
Feature 2	0.8	0.723115	9.610625	0.780018	2.49778
Feature 3	0.6	0.542337	9.6105	0.573624	4.396

We evaluate the shrinkage of each feature and obtain the compensating scale factor of each feature. We multiply each compensating scale factor to the whole initial mesh and features vertices on the initial mesh. The scaled initial meshes is treated as boundary energy $E_{boundry}$ and the scaled feature vertices is treated as size energy E_{size}. The problem is then solved by using the optimization framework discussed in Section 4. In this paper, we focus on circular feature. We take the radius of the circular features as the parameter to evaluate the shrinkage due to subdivision surface and determine the compensating scale factor. The user first select the feature vertices on initial mesh. We perform principal component analysis to find the cylinder axis of the user selected vertices. We first compute the covariance matrix C of the selected vertices on the initial mesh,

$$C = V_i^T V_i \tag{13}$$

We perform SVD on C. The result is shown on Eqn. (6). Each column vector of A represents the axis of the cylinder. Since the radial axes are the same, the cylinder axis is either the first column A(1) or the last column A(3) of A. The maximum point-axis distance for each axis is denoted as D_1,D_2,D_3. By comparing $|(D_3 - D_2)|$ and $|(D_1 - D_2)|$, we obtain the cylinder axis A_c. As the circular axes

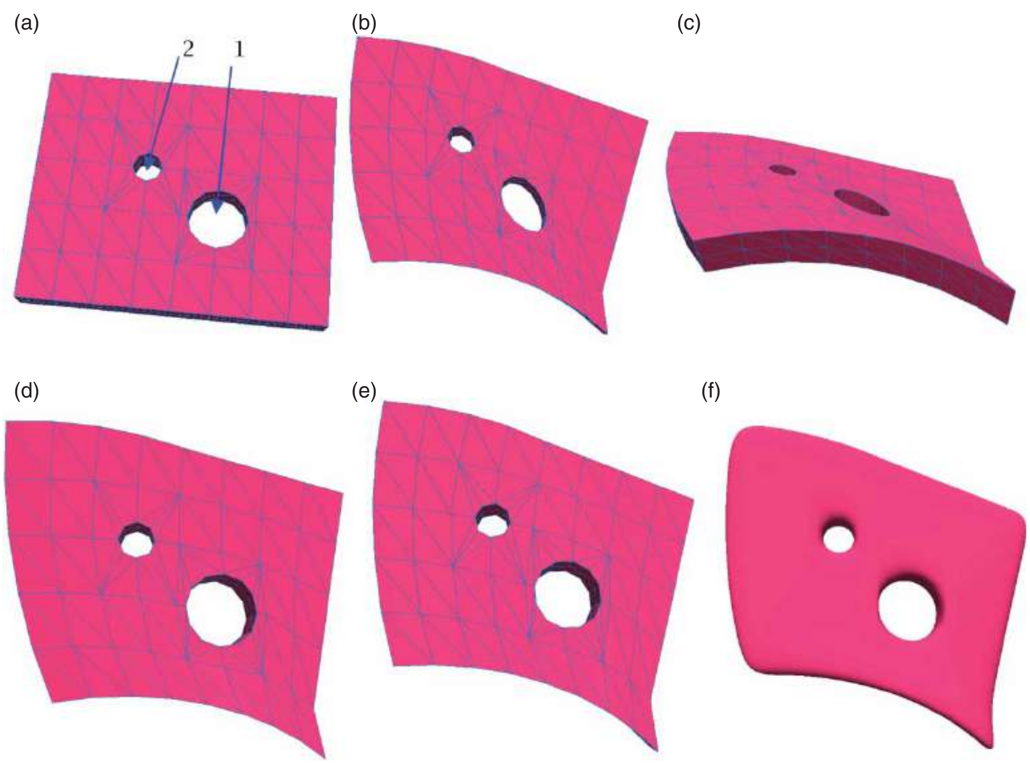

Figure 3. (a) User input mesh with feature 1 and feature 2 denoted, (b) Deformation without constraints, (c) Another view of deformed model, (d) Deformation with constraints: Retained feature shape from deformation, (e) Scaled up of (b) to compensate shrinkage by Loop subdivision surface, (f) Limit surface with retained shape and size of the features.

A_{r1} and A_{r2} are on the same plane, the difference between their maximum point-axis distances is close to zero. We denote the radius of circular feature on the initial mesh as D' and that on the limit surface as D*. D is defined as the minimum distance between V_i and A_c,

$$D = \min \| V_i - V_i(A_c A_c^T) \|_2 \tag{14}$$

The compensating scale factor SF is formulate as,

$$F = D'/D* \tag{15}$$

There is a compensating scale factor for each feature. Therefore there is a set of F $\{F_1, F_2, \ldots, F_m\}$.

Let P_b be the projection operator in $E_{boundry}$ and P_s be the projection operator in E_{size}.

$$P_b(X') = F_i X' \tag{16}$$

$$E_{boundry} = \sum_{i=1}^{m} \| X' - P_b(X') \|_2^2 \tag{17}$$

$E_{boundry}$ is used to find the general size of the mesh. Since E_{size} is used to restore the radius of the feature. the scaling factor is only applied to the radius of circular features only, but not the height of cylinder. P_s is formulated as,

$$P_s(V_i') = V_i'[I_{3x3} - (F_i - 1) * (T_{r1} + T_{r2})] \tag{18}$$

$$\text{where} \quad T_{r1} = A_{r1} A_{r1}^T \text{ and } T_{r2} = A_{r2} * A_{r2}^T \tag{19}$$

$$E_{size} = \sum_{i=1}^{m} \| V_i' - P_s(V_i') \|_2^2 \tag{20}$$

The total energy for the initial mesh adjustment is formulated as,

$$E_{adjust} = \lambda_f E_{size} + \lambda_b E_{boundry} \tag{21}$$

The weights (λ_f,λ_b) we used in the experiments are (2,1). Fig. 1 and Fig. 3 to Fig. 5 show the results of our experiment. Tables 2 to 5 summarize the result of initial mesh adjustment of Fig. 1, Fig. 3, Fig. 4 and Fig. 5 respectively.

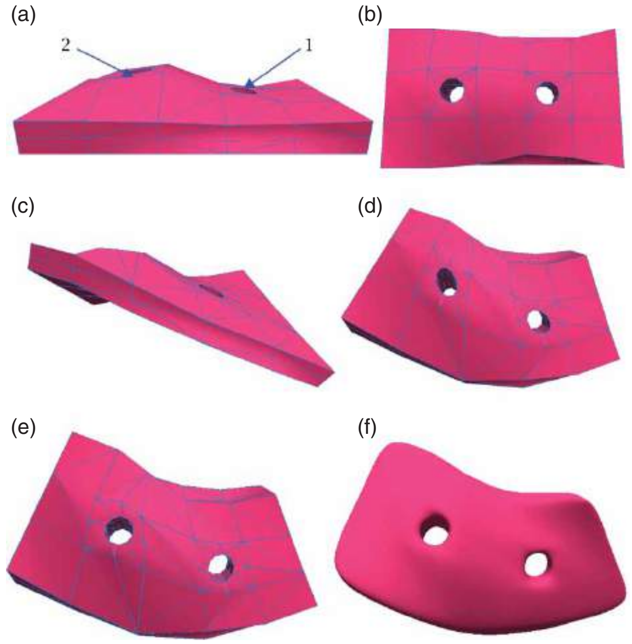

Figure 5. (a) User input mesh, (b) Another view of the input mesh, (c) Deformation without constraints, (d) Another view of the deformed mesh, (e) Retained feature shapes on size adjusted initial mesh, (f) Limit surface with retained shape and size of feature.

Table 2. Result of adjusted mesh shown in Fig. 1.

	Radius on adjusted Initial mesh	Radius on adjusted Limit mesh	Target radius
Feature 1	0.415995	0.348929	0.35
Feature 2	0.820181	0.801573	0.8
Feature 3	0.628302	0.600672	0.60

Table 3. Result of adjusted mesh shown in Fig. 3.

	Radius on adjusted Initial mesh	Radius on adjusted Limit mesh	Target radius
Feature 1	1.0358	1.00641	1
Feature 2	0.553111	0.49999	0.5

Table 4. Result of adjusted mesh shown in Fig. 4.

	Radius on adjusted Initial mesh	Radius on adjusted Limit mesh	Target radius
Feature 1	0.553177	0.50000	0.5
Feature 2	0.554165	0.50006	0.5

Table 5. Result of adjusted mesh shown in Fig. 5.

	Radius on adjusted Initial mesh	Radius on adjusted Limit mesh	Target radius
Feature 1	0.541133	0.50002	0.5
Feature 2	0.530324	0.50002	0.5

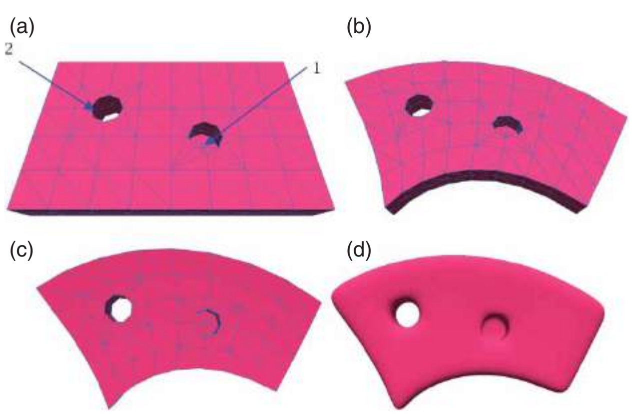

Figure 4. (a) User input mesh, (b) Deformation without constraints, (c) Retained feature shapes on size adjusted initial mesh, (d) Limit surface with retained shape and size of feature.

6. Discussion

We have implemented our method using C++ language on a i5-4570 3.20 GHz machine with 8GB RAM. We use the Cholesky solver and SVD solver with the *Eigen* library. The computational time for Fig. 1, Fig. 3 to Fig. 5 are 983 ms, 749 ms, 687 ms and 1014 ms respectively. The half-edge data structure is used for representing subdivision surface.

Results show that our method retains the size and shape of circular features on the limit mesh of the subdivision surface. Both the alternating local/global optimization with projections framework and Loop subdivision surface are easy to implement. Our method utilizes the multi-resolution essence of subdivision surface. The procedure for retaining feature is performed on the coarse input mesh, reducing computation time and effort. The initial mesh adjustment step takes the features retained coarse mesh as input instead of evaluating the compensating scale factor at the beginning and using the scale factor in the feature retaining process. This is to avoid having a scale factor influenced by the scaling effect on the model due to deformation. The objective of initial mesh adjustment step is to compensate the shrinkage of the features due to Loop subdivision surface.

Though our method can retain circular features on deforming subdivision surface, there are short comings for our proposed method. Firstly, conflicts between constraints cannot be detected. Distortion of engineering features may happen if the conflict exists. Moreover, this method does not avoid self-intersection of edges although closeness energy and Laplacian energy are included to penalize the deformation. Finally, subdivision surface may deform the model undesirably due to the extraordinary vertices on the input mesh.

For future work, we are interested in applying additional constraint types, like different element forms, or feature relations like symmetry and orthogonality, or inequality constraints like the range of distance changes allowed for boundary vertices, to be retained on subdivision surface. Moreover, we are also interested in using the optimization framework to adjust or perform remeshing on the initial mesh of subdivision surface, such that we are able to alleviate the undesired distortion on subdivision surface due to the extraordinary vertex.

7. Conclusion

We presented a framework to retain circular feature shape and size on deforming subdivision surface. Local/global optimization framework was used to retain circular feature shape on initial mesh. After that, scaling factor of each feature was evaluated base on the difference in radius of the circular features between the initial mesh and the limit mesh of a subdivision surface. The scaling factors were applied to corresponding features and the initial mesh. The scaled features and meshes were obtained by using the local/ global optimization framework to determine an optimum initial mesh that compensates the shrinkage due to Loop subdivision surface. Experimental results showed that circular features shape and size can be retained on Loop subdivision surface. In summary, constrained deformation can be combined with subdivision surface to retain circular feature shape and size in a multi-resolution manner.

Acknowledgements

The work is partially supported by a Direct Grant (No. 2050492) from the Chinese University of Hong Kong, and a Grant from the Research Grants Council of the Hong Kong Special Administration Region (Project No. 412913).

ORCID

Yiu Wai Tsoi http://orcid.org/0000-0003-4377-026X

References

[1] Alexa, M.: Differential coordinates for local mesh morphing and deformation, The Visual Computer, 19(2), 2003, 105–114.

[2] Botsch, M.; Pauly, M.; Gross, M. H.; Kobbelt, L.: PriMo: Coupled Prisms for Intuitive Surface Modeling, Proc. Eurographics/ACM SIGGRAPH Symp. Geometry Processing, 2006, 11–20.

[3] Botsch, M.; Sorkine, O.: On linear variational surface deformation methods, IEEE Transactions on Visualization and Computer Graphics, 14(1), 2008, 213–230. http://dx.doi.org/10.1109/TVCG.2007.1054

[4] Bouaziz, S.; Deuss, M.; Schwartzburg, Y.; Weise, T.; Pauly, M.: Shape-Up: Shaping Discrete Geometry with Projections, Computer Graphics Forum, 31(5), 2012, 1657–1667. http://dx.doi.org/10.1111/j.1467-8659.2012.03171.x

[5] Deng, B.; Bouaziz, S.; Deuss, M.; Zhang, J.; Schwartzburg, Y.; Pauly, M.: Exploring local modifications for constrained meshes, Computer Graphics Forum, 32(2), 2013, 11–20. http://dx.doi.org/10.1111/cgf.12021

[6] Eigensatz, M.; Pauly, M.: Positional, Metric, and Curvature Control for Constraint-Based Surface Deformation, Computer Graphics Forum, 28(2), 2009, 551–558. http://dx.doi.org/10.1111/j.1467-8659.2009.01395.x

[7] Fu, H.; Kin-Chung Au, O.; Tai, C. L.: Effective derivation of similarity transformations for implicit Laplacian mesh editing, Computer Graphics Forum, 26(1), 2007, 34–45. http://dx.doi.org/10.1111/j.1467-8659.2007.00940.x

[8] Lipman, Y.; Sorkine, O.; Levin, D.; Cohen-Or.: Linear rotation-invariant coordinates for meshes, ACM Transactions on Graphics (TOG), 24(3), 2005, 479–487. http://dx.doi.org/10.1145/1073204.1073217

[9] Loop, C.; Smooth subdivision surfaces based on triangles, Master's thesis, University of Utah, 1987.
[10] Ma, W.: Subdivision surfaces for CAD—an overview, Computer-Aided Design, 37(7), 2005, 693–709. http://dx.doi.org/10.1016/j.cad.2004.08.008
[11] Sorkine, O.; Alexa, M.: As-rigid-as-possible surface modelling, Symposium on Geometry processing, 4, 2007.
[12] Umeyama, S.: Least-squares estimation of transformation parameters between two point patterns, IEEE Transactions on pattern analysis and machine intelligence, 13(4), 1991, 376–380. http://dx.doi.org/10.1109/34.88573
[13] Vaxman, A.: A Projective Framework for Polyhedral Mesh Modelling, Computer Graphics Forum, 33(8), 2014, 121–131. http://dx.doi.org/10.1111/cgf.12405
[14] Zhao, X.; Tang, C. C.; Yang, Y. L.; Pottmann, H.; Mitra, N. J.: Intuitive design exploration of constrained meshes, Springer Vienna., 2013, 305–318
[15] Zorin, D.: Subdivision zoo, Subdivision for modeling and animation, 1999, 65–104.