PAPER • OPEN ACCESS

Application of Interpolated Loop Subdivision Mesh Deformation in Ecological Environment Animation

To cite this article: Shuzhen Zeng et al 2021 IOP Conf. Ser.: Earth Environ. Sci. 693 012104

View the article online for updates and enhancements.

You may also like

- <u>Study on urban domestic waste recycling</u> process and trash can automatic subdivision standard Huaming Peng and Junxiang Zhou
- <u>The Impacts of Building Contraventions</u> and Informal Residential Land Subdivision on the Quality of Life in Baghdad A.S. Hajer Al Ishaqee and A. Enam Albazaz
- <u>The partition dimension of subdivision</u> graph on the star Amrullah, S Azmi, H Soeprianto et al.

241st ECS Meeting

May 29 – June 2, 2022 Vancouver • BC • Canada Extended abstract submission deadline: **Dec 17, 2021**

Connect. Engage. Champion. Empower. Accelerate. Move science forward



This content was downloaded from IP address 116.1.3.224 on 13/12/2021 at 01:02

Application of Interpolated Loop Subdivision Mesh Deformation in Ecological Environment Animation

Shuzhen Zeng, Zhuo Shi*, Ye Feng, Mingrui Li, Meng Wang and Jing Shen

School of Computer and Information Security, Guilin University of Electronic Technology, 1 Jinji Road, Qixing District, Guilin, Guangxi, China. Email: shzh@guet.edu.cn.

Abstract: In 3D computer vision processing, the introduction of deep learning into 3D research has always attracted much attention. In this paper, we introduce 3D mesh deformation into the animation of ecological environment research, adopt the 3D processing tool of Pytorch3D, and propose a 3D mesh deformation method based on the interpolation Loop subdivision. To prove the superiority of this algorithm, we compare the mesh deformation under the Loop subdivision and interpolation Loop subdivision. The experimental results show that after we adopted the interpolation Loop subdivision when the mesh deformation predicts the target mesh, the result of the deformation mesh is better than the Loop subdivision in terms of details. Thus, mesh deformation can be applied in the animation of the ecological environment, such as the "camouflage" of animals, which is of great significance to the animation or film related to the ecological environment.

1. Introduction

Animation has been playing a significant role in the study of the ecological environment. With the rapid development of 3D technology, 3D geometric data is gradually becoming the fourth type of multimedia data after sound, image, and video. Subdivision algorithm is aimed at a specific polygon processing, it can make a specific shape into a good shape. while danger exists at the moment, some animals protecting themselves, react to the environment in a stressful way, and "disguising" themselves into a different shape to cope with it. Thus, we focus on the self-protection of animals in the ecological environment and make a study on the grid deformation of animals.

The subdivision surface is divided into approximation surfaces and interpolation surfaces. Unlike the surface generated by the approximation model, which does not go through the initial control mesh, the interpolation mode has each control vertex of the initial grid in the generated surface. So the interpolation surface is easy to control the shape of the surface. In the late 1970s, Catmull and Clark proposed the famous Catmull-Clark subdivision mode [1] and they analyzed the discrete method of cubic uniform B-spline surface, which was marking that the subdivision method officially became the method of the surface model. Next, Doo adopted the method of discrete Fourier transform [2] to analyze the convergence of the Catmull-Clark subdivision, which created a precedent for the feature analysis of the convergent matrix of subdivision algorithm. From the late 1980s to the early 1990s, some famous segmentation methods appeared successively, such as four-point interpolation subdivision curve [3], Butterfly subdivision [4], Loop subdivision [5], and so on. Therefore, the convergence and continuity analysis theory of regular subdivision curves and surfaces was gradually improved. But there was a lack of connection between the patterns. Later, Zorin proposed an improved method for Butterfly surface segmentation [6] and extended classical Butterfly surface [4] to any topological mesh. Subsequently, a large number of subdivision modes began to appear to adapt to



different application requirements: $\sqrt{3}$ subdivision [7], interpolation $\sqrt{3}$ subdivision [8], $\sqrt{2}$ subdivision [9], interpolation $\sqrt{2}$ subdivision [10], Kobbelt subdivision [11], quadrilateral grid three-

segment subdivision[12], and so on. In this period, more importantly, the subdivision method was widely used, and a lot of research results were obtained in the multi-resolution analysis of complex mesh surfaces. In terms of the fusion of segmentation methods, many scholars have studied different methods, for example, Osswald[13] realized the fusion of triangular mesh subdivision, Velho[14] realized the fusion of quadrilateral mesh subdivision, etc. Similar studies on subdivision fusion [15][16][17]had the same problem, that was, almost all of them achieved fusion only through local interpolation, and lacked a unified form of expression to achieve fusion through interpolation approximation. After a series of research and analysis, Lin proposed a unified form to represent the fusion of interpolation subdivision and approximation subdivision [18][19]. By adding parameters to control the direction and size of mesh deformation in interpolation and approximation, the fusion of cubic B-spline and four-point subdivision curve was realized, and the convergence and smoothness analysis was carried out. Shi proposed interpolation Loop subdivision [20], which could better maintain the local characteristics of the mesh, we adopt the interpolation Loop subdivision [20], which can better transform the target mesh, and the effect of fitting the target mesh is better.

In this paper, we combine the interpolation Loop subdivision method with the Pytorch3D tool to achieve the 3D mesh deformation in an ecological environment. The aim is to use the shape of the self-protection phenomenon made by animals in response to danger to restore the original appearance of animals. For its reduction process, its meshes are very important to form the original shape of the animal. Thus, we use the source grid after interpolation Loop subdivision processing to complete the prediction of the appearance of animals, making the source mesh closer and closer to the original appearance of animals. In this paper, Loop subdivision and interpolation Loop subdivision methods are used for comparison. The experimental results show that, under the same iteration times, the mesh deformation of the interpolation Loop subdivision has a better ability to transform into the original shape of the animal, especially the local features of the animal are more obvious.

2. Overview of Mesh Deformation Algorithms

The traditional Loop subdivision is approximate, and the surface they generated does not go through the vertices in the initial control mesh, which inevitably causes the limit surface after subdivision to shrink. However, the limit surface produced by the interpolated subdivision mode can't shrink, and it is easy to control the shape of the surface. So we adopt the interpolation Loop subdivision to expand the subdivision template of Loop by adding more control vertices, which is making the edge vertices supplement the central vertices. Therefore, the interpolation Loop subdivision is adopted in this paper to carry out mesh deformation.

Facebook has built the Pytorch3D library to promote 3D deep learning research. The popular existing methods were those proposed by Kato[21] and Liu[22], but the existing methods either do not support batch processing or they assume that the meshes have the same number of vertices and faces. CUDA implementations are only available in existing projects, so you can't use them without a GPU. Pytorch3D library can handle different Numbers of vertices and faces, and also supports GPU, which greatly simplifies 3D mesh deep learning research. Thus, to speed up mesh deformation, we use the Pytorch3D library and GPU to train. The mesh predictor we adopted is the same as the Mesh-RCNN[23], and the shape loss is approximate to the Mesh loss. It is challenging to define the loss of local operations on triangular meshes. We use a loss function defined on a finite set of points which is used to represent a mesh by densely sampling point clouds from the surface of the mesh. Consistent with [24], differentiable network sampler is used to evenly sample points from the mesh surface, point cloud from the source grid P^i , and point cloud from the target grid Q^i . The calculation process of grid loss is as follows: given two sets of point cloud P, Q and normal vector, make $\wedge_{PQ} = \{(p, \arg\min_a || p - q ||) : p \in P\}, (p, q)$ becoming a set of data pair (P, Q), where q is the nearest

neighbor of p in Q, and let the u_p become the normal vector of point P. The chamfer distance between point cloud P and Q is:

$$L_{\text{cham}}(P,Q) = |P|^{-1} \sum_{(p,q) \in \wedge_{P,Q}} ||p-q||^2 + |Q|^{-1} \sum_{(q,p) \in \wedge_{Q,P}} ||q-p||^2$$
(1)

The normal distance is:

$$L_{\text{norm}}(P, Q) = -|P|^{-1} \sum_{(p,q) \in A_{P,Q}} |u_p \cdot u_q| - |Q|^{-1} \sum_{(q,p) \in A_{Q,P}} |u_q \cdot u_p|$$
(2)

Chamfer distance and normal distance measure mismatches and normal between two-point clouds, but only simply minimizing these two distances can result in mesh degradation. Therefore, the highquality mesh prediction requires extra shape adjusters, so we adopt edge loss is:

$$L_{\text{edge}}(V, E) = \frac{1}{|E|} \sum_{(v, v') \in E} ||v - v'||^2$$
(3)

In addition, we also added the Laplace loss[25], which can impose smoothing constraints.

Our mesh deformation method is that subdividing the initial mesh by interpolation Loop to obtain the source mesh, and use the source mesh to predict the target mesh, which is the original appearance of the animal. The algorithm process is shown in figure 1.



Figure 1. Mesh deformation flow chart.

3. Experiment Part

Our implementation of this algorithm is using Intel(R) Core(TM) I5-5200U CPU @2.20ghz and 8GB of memory. We test the algorithm for mesh deformation and interpolation Loop subdivision under the Ubuntu system, and the GPU is the NVIDIA Geforce 930M. In this experiment, the original mesh is subdivided by Loop and interpolation Loop to obtain the source mesh, selecting the target mesh (this is the 3D animal model), and Pytorch3D Complete the process of mesh deformation. Our experiment is with the same number of sampling points (this is point cloud). The result of the experiment is the deformed result mesh of the source mesh. The purpose of the deformation is to be as close as possible to the target mesh. As we can be seen from figure 4, the initial mesh adopts the Loop subdivision mesh, during the predicted deformation process of Bunny, the ear of the Bunny is relatively sharp and there is a broken Angle, which is far from the target mesh and the deformation result is poor. This is

Table 1. Hausdorff distance between the deformation result mesh and the target mesh.

Subdivision	bunny	mousehead
Loop subdivision	0.025621	0.07518
Interpolated Loop subdivision	0.035352	0.013013



Figure 2. Target mesh:Bunny.



Figure 5. The interpolated Loop subdivision



Figure 3. The Loop subdivision result mesh.



Figure 4. The deformation results of figure 3.



Figure 6. The deformation results of figure 5.

result mesh.



Figure 7. Target mesh: mousehead.



Figure 8. The Loop subdivision result mesh.



Figure 9. The deformation results of figure 8.



Figure 10. The interpolated Loop subdivision result mesh.



Figure 11. The deformation results of figure 10.

because Pytorch3D uniformly sampled the surface of the source and target meshes when it predicts the target meshes.Pytorch3D sampled the source mesh and target mesh to generate barycentric coordinates randomly and uniformly and achieve the goal of predicting the target mesh. The Loop subdivision of the source mesh can result in the mesh is too smooth, which will lead to sharp local features or broken angles of the deformed structure mesh. In this paper, the source mesh subdivided by interpolation Loop is used to overcome this shortcoming. As can be seen from figure 6, after the source grid in this paper adopts the interpolation Loop subdivision, the shape of the ear of the deformed target mesh is roughly consistent with that of the target mesh, and the deformation effect is obviously better. The Bunny's cheeks and calves are also relatively better than the Loop subdivision mesh distortion. Figure 9 shows that the target mesh is mouseHead, and the deformation result mesh of Loop subdivision is too sparse at the ears and nose to fit the target mesh well. It can be seen from figure 11 that the mesh deformation results of interpolation Loop can fit the mouseHead well at the ears and nose, which is almost consistent with the target mesh.

According to the Hausdorff distance in table 1, the Hausdorff distance between the deformation results and the target grid is very close. However, according to the analysis in shapes, the mesh deformation results of interpolation Loop are better in the local features of the target mesh fitted.

To sum up, the mesh after interpolation Loop subdivision to complete deformation is better than the Loop subdivision, our method can better maintain the local features of the target meshes.

4. Conclusion

Mesh deformation can apply to the shape deformation of animal stress self-protection in an ecological environment, which is of great significance for animation research and virtual reality in an ecological environment. In our paper, we propose a mesh deformation method which bases on the interpolation Loop subdivision, and it can predict the target mesh well and fit the local characteristics of the target mesh better. During the training, we adopt the neural network optimizer SDG which can accelerate the deformation of the grid. In the study of ecological environment animation, the shape of animals changes when they "camouflage" or protect themselves, so our mesh deformation method can restore their original appearance. In the future, we will continue to study the metamorphosis between animals to achieve more realistic effects.

5. References

- [1] E. Catmull, and J. Clark. Recursively generated B-spline surfaces on arbitrary topological meshes. Computer Aided Design, 1978, 10(6): 350~355
- [2] D. Doo and M. Sabin. Behavior of recursive division surfaces near extraordinary points. Computer Aided Design, 1978, 10(6):356~360
- [3] N. Dyn, J. A. Gregory and D. Levin. A 4-point interpolatory subdivision scheme for curves design. Computer Aided Design. 1987, 4: 257~268
- [4] N. Dyn, D. Levin and J. A. Gregory. A butterfly subdivision scheme for surface interpolatory with tension control. ACM Transactions on Graphics. 1990, 9(2): 160~169

- [5] C. Loop. Smooth subdivision surfaces based on triangles. [Master's thesis], Utah: University of Utah, Department of Mathematics, 1987
- [6] D. Zorin, P. Schroder, and W. Sweldens. Interpolating subdivision for meshes with arbitrary topology. In: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques, 1996, 189~192
- [7] L. Kobbelt, subdivision. In: Proceedings of the 27th annual conference on Computer graphics and interactive techniques, 2000, 103~112
- U. Labsik, G. Greiner. Interpolatory subdivision. Computer Graphics Forum. 2000, 19(3):131~138
- [9] G. Li, W. Ma, H. Bao. Interpolatory subdivision surfaces. In: Proceedings of Geometric Modeling and Processing, Beijing, 2004, 185~194
- [10] G. Li, W. Ma, H. Bao. Subdivision for Quadrilateral Meshes. The Visual Computer. 2004, 20:180~198.
- [11] L. Kobbelt. Interpolatory subdivision on open quadrilateral nets with arbitrary topology. Computer Graphics Forum. 1996, 15(3):409~420
- [12] G. Li, W. Ma. Interpolatory ternary subdivision surfaces. Computer Aided Geometric Design, 2006, 23(1):45~77
- [13] Oswald P. Designing composite triangular subdivision schemes[J]. Computer Aided Geometric Design, 2005, 22(7): 659-679.
- [14] Velho L. Using semi-regular 4– 8 meshes for subdivision surfaces[J]. Journal of Graphics Tools, 2000, 5(3): 35-47
- [15] J. Stam, C. Loop, Quad/Triangle subdivision, Computer Graphics Forum, 2003,22(1): 79~86.
- [16] J. Peters, L.J. Shiue, Combining 4- and 3- direction subdivision. ACM Transactions on Graphics,2004.23(4): 980~1003.
- [17] S. Schaefer, J. Warren, On C2 subdivision for triangle/quad meshes. Transactions on Computer Graphics, 2005.24(1):28~36.
- [18] Shujin Lin, Fang You, Xiaonan Luo, Li Zheng. Deducing interpolating subdivision schemes from approximating subdivision schemes[C]//ACM Transactions on Graphics (TOG). ACM, 2008, 27(5): 146.
- [19] Shujin Lin, Xiaonan Luo, A Unified Interpolatory and Approximation Subdivision Scheme, Eurographics 07, Prague, Czech, Sept.3-7, 2007, Short paper, In Eurographics Digital Library.
- [20] Zhuo Shi, Jinshu Lin, Xiaonan Luo, Renhong Wang. Interpolatory and Mixed Loop Schemes. Pacific Graphics 2008, Computer Graphics Forum. 2008, 27(7):1829-1835.
- [21] Kato H, Ushiku Y, Harada T. Neural 3d mesh renderer[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 3907-3916.
- [22] Shichen Liu, Weikai Chen, Tianye Li ,Hao Li. Soft rasterizer: A differentiable renderer for image-based 3d reasoning[C]//Proceedings of the IEEE International Conference on Computer Vision. 2019: 7708-7717.
- [23] Gkioxari G, Malik J, Johnson J. Mesh R-CNN[J]. 2019.
- [24] Blanz V, Vetter T. A morphable model for the synthesis of 3D faces[C]. international conference on computer graphics and interactive techniques, 1999: 187-194.
- [25] Desbrun M, Meyer M, Schroder P, et al. Implicit fairing of irregular meshes using diffusion and curvature flow[C]. international conference on computer graphics and interactive techniques, 1999: 317-324.