

A Feature Preserved Mesh Subdivision Scheme for Biomedical Mesh

1st Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address

2nd Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address

3rd Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address

Abstract—As biomedical data in 3D spaces increasingly collected, there is pressing need for efficient and accurate applications in bioinformation analysis. For biomedical purpose, mesh subdivision techniques are commonly used to adaptively generate multi-resolution meshes for fast or accurate algorithms. Compared with current methods, the smoothing methods for each subdivision algorithm will moderate features from the original mesh. In this paper, we propose an feature focused mesh subdivision scheme, which generates a visually sensitive and a more precise result compared with commonly used subdivision methods.

Index Terms—mesh, subdivision, keypoints, feature, biomedical

I. INTRODUCTION

Recent years, 3D computer graphical technology works difusely as biomedical assistants, including utilizations of three dimensional CT (3D CT) reconstruction [1]–[4], presentations of skeletons and tissues [5], and comparison of genome [6]–[9], and etc, due to its visual obviousness. The technique is widely used for presenting facial bones and their connections, confirmation of borderline or size of diseased tissues and their relation to the adjacent tissues. With the aid of bio-images and bio-models, diagnosis and operations on clinical medicine accomplished a more accurate precision. Moreover, substantial effort has been made around world to determine spatial expression patterns of genes in mammalian genome using experimental techniques such as *in situ* hybridization(ISH) [10]. Performing ISH on multiple subjects yields expression images of various genes over the common anatomical structure and comparing these images reveals the spatial relations between genes, which are often key to understanding their functional relations [11] [12].

The pros of using Computer Graphics including 3D spatial and triangular meshes [13]–[15] to expose nicety of bioinformation has created an ever-titanic amount of spatial data (in the form of 2D medical images, and 3D bio-models) which requires efficient and accurate process and analysis consequently. The anatomical differentiations among these biomedical models aggravate the computational challenges involved in visual and medical comparison among data collected from different modes [16].

Restrained by the calculation performance of physical devices, medical detailed expression and accuracy of biomedical

data, in which the rudimentary expression are 3D meshes (triangular form, quad form, combined form), is circumstanced. Solutions emerge as medical processing require, one of which is using meshes without redundant vertices and faces. The contradiction of biomedical precision and data optimization will generate as model shrinking. Hence, feature-based mesh subdivision present start-of-the-art results [16].

Using feature-based subdivision to suitably amplify medical precision and organize spatial data into multi-resolution versions, on which visual comparison and diagnosis will have a higher performance [16]. Meanwhile, the multi-resolution structure of a subdivision mesh further gives rise to fast algorithms for processing and accurate comparison for tiny biodivergence.

Commonly used mesh subdivision methods including Butterfly method [27], Loop subdivision [25], and Catmull-Clark subdivision [26], whose experimental result will be illustrate in Fig.1. Moreover, Xie [17] provide a solid-shell element based triangular subdivision to avoid interpenetration. Liu [18] introduced a dense reconstruction algorithm for mesh subdivision. Amresh [19] developed a subdivision scheme derivate from Loop scheme and using watershed segmentation. Dirc Rose [20] proposed an adaptive process that stores the next splitting vertex and temporary triangle based on Modified Butterfly scheme. Kobbelt [21] developed refinement for both his Kobbelt scheme and newly introduced $\sqrt{3}$ subdivision. Seeger [22] introduced a subdivision scheme based on Butterfly Scheme using quark. In their method, subdivision is controlled by the faces of the original mesh and mesh features are not suitably preserved.

In this paper, we propose a keypoint-based feature preserved mesh subdivision scheme including imbalanced keypoint detection, feature preserved mesh subdivision, and model-depend wavelike noise elimination to maintain features of original meshes while give out a more adaptive mesh derivative. Subdivision result will be compared with same meshes after same time of iterations, whose visual difference is unambiguously discerned. Results show the promise of proposed method on feature maintained mesh subdivisions.

The rest of the paper is organized as follows: In Section II, we present an imbalanced keypoints detection method to locate features on meshes, and provide our basic idea of Feature

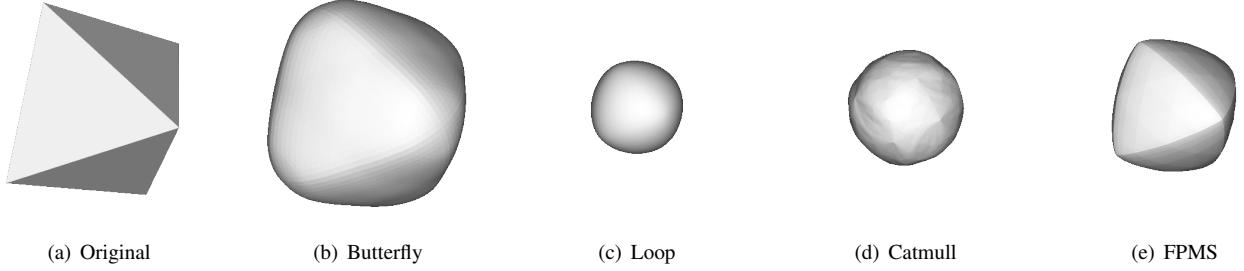


Fig. 1: a) Original Mesh. b) Butterfly Subdivision. c) Loop Subdivision. d) Catmull Subdivision. e) Feature Sensitive Mesh Subdivision. All subdivision algorithms have been applied five times on original mesh

Preserved Mesh Subdivision with its improved version to efface model-related noise wave on the mesh. Then in Section III, we combine feature detection method with our subdivision algorithm to establish Keypoint-based Feature Preserved Mesh Subdivision Scheme. Moreover, Experimental comparison and conclusion will be arranged in Section IV and Section V, respectively.

II. COUPLING MODULES OF SUBDIVISION SCHEME: FUNDAMENTAL SUBALGORITHMS

In this section, we will give an overview of algorithm modules within our proposed scheme. A brief introduction of our previous work on imbalanced-vertex based feature detection [23], extended from Li's method [24] on imbalanced point detection, will first be exhibited. We then provide our algorithm kernel, Feature Preserved Mesh Subdivision, while propose a corresponding optimization solution to eliminate model-depend wavelike noise generated from subdivision. Moreover, the kernel module follows subdivision framework composed of linear triangular face subdivision and vertex smoothing.

A. Imbalanced Keypoint Detection

We first propose a geometric feature-based vertex operator, which is a two-dimensional implementation of keypoints detection, to pick up imbalanced keypoints. We extend Li's [24] previous work on imbalanced keypoints detection to triangular meshes.

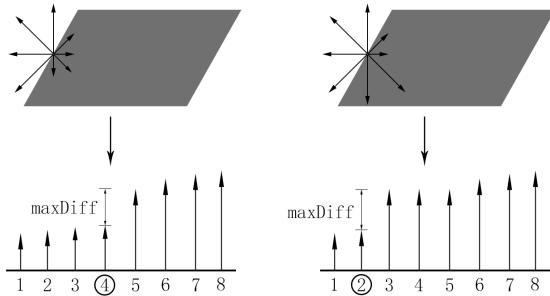


Fig. 2: Illustration of the imbalance selection. Eight arrows are sorted in term of their magnitudes. left) balanced edge point, where the index of maximum difference is 4. right) imbalanced edge point, where that is 2

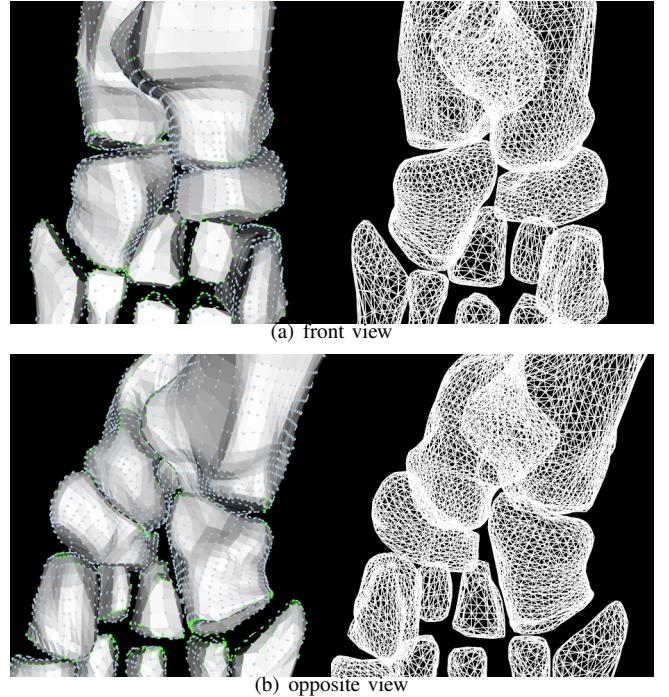


Fig. 3: Imbalanced keypoints on footbone mesh where white represent normal points and green represent imbalanced points. a) front view. b) opposite view

The basic idea of vertex operator is using projections to transform 3D geometric features to two-dimensional space. Let $M(V, F, N_F)$ be a triangular mesh where V is the set of vertices, F is the set of faces, and N_F represents the set of face normals. Suppose a projection P_T will project any vector on the plane T , 3D geometric mesh features will be transformed to two-dimension when T is the tangent plane of given mesh vertex in vertex set V . Face normals N_F will then be transformed into its projection vector set N_F^T . Finally, we settle all normalized vectors in N_F^T to a polar coordinates and calculate the set of cross angles between each vector side by side, denote as A_F .

Imbalanced point selection in 2D images aims to minimize the occurrences of edge points [24] as illustrated in Fig.2. Denote I a gray value image, p a local point, $\theta_i = \frac{(i-1)2\pi}{N}$,

and $l_i = (\cos \theta_i, \sin \theta_i)$ for $i = 1, 2, \dots, N$. Denote $\frac{\partial I}{\partial l_i}(p)$ a directional derivative of p along l_i direction. We cluster $\{\frac{\partial I}{\partial l_i}(p)\}_{i=1}^N$ into two classes in terms of their magnitudes $|\frac{\partial I}{\partial l_i}(p)|$. If two clusters have the same size, the image point p is balanced.

The sorting method proposed in Li's method is to classify $\{\frac{\partial I}{\partial l_i}(p)\}_{i=1}^N$, which can be generalized to extract 3D imbalanced vertices with proposed operator. Let \maxDiff be the max difference and D be the index of maximum difference:

$$\maxDiff = \max_j (\alpha_{j+1} - \alpha_j)$$

$$D = \arg \min_j (\alpha_{j+1} - \alpha_j)$$

where α represents value in A_F , and $1 \leq j \leq N - 1$. Given a threshold on homogeneity ε , the imbalanced vertex can be defined under the condition that $\maxDiff < \varepsilon$:

$$IMB(v_i) = \begin{cases} 1 & D_i < \frac{N}{2} \\ 0 & \text{else} \end{cases}$$

B. Feature Preserved Mesh Subdivision

Loops method [25] can be expressed as linear subdivision and an averaging scheme to approximate a spherical surface, which will efface edge and vertex features. The phenomenon is common when applying Catmull method, and Butterfly method. When the mesh is rigid, these iterations will fail as illustrated in Fig.1. To moderate the feature friction after subdivision and highlight original edge and vertex features, we provide a Feature Preserved Mesh Subdivision to generate feature adaptive results.

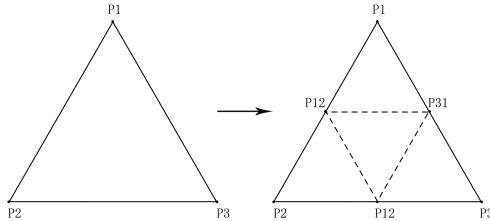


Fig. 4: Linear one to four subdivision for triangular subdivision schemes

Our proposed method implements linear one to four triangular mesh subdivision to increase the details in a mesh as illustrated in Fig.4, and smooths by proposed smoothing algorithm to accomplish surface approximation after subdivision while actualize shape retention.

Denote the original mesh as M_i , and edge point generator G_{FPMS} as illustrated in Fig.7. To perform linear triangular subdivision, we insert points created by generator G_{FPMS} on the edge of each triangle to the hash map $H_n(v_i, h_i)$ for storing vertices coordination v_i and their handles h_i as corresponding hash key. For each vertex on mesh $\{v_1, v_2, \dots, v_n\}$, check if the generated middle edge point v_k is already in the map. If so, get its handle h_k for the on-coming face generation,

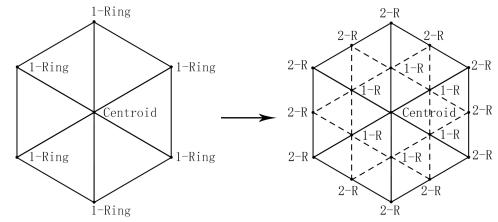


Fig. 5: One-ring vertices derivative

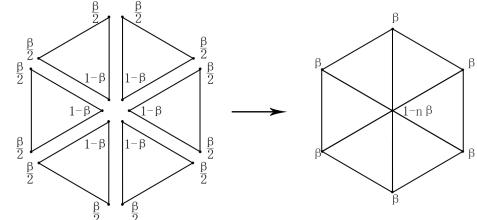


Fig. 6: One-ring neighbor weighted centroid method for triangular vertices

else, insert the point v_k into the mesh and create its handle h_k while update vertices hash map $H_{n+1}(v_i, h_i)$. Finally, form the new triangular surfaces using vertex handles geometrically anticlockwise, and eliminate elder redundant faces simultaneously. Each triangle will then be split into four sub-triangles and original mesh M_i is subdivided to M_{i+1} .

Algorithm 1 Linear Triangular Mesh Subdivision

Input : Original Mesh M_i
Output : Subdivided Mesh M_{i+1}

```
for each triangular face  $f_i$  in  $M_i$  do
    get 3 middle point on edges of  $f_i$ ;
    generate 4 subsurfaces anticlockwisely;
end for
```

Algorithm 2 Feature Preserved Mesh Subdivision

Input : Original Mesh M_i
Output : Subdivided Mesh M_{i+1}

```
for each triangular face  $f_i$  in  $M_i$  do
    get 3 edge point on edges of  $f_i$  with  $G_{FPMS}$ ;
    generate 4 subsurfaces anticlockwisely;
end for
{NOTE:  $M_i$  is now subdivided into  $M_{i+1}$ }
```



```
for each vertex  $v_i$  in  $M_{i+1}$  do
     $N(v_i)$  = 1-ring neighbors of  $v_i$  on current mesh  $M_{i+1}$ ;
     $v_i$  = 1-ring Neighbor Weighted Centroid method on  $v_i$ ;
end for
{NOTE:  $M_{i+1}$  is now smoothed}
```

Smoothing for triangular meshes will be applied on not only the previous vertices on M_i but all vertices on the generated mesh M_{i+1} whose two-ring vertex set is derived from one-ring vertices of previous mesh M_i as illustrated in Fig.5. For each vertex on M_{i+1} , we use a one-ring neighbor weighted centroid method for averaging shown in Fig.6. The weight of each neighbor β is decided by the number of one-ring neighbors n .

$$\beta = \frac{5}{8} - \left(\frac{3}{8} + \frac{1}{4} \cos \frac{2\pi}{n} \right)^2$$

C. Model-depend Wavelike Noise Elimination

By using Feature Preserved Mesh Subdivision, an edge and vertex aware result is generated. But the method will generate model-depend wavelike noise on the mesh, as illustrated in Fig.8.

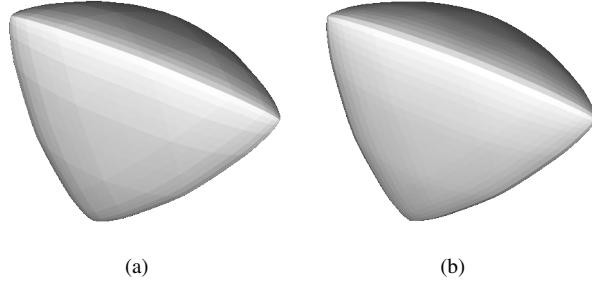


Fig. 8: a) Model-depend wavelike noise. b) Remove wavelike noise by Smoothness-focused Mesh Subdivision. All subdivisions algorithm has been applied five times on original meshes

We propose Model-depend Wavelike Noise Elimination, a combination of Feature Preserved Mesh Subdivision and Smoothness-focused Mesh Subdivision, to eliminate noise. The Subdivision scheme consists of several iterations of Feature Preserved Subdivision and an iteration of Smoothness-focused Mesh Subdivision at the end of algorithm. The implementation of Smoothness-focused Mesh Subdivision resembles the Feature Preserved Mesh Subdivision, except the generation of edge points and smoothing operator.

Unlike edge points generator G_{FPMS} in feature preserved subdivision, edge points in the last iteration of noise elimination algorithm are generated based on weighted neighbors illustrated in Fig.7, where directly connected vertices take up a greater weight, say $3/8$ each, and indirect weight slightly less, say $1/8$ each.

Neighbor vertices for smoothing method depends on previous one-ring vertices on M_i , but not newly generated vertices on M_{i+1} , which is different from the implementation of Feature Preserved Mesh Subdivision. The weight of each neighbor follows the same principle introduced in Feature Preserved Mesh Subdivision.

Algorithm 3 Model-depend Wavelike Noise Elimination

Input : Original Mesh M_i
Output : Subdivided Mesh M_{i+1}

several iterations on Feature Preserved Mesh Subdivision;
{NOTE: oncoming is Smoothness-focused iteration}

```

for each triangular face  $f_i$  in  $M_i$  do
    get 3 edge point on edges of  $f_i$  with  $G_{SMS}$ ;
    generate 4 subsurfaces anticlockwise;
end for
{NOTE:  $M_i$  is now subdivided into  $M_{i+1}$ }

for each vertex  $v_i$  in  $M_{i+1}$  do
     $N(v_i)$  = 1-ring neighbors of  $v_i$  on previous mesh  $M_i$ ;
     $v_i$  = 1-ring Neighbor Weighted Centroid method on  $v_i$ ;
end for
{NOTE:  $M_{i+1}$  is now smoothed}

```

III. KEYPOINT-BASED FEATURE PRESERVED MESH SUBDIVISION SCHEME

Based on imbalanced keypoint detection, all the vertices are classified into two categories. According to the properties of imbalanced vertices, our keypoints distributed mainly at the boundaries and detailed parts, which should be remained after subdivision but slightly adjusted with neighbors. In this section, we propose a Keypoint-based Feature Preserved Mesh Subdivision Scheme, which contains edge point generation and smooth operator, to balance the emphasis of original features and smoothing extent.

The unit of edge point generation in our provided method is a couple of triangles back to back as illustrated in Fig.7. Denote edge point generators in Feature Preserved Mesh Subdivision and Smoothness-based Mesh Subdivision as $G_{FPMS}(N)$ and $G_{SMS}(N)$ respectively, where N is the set of neighbor points.

The balance can be guided by the amount of imbalanced keypoints in an isolated generation unit and implemented by a weighted combination of $G_{FPMS}(N)$ and $G_{SMS}(N)$. For a precise and accurate calculation, the weight of two generators should be geometrically related and symmetric. The weight of $G_{FPMS}(N)$ and $G_{SMS}(N)$ will be defined as $W_{G_{FPMS}}$ and $W_{G_{SMS}}$, which depend on the number of keypoints in a generation unit, and obey the following principles.

$$W_{G_{FPMS}} = \sum_{i=1}^{|N|} Key(N_i)$$

$$W_{G_{SMS}} = |N| - \sum_{i=1}^{|N|} Key(N_i)$$

$$Key(N_i) = \begin{cases} 1 & N_i \text{ is a keypoint} \\ 0 & \text{else} \end{cases}$$

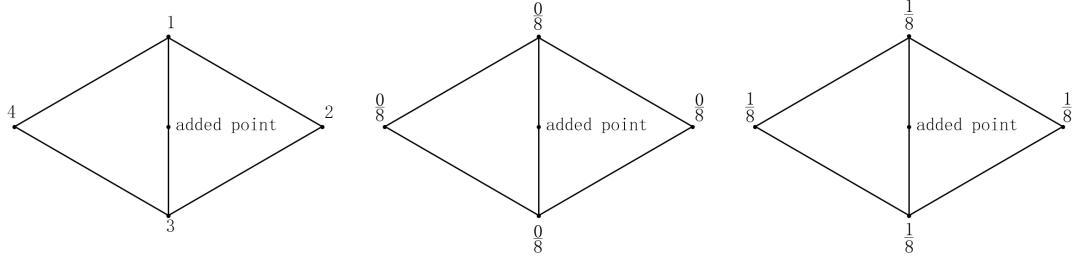


Fig. 7: Unit of edge point generator (left). Feature Preserved Mesh Subdivision edge point generator (middle). Smoothness-focused Mesh Subdivision edge point generator (right). Vertex weight of generators are tags with positions

A combination of two edge point generator $G(N)$ will be defined as:

$$G(N) = W_{G_{FPMS}} G_{FPMS}(N) + W_{G_{SMS}} G_{SMS}(N)$$

Smooth operator will also be guided by keypoints, which extends the previous smoothing scheme. Neighbor vertices for smooth operator will be different which depends on whether the centroid vertex is a keypoint or not. If so, neighbor vertices should be elder one-ring vertices on M_i , else, one-ring vertices on M_{i+1} .

Algorithm 4 Keypoint-based Sensitive Mesh Subdivision

Input : Original Mesh M_i
Output : Subdivided Mesh M_{i+1}

imbalanced keypoints detection on M_i ;
{NOTE: vertices have been classified into two class}

```

for each triangular face  $f_i$  in  $M_i$  do
    get edge point  $e_{FPMS}$  on edges of  $f_i$  with  $G_{FPMS}$ ;
    get edge point  $e_{SMS}$  on edges of  $f_i$  with  $G_{SMS}$ ;
    average position of edge points by  $G(N)$ ;
    generate 4 subsurfaces anticlockwisely;
end for
{NOTE:  $M_i$  is now subdivided into  $M_{i+1}$ }

for each vertex  $v_i$  in  $M_{i+1}$  do
    if  $v_i$  is a keypoint then
         $N(v_i) = 1\text{-ring neighbors of } v_i \text{ on previous mesh } M_i$ ;
         $v_i = 1\text{-ring Neighbor Weighted Centroid method on } v_i$ ;
    else
         $N(v_i) = 1\text{-ring neighbors of } v_i \text{ on current mesh } M_{i+1}$ ;
         $v_i = 1\text{-ring Neighbor Weighted Centroid method on } v_i$ ;
    end if
end for
{NOTE:  $M_{i+1}$  is now smoothed}

```

IV. EXPERIMENT

We test different subdivision methods on several biomedical data, including phalanx (distal phalanx, middle phalanx, and

proximal phalanx), cuneiform bone (intermediate cuneiform bones, entocuneiform, ectocuneiform), central anlrlle bone, cuboid bone, etc, provided by Florida State University.

As the time of subdivision methods iteration increased, the performance of different algorithm is becoming easier to differentiate, as illustrated in Fig.9. Except Catmull method, all the subdivision methods implement one to four triangular faces subdivision, which means each mesh in a row has the same number of vertices and faces, except Catmull subdivision. The number of vertices and triangular faces in each iteration follows the following principles when apply one to four mesh subdivision schemes, where $V(\cdot)$ and $F(\cdot)$ represent total amount of vertices and faces on a mesh respectively.

$$V(M_{i+1}) = V(M_i) + \frac{3}{2}F(M_i)$$

$$F(M_{i+1}) = 4F(M_i)$$

Obviously, subdivision result after Loop method has a better spherical resembling appearance, while Feature Preserved Subdivision provide a better property of edge feature remaining. Butterfly subdivision leads mesh to a more boundary malleable shape, while Catmull present more vertex details. Smoothness-focused Subdivision accomplish an edge feature remaining and smooth result, but compared with Keypoint-based Feature Preserved Subdivision, the later algorithm provides a better result.

Fig.10 and Fig.11 exhibit more experimental comparison on different meshes of biomedical utilization. For comparison purpose, mesh structures with only vertices and faces illustrated in 9 are abbreviated.

V. CONCLUSION

Proposed Feature Sensitive Subdivision method works well on edge features but it will generate troublesome model-shape noise waves. By using the idea in Advanced Feature Sensitive Subdivision, noise waves problem will be solved and generate a smoother result. Compared all the methods mentioned in the figures, keypoints-based Feature Sensitive Subdivision has a better feature based result.

REFERENCES

- [1] DENG J F, LI J X, LI L, et al. FPGA accelerate 3D CT reconstruction[J]. Application of Electronic Technique, 2010, 9: 030.
- [2] Koopman D, Dalen J, Arkies H, et al. A small voxel FDG-PET/CT reconstruction improves the visual evaluation of axillary lymph nodes in patients with breast cancer[J]. Journal of Nuclear Medicine, 2016, 57(supplement 2): 1502-1502.
- [3] Muthusami P, Shkumat N, Rea V, et al. CT reconstruction and MRI fusion of 3D rotational angiography in the evaluation of pediatric cerebrovascular lesions[J]. Neuroradiology, 2017: 1-9.
- [4] McCann M T, Nilchian M, Stampaponi M, et al. Fast 3D reconstruction method for differential phase contrast X-ray CT[J]. Optics express, 2016, 24(13): 14564-14581.
- [5] X. Tao, Application of 3D Reconstruction in Maxillofacial Surgery. Acta Universitatis Medicinae Tangi. 1999.
- [6] Lein E S, Hawrylycz M J, Ao N, et al. Genome-wide atlas of gene expression in the adult mouse brain[J]. Nature, 2007, 445(7124): 168.
- [7] Bonev B, Cavalli G. Organization and function of the 3D genome[J]. Nature Reviews Genetics, 2016, 17(11): 661-678.
- [8] Tjong H, Li W, Kalhor R, et al. Population-based 3D genome structure analysis reveals driving forces in spatial genome organization[J]. Proceedings of the National Academy of Sciences, 2016, 113(12): E1663-E1672.
- [9] Caspermeyer J. Principles of 3D Genome Folding and Gene Expression Studied across Species[J]. Molecular Biology and Evolution, 2017, 34(6): 1548-1548.
- [10] Carson J P, Thaller C, Eichele G. A transcriptome atlas of the mouse brain at cellular resolution[J]. Current opinion in neurobiology, 2002, 12(5): 562-565.
- [11] Ju T, Warren J, Eichele G, et al. A geometric database for gene expression data[C]//Symposium on geometry processing:[proceedings]/sponsored by the Eurographics Association in cooperation with ACM SIGGRAPH. Symposium on Geometry Processing. NIH Public Access, 2003, 2003: 166.
- [12] Schaefer S, Hakenberg J, Warren J. Smooth subdivision of tetrahedral meshes[C]//Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing. ACM, 2004: 147-154.
- [13] Brentzen J A, Abdushitov R, Singh K. Interactive shape modeling using a skeleton-mesh co-representation[J]. ACM Transactions on Graphics (TOG), 2014, 33(4): 132.
- [14] Guskov I V, Schrder P, Sweldens W. Non-uniform relaxation procedure for multiresolution mesh processing: U.S. Patent 8,830,235[P]. 2014-9-9.
- [15] Zhang Y, Ma L. A Scaling Method of Sensitive Objects Based on Loss Constraint Triangle Mesh Deformation[J]. Journal of Image and Graphics, 2015, 3(2).
- [16] Ju T, Carson J, Liu L, et al. Subdivision meshes for organizing spatial biomedical data[J]. Methods, 2010, 50(2): 70-76.
- [17] Xie Q, Sze K Y, Zhou Y X. Drape simulation using solid-shell elements and adaptive mesh subdivision[J]. Finite Elements in Analysis and Design, 2015, 106: 85-102.
- [18] Liu H H. An Improved refinement algorithm of triangular mesh subdivision based on minimum weight theory[C]//Applied Mechanics and Materials. Trans Tech Publications, 2014, 513: 2552-2555.
- [19] Amresh A, Farin G, Razdan A. Adaptive subdivision schemes for triangular meshes[J]. Hierarchical and Geometric Methods in Scientific Visualization, 2003, 13(4): 319-327.
- [20] Rose D, Kada M, Ertl T. On-the-Fly Adaptive Subdivision Terrain[C]//VMV. 2001: 87-92.
- [21] Kobbelt L. $\sqrt{3}$ -subdivision[C]//Proceedings of the 27th annual conference on Computer graphics and interactive techniques. ACM Press/Addison-Wesley Publishing Co., 2000: 103-112.
- [22] Seeger S, Hormann K, Husler G, et al. A Sub-Atomic Subdivision Approach[C]//VMV. 2001: 77-86.
- [23] K. Feng, Q. Li, Y. Gong, J. Yang, S. Peng. Detection of Imbalanced Vertices in 3D meshes. ICDH 2016.
- [24] Li Q, Ye J, Kambhamettu C. Interest point detection using imbalance oriented selection[J]. Pattern Recognition, 2008, 41(2): 672-688.
- [25] Loop C. Smooth subdivision surfaces based on triangles[J]. 1987.
- [26] Catmull E, Clark J. Recursively generated B-spline surfaces on arbitrary topological meshes[J]. Computer-aided design, 1978, 10(6): 350-355.
- [27] Zorin D, Schrder P, Sweldens W. Interpolating Subdivision for Meshes with Arbitrary Topology[J]. Acm Siggraph Computer Graphics, 1996, 30:pgs. 43-53.

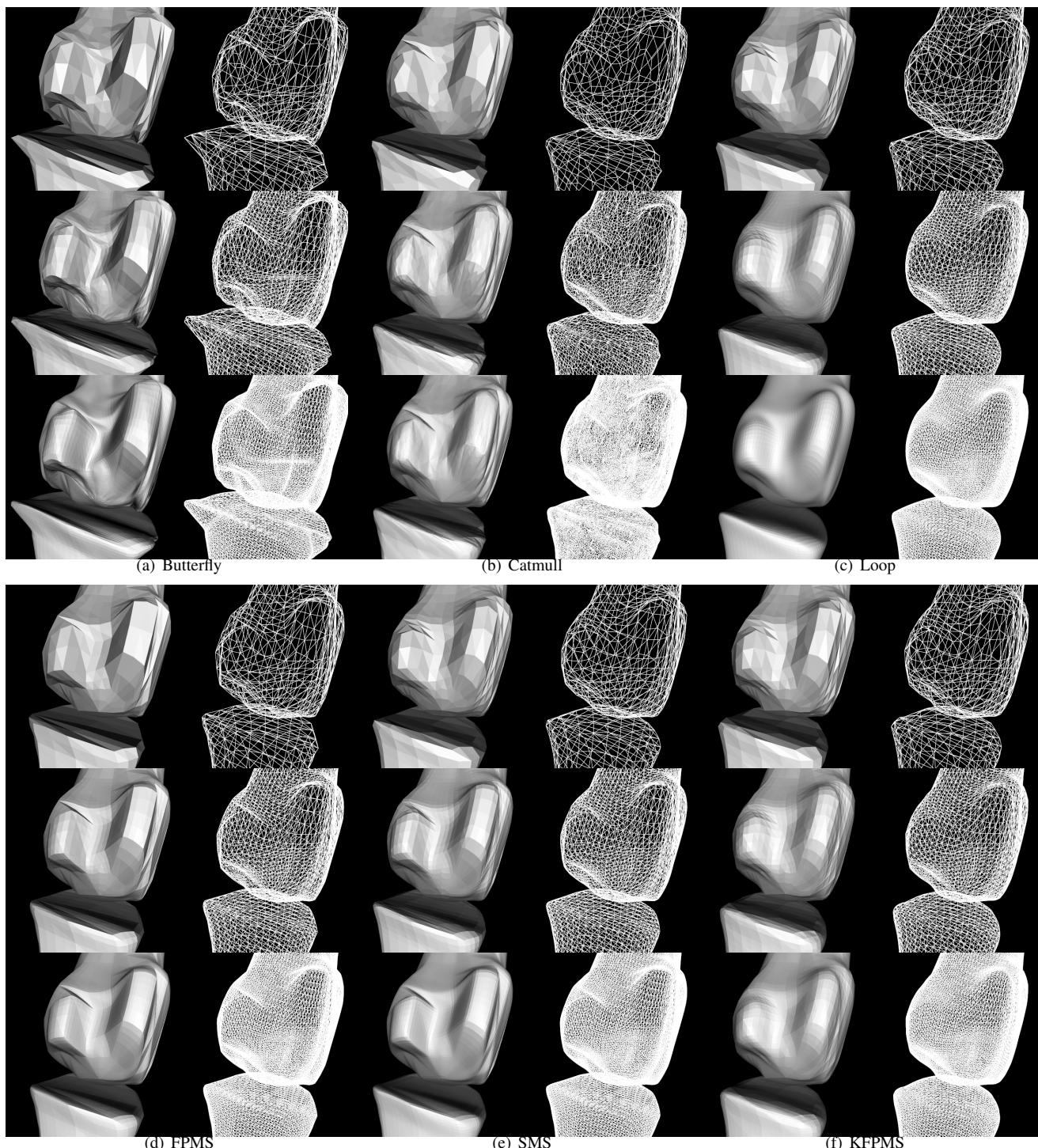


Fig. 9: Subdivision Comparison on biomedical data with iteration increasing by row. a) Butterfly Method. b) Catmull Method. c) Loop Method. d) Feature Preserved Mesh Subdivision. e) Smoothness-focused Mesh Subdivision. f) Keypoints-based Feature Preserved Mesh Subdivision.

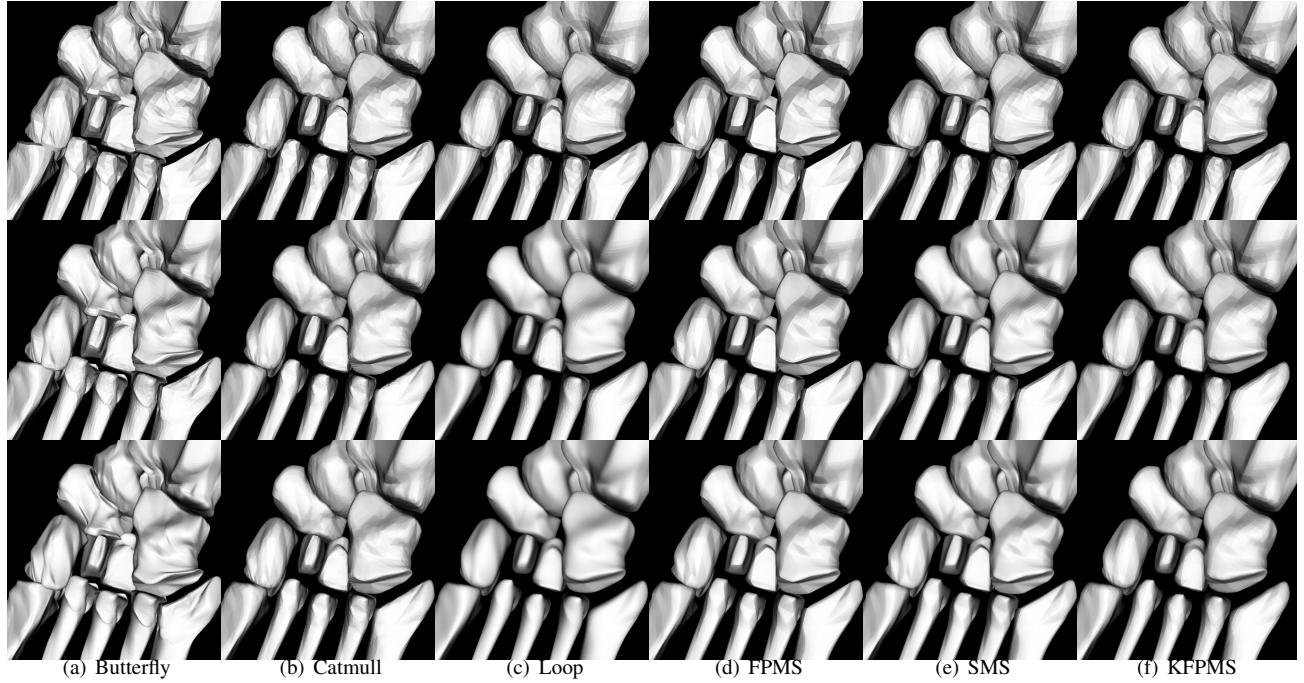


Fig. 10: Subdivision Comparison on biomedical data with iteration increasing by row. a) Butterfly Method. b) Catmull Method. c) Loop Method. d) Feature Preserved Mesh Subdivision. e) Smoothness-focused Mesh Subdivision. f) Keypoints-based Feature Preserved Mesh Subdivision.

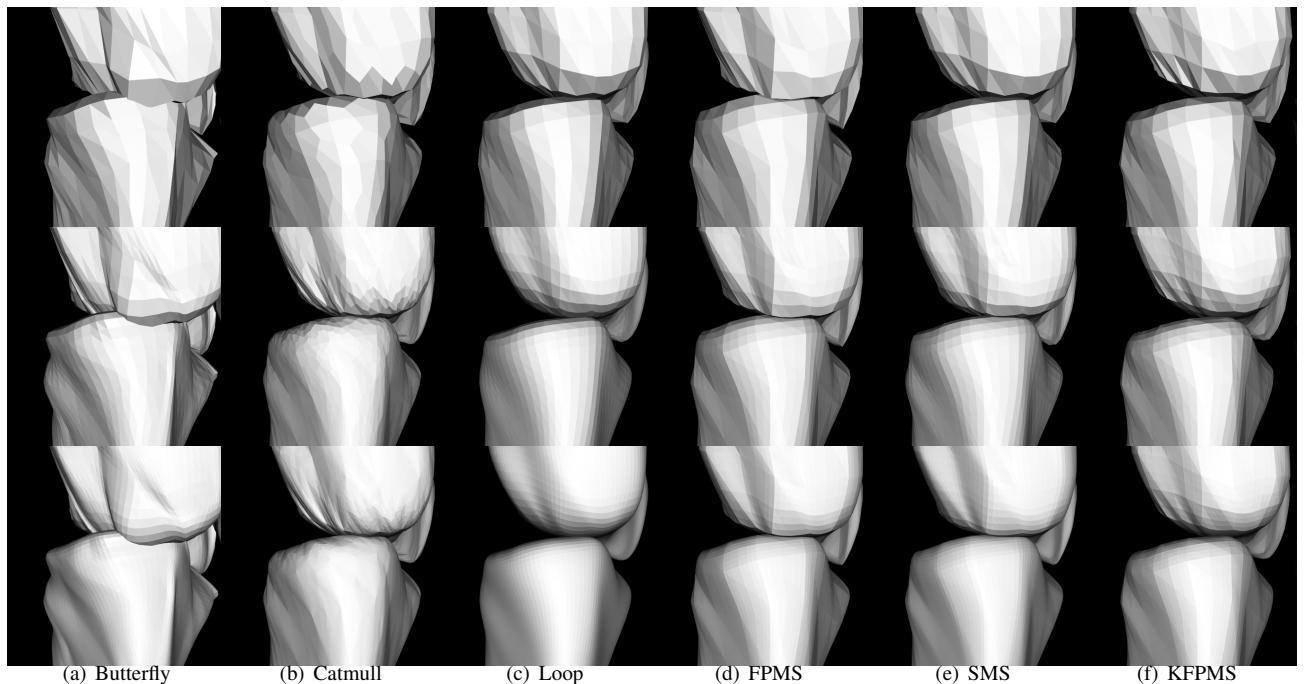


Fig. 11: Subdivision Comparison on biomedical data with iteration increasing by row. a) Butterfly Method. b) Catmull Method. c) Loop Method. d) Feature Preserved Mesh Subdivision. e) Smoothness-focused Mesh Subdivision. f) Keypoints-based Feature Preserved Mesh Subdivision.