2014 International Conference on Virtual Reality and Visualization

A survey on recent Approaches of Mesh Compressions

Huaiyu LI, Weiliang MENG, Xiaopeng ZHANG*
(National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China)
E-mail:xiaopeng.zhang@ia.ac.cn

Abstract

In this paper, we give a survey on mesh compression approaches of the past few years. Since this technique has gradually become mature, we not only have summarized the previous methods such as single-rate mesh compression and progressive mesh compression, but also review mesh compression methods with random accessibility. The classification for each kind of methods is given, and the trend of mesh compression techniques development is analyzed based on the limitation of the current approaches.

Keywords: mesh compression, progressive mesh, random accessibility

1. Introduction

Since the mesh compression concept was proposed in the early 1990's, it has been a hot research topic for many years. In recent years, not only the computing power of the PC but also the network bandwidth has made a huge leap comparing with these resources decades ago. However, to achieve a higher level of realism of the 3D model, the ever-increasing precision of 3D meshes are required in the domains such as medical imaging, computer-aid design, digital heritage and entertainment, and current hardware still can't satisfy the storing, accessing and transmitting these meshes data efficiently and rapidly. In order to get an excellent user experience, the efficient mesh compression algorithm is indispensable and the interactivity capability should be enhanced by using appropriate methods.

The methods for mesh compression can be divided into three categories: Single-rate compression, progressive mesh compression and random accessible mesh compression. And in this survey, some basic definitions about mesh attributes are first briefly described before reviewing the mesh compression approaches in section 2, which are usually mentioned in the relevant literatures. Then we intend to review the single-rate compression techniques and progressive compression algorithms simply in section 3, because there is a good survey could be found in [1-2]. In the section 4, we will mainly focus on the development of progressive mesh compression techniques in the recent decade. And then we will present the new trend of mesh compression with random accessibility in section 5. Finally, the conclusion and some discussions are given in section 6.

2. Basic Concepts

A 3D mesh model is usually represented as a collection of vertices, edges and faces, including geometry information, connectivity information and other attributes information.

The geometry information of the mesh refers to the positions of the vertices of a mesh in the space, and the connectivity information is the connecting relations among vertices, edges and faces, and it could also be called topology information. The attributes of the mesh include the normal of the face and vertices, the color of the vertices, the texture coordinates, and so on.

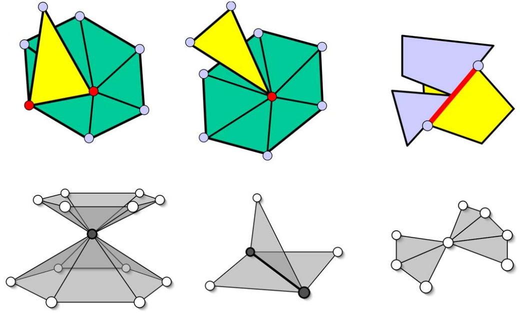


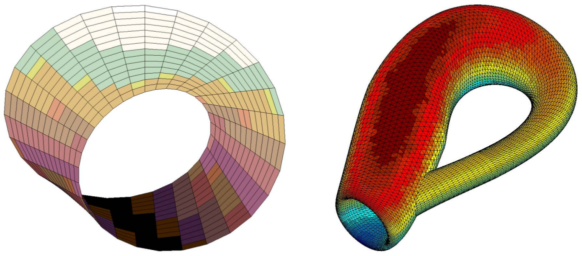
Figure 1. Some non-manifold meshes

Manifold is an important concept in mesh compression. A mesh is a manifold if each edge is incident to only one or two faces and the faces incident to a vertex formed a closed or an open fan. The orientation of a face is a cyclic order of the incident vertices. The orientation of two adjacent faces is compatible. If the two vertices of the common edge are in opposite order. A manifold mesh is orientable if any two adjacent faces

978-1-4799-6854-1/14 $31.00 © 2014 IEEE
DOI 10.1109/ICVRV.2014.2

have compatible orientation. Some examples of non-manifold meshes are illustrated in Fig1.

However, not all the manifold meshes are orientable. The most well-known ones are Mobius band and Klein Bottle. The Mobius band is an one-sided manifold with boundary,i.e., a circle(Fig. 2a), and the Klein bottle is an example of a non-orientable surface with no boundary(Fig. 2b).



(a) Mobius band (b) Klein bottle

Figure 2. Mobius band and Klein bottle, examples of non-orientable meshes

For the mesh compression technique, we mainly focus on the compression of the geometry and connectivity information. The other attributes information of the mesh can be compressed based on the similar principle.

3. Single-rate compression

Single-rate compression schemes mainly focus on the reduction of storage size. Since a typical mesh contains connectivity data and geometry data, these two parts are usually encoded separately in mesh compression algorithms. At present, the techniques of single-rate compression can be divided into two main categories: the connectivity coding and the geometry coding.

3.1 The connectivity coding

In [1], the algorithms for connectivity compression are classified into six classes: the indexed face set, the triangle strip, the spanning tree, the layered decomposition, the valence-driven approach, and the triangle conquest.

The indexed face set method denote a triangular mesh as an indexed face set which consists of a coordinate array that lists the coordinates of all vertices and a face array that lists the index of each face’s three vertices. However, the repeat of the vertex references degrade the efficiency of connectivity coding.

The triangle strip method divides a 3D mesh into long strips of triangle and then these strips are encoded. This method such as [3] is suitable for meshes with any topology, especially better for long triangle strips.

In the spanning tree method, a run is a basic coding unit. Based on this method, Taubin and Rossignac[4] develope an algorithm that can encode general manifold meshes such as meshes with boundary, meshes with arbitrary genus and non-orientable meshes. But their method cannot directly deal with non-manifold meshes.

Bajaj et al.[5] employ a layered structure of vertices for the connectivity coding method to decompose a triangular mesh into several concentric layers of vertices, and then construct triangle layers within each pair of adjacent vertex layers. The mesh connectivity is denoted as the total number of vertex layers, the layout of each vertex layer, and the layout of triangles in each triangle layer.

Valence-driven approach has an excellent compression ratio performance. The approach was first proposed by Touma And Gotsman[6], and then Alliez and Desbrun[7] further improve the performance of this approach. But Alliez and Desburn’s algorithm is only applicable to orientable manifold mesh.

The triangle conquest approach is similar to the valence-driven approach. Readers can refer the edgebreaker algorithm[6] as an example of the triangle conquest approach.

Table 1 lists the single-rate compression algorithms ratios in bits per vertex (bpv) separately, which are extracted from the table in [2] for a quick view, and readers can get more details from the Pierre and Gotsman’s work [2].

Table 1. Compression ratio of single-rate algorithms

Category	Algorithm	Ratio(bpv)
Indexed face set	VRML format	$6\log_2 v$
Triangle strip	Deering[3]	11
Spanning tree	Taubin [4]	2.48-7.0
Layered decomposition	Bajaj et al.[5]	1.4-6.08
Valence-driven	Alliez [7]	0.024-3.24
Triangle conquest	Rossignac[6]	4

3.2 The geometry coding

For the connectivity coding algorithm, current best performance is regarded as being very close to the optimum, and as the geometry data dominates the total mesh data, now mesh compression research mainly shift to geometry coding. Compared to lossless encode scheme for connectivity data, the geometry data is usually encoded in a lossy manner. In order to exploit the high correlation between the positions of adjacent vertices, most of the geometry compression algorithms follow a three-step procedure: the quantization of vertex position, prediction and Entropy coding. Different geometry coding methods will use different prediction methods, such as delta prediction, linear prediction and quadratic prediction.

For more details about single-rate compression methods, there are two excellent survey [1-2] for readers to explore.

4. Progressive mesh compression

The progressive mesh compression techniques use the notion of refinement, and the original mesh is transformed into hierarchy or a sequence of refinements applied to a simple, coarse mesh. Extract the levels of detail of mesh is allowed during the decompression, and when the data is received and decompressed, the user can see the gradual mesh refining process. Depending on the viewpoint or the capabilities of visualization devices, the most appropriate level of detail will be displayed. The main challenge of the progressive mesh compression algorithms is to achieve the best rate-distortion performance. The generated levels of detail must be as close as the initial mesh. The progressive mesh can be illustrated by Fig3.

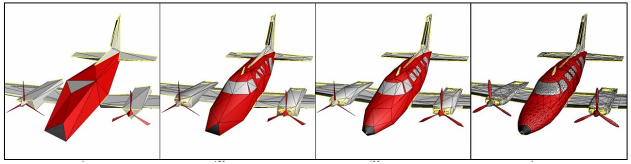


Figure 3. The illustration of progressive mesh from [7].

The progressive mesh compression techniques generally can be classified into two main categories: the connectivity-based compression and the geometry based compression. Although these methods seem to be similar to those in single-rate compression schemes, the purposes and results are totally different.

4.1. Connectivity-based compression

The concept of progressive mesh is first introduced by Hoppe [7], which uses the edge collapse operation to successively decimate a mesh. When an edge is collapsed, its two end points are merged into one, and two triangles incident on this edge are removed. The compressed data contains the coarse mesh followed by all parameters required for the vertex splits operation which is the reverse operation of edge collapse. The main advantage of this method is its high multi-resolution granularity, and it is possible to perform selective refinement during the decoding. Finally, this method get a compression ratio of 37 bits per vertex (bpv) with 10 bits quantization. After that, Popovic and Hoppe in [8] generalized the PM representation to arbitrary simplicial complexes, and with that representation, a model generally requires about 50bpv with 10 bits quantization.

In order to make the compression ratio closer to the single rates method, Taubin et al. [9] are inspired by the single-rate topological surgery algorithm and create a new progressive mesh compression scheme using a representation called progressive forest split (PFS). This representation encodes a manifold triangular mesh with a base mesh and a sequence of forest split operations. The forest split operation includes cutting the mesh through several sets of connected edges, filling the generated holes with triangles and relocating the vertices. Because of the expense of reduced granularity, PFS method can achieve a much higher compression ratio than previous method. Pajarola and Rossignac[10] impose some restriction for choosing the candidates to the edge collapse operations to improve the compression rates. The operations are grouped into batches during the traversal of a spanning tree, and the geometry coding are improved by using a butterfly predictor. Karni et al. [11] create an efficient vertex rendering sequence composed of series of incident vertices, and the mesh is decimated by collapsing edges along this sequence. Their method allows to render all the LODs rapidly and get a better compression ratios at the same time, but the multiresolution granularity is impacted compared to the PM representation.

Many other progressive compression schemes use vertex removals instead of edge collpases. Based on vertex removal followed by a local patch retriangulation, Li and Kuo[12] encode the connectivity with a local index and a global index that locates pattern in the whole mesh, and the geometry data is encoded with a barycentric error prediction. In [13], Alliez and Desbrun employ a single-rate encoder to the progressive mesh compression scheme, which decimates the mesh by two deterministic patch traversals and then encodes the connectivity through the valence of the removed vertices and geometry through the patch barycentric error prediction in a local Frenet frame respectively. Finally, the obtained compression rate is about 13 bpv with 10 bits quantization.

Valette et al. used a wavelet framework to build a progressive mesh compression algorithm, which progressively decimates the initial mesh with a subdivision scheme tailored to irregular meshes. The connectivity is encoded in face subdivision operations and the geometry data is encoded through a wavelet lifting scheme. The compression rate is about 19bpv with 12bits quantization.

4.2. Geometry-based compression

Since the compressing geometry is generally more effective than the compressing connectivity, Gandoin and Devillers [14] focus their effort on geometry compression. In their scheme, the vertex positions are stored in a KD-tree, the number of points located in each cell of the KD-tree hierarchy is encoded into entropy, and the connectivity is encoded using vertex splits. Peng and Kuo [20] also give a geometry-based progressive mesh compression scheme based on an Octree data structure, using the geometry data of the neighbor vertices during the vertex splits to predict the connectivity of the mesh. This algorithm can compress triangle meshes into 15bpv with 12 bits quantization.

4.3. Other compression scheme

Besides connectivity-based and geometry based schemes, many other methods have been used. The wavelet transform is utilized in the a progressive mesh compression algorithm proposed by Khodakovsky et al.[15] and then is improved in [16] through a normal mesh representation. According to the successful use of spectral coding in the compression of 2D image and video data, Karni and Gotsman[17] use the spectral theory on 3D mesh to compress geometry data. After being quantized and entropy coding, the spectral coefficients are sufficient to decompress a good approximation of the initial mesh. Recently, Mamou et al. [18] create an algorithm based on the Laplacian matrix of a mesh, which approximates the mesh with a heat equation and a minimal set of control points, and then encodes the vertex locations as residuals from the approximation, with encoding the connectivity by the single-rate encoder from [19]. This scheme obtains an excellent compression ratio, but its high complexity is time-consuming especially on solving the heat equation.

In recent work [22], based on a mesh-aware valence coding scheme for multiresolution meshes, Junho Kim use a Bayesian AD coder to gain a better compression ratio in connectivity coding than the original Alliez Desbrun[13] coder. The Bayesian AD coder indirectly encodes the valence according to its rank in a sorted list with respect to the mesh-aware scores of the possible valences, and experimental results shows that this coder get an improvement of 8.5-36.2% in connectivity coding compared to the original AD coder.

A progressive 3D triangular mesh compression algorithm built on the MOG-based Bayesian entropy coding and the gradual prediction scheme is given by Dae-Youn Lee et al. [23]. They use MOG models to get probabilities of topology symbols and encode the probabilities by an arithmetic coder. For geometry encoding, vertices are divided into groups and the information in already encoded groups is used to predict later groups. The simulation results demonstrate that their algorithm can provide better performance than conventional wavelet-based coder.

4.4. Compression for general polygon meshes

The methods discussed above are all used for triangle mesh compression. For general polygon meshes that may include quads or pentagons or other polygons, they must be transformed into triangle meshes first before using those algorithms. A classical approach to deal with general polygonal meshes depends on first triangulating the polygon mesh and then use an existing method aiming at triangle mesh. This approach is proposed by Taubin et al.[9] to extend the progressive forest split algorithm. Peng and Kuo [20] discuss the progressive compression of polygon meshes by an octree coder, and their algorithm can compress arbitrary connectivity between vertices, making modify the face construction algorithm to reconstruct polygon faces possible. The mesh connectivity is encoded through vertex splits and efficient prediction of pivots vertices.

In recent work [21], Adrien Maglo et al. present a new algorithm for the progressive compression of manifold polygon meshes, decimating the input surface by several traversals and generate successive levels of detail through a specific patch decimation operation that combines vertex removal and local remeshing. The mesh connectivity is encoded by Boolean error prediction, while the geometry is encoded with a barycentric error prediction of the removed vertex coordinates and a local curvature prediction. The methods they use to improve the rate-distortion performance is a wavelet formulation with a lifting scheme and an adaptive quantization technique. According to the experiment results, the approach can effectively handle surface meshes with arbitrary face degree in terms of compression rates and rate-distortion performance.

Table [2] gives a quick glimpse for progressive mesh compression algorithms, part of which are extracted from the table in [1], and others from recent original papers. Readers can get more details by referring the corresponding papers for better understanding. .

Table 2. Progressive mesh compression ratios.

Category	Algorithm	Ratio(bpv)	Feature
Progressive meshes	Hoppe[7]	37	High granularity
PFS	Taubin et al.[9]	7-15	Higher compression ratio
Valence-driven conquest	Alliez and Desbrun[13]	3.7-16	Good rate-distortion ratio
KD-tree decomposition	Gandoin and Devillers[14]	10-17	Encoding triangle soups
Octree decomposition	Peng and Kuo[20]	Improve 40-90% of [14]	Arbitrary connectivity
Bayesian AD coder	Junho Kim[22]	Improve 8-36% of[13]	Improve connectivity coding
Entropy coding	Dae-Youn Lee et al.[23]	7-20	Faster decoding
	Adrien Maglo et al.[21]	8-18	For manifold polygon meshes

5. Random accessible mesh compression

As we have described above, the single-rate compression algorithms build a kind of compact representation of a 3D mesh. But to obtain the initial mesh, the input compressed data have to be fully decompressed. For very large meshes, this process is very time-consuming. While the progressive mesh could compress the original meshes into many levels of detail and visualize successive LODs during the decompression. In some cases this property is not enough for excellent interactive visualization. To access a specific region of the mesh, the single-rate and progressive algorithms have to fully decompress the mesh.

The random accessible mesh compression approaches can make a different experience of interactive visualization. This approach allows to decompress only specific parts of the mesh that the user is interested in, but in some cases, the users could not have any overview of the other regions while they access the requested regions of the mesh. This is the main defect of the approach, but the problem can be well solved by combining the progressive and random accessible techniques to decompress different parts of a mesh at different levels of detail. The random accessible mesh compression is shown in Fig. 4.

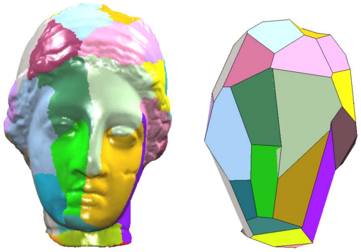
Figure 4. Illustration of Random accessible mesh from [36]. Note that different part of the dragon can be displayed in different resolution.

The purpose of the random accessible mesh compression is to partially remove the dependencies between the mesh elements, and two main paradigms are proposed in the literature: the cluster-based methods and the hierarchical representation methods.

5.1. The cluster-based methods

Junho Kim et al. [25] give a selective refinement scheme of progressive meshes, using the concept of a dual piece to enumerate and visualize the set of selectively refined meshes for a given mesh. Since they only focus on the topology to a selectively refined mesh, the method can cause the triangle flipping problem.

After that, Choe and Kim [26] come up with an effective framework of random accessible mesh compression scheme that allows the decoding of the desirable parts in an arbitrary order without decoding the other non-interesting parts. The concept “charts” is introduced which is the separate segments decomposing the given mesh and the common boundary between two adjacent charts is defined as “wire”. Before encoding, the original mesh with the chartification is processed to construct a polygonal mesh called wire-net mesh, as shown in the Fig 5.



(a) chartification (b) wire-net mesh

Figure 5. Mesh chartification and corresponding wire-net mesh from [26].

For encoding, after generating the chartification of the given mesh to get the structure of the wire-net mesh, wires and charts, the technique proposed by Khodakovsky et al. [27] is adopted to compress the connectivity of a wire-net mesh and use the parallelogram prediction [28] for geometry coding. Then, the wire and chart are encoded separately by using a linear prediction and the Angle-Analyzer which is a single-rate compression algorithm with excellent compression ratio. The Fig.6 illustrates the file structure of their scheme.

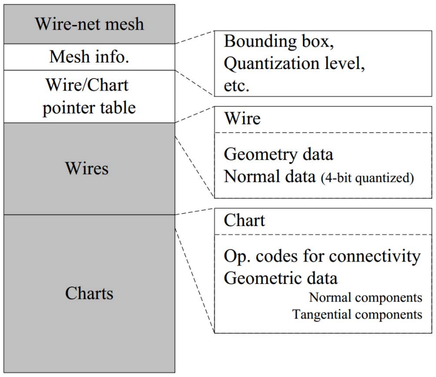


Figure 6. File data structure from [26].

According to the experiment results, high compression ratio can be achieved, only a slightly worse than the ratio of single-rate compression in most cases.

Based on the refinement framework [25], Junho Kim et al. [24] give a multiresolution random accessible mesh compression algorithm providing the progressiveness as well as the random accessibility, compared with the approach [26] that takes a chart as the atomic unit for random accessibility but does not support progressive decoding for selected parts. The distinguishable point of their work is the asymmetry between the encoder and decoder, but this leads to restrictions in compression performance. During the decompression, a vertex can be split even if its neighbors are not identical to the neighbors when the edge was collapsed, which in turn achieves a well-grained multiresolution random access. The compression ratio is not superior comparing with [26], but obtaining the progressivity in [26] is more difficult. And in their most recent paper [29], Choe and Kim provide random accessible mesh compression with better compression ratio and explicit control of random accessibility, and this improved framework can also be scaled up to handle very large meshes.

Yoon and Lindstrom [32] propose a cluster-based random accessible compression scheme, which is based on streaming mesh compression in order to preserve the input order of triangles and achieve relatively high compression and decompression throughput. Two major components are mainly included: a cluster-based order-preserving mesh compression method and a runtime decompression framework that transparently supports random access. Nevertheless, the compression rates are not very competitive compared to the approach of Choe et al. [29]

In the recent work of Maglo et al. [30], the cluster-based random accessible approach from [29] is extended to support the progressive compression of the clusters. The method is targeted at visualizing of large meshes and supporting the encoding of the vertex colors. The original single-rate cluster compression algorithm is replaced by a progressive one which is based on [13]. In order to prevent duplication of the geometry information of the chart border vertices, progressive mesh encoder from [31] is modified to encode these vertices with the wire data. This approach also integrates a time consuming clustering step, and uses a post-processing step to stitch the clusters together in order to fill the boundary holes during the compression at the targeted levels of detail.

Maglo et al. [35] proposed a novel random accessible and lossless mesh compression with progressivity algorithm called POMAR. Their compression algorithm consists of three major tasks: mesh decimation, global level of detail compression and clustered level of detail compression. The first step uses the half-edge collapse to decimate the mesh. When generating a new LOD, a judgment of face normal is used to avoid the collapsed edge violating the manifold property. The performed halfedge collapse is recorded to be used in the reconstruction step that uses the vertex split operation and recorded collapsed operation to reconstruct the successive levels of detail before the encoding of the operation. This scheme is also cluster-based. To achieve random accessible visualization, the clusters are divide into global clusters and clusters in each LOD. The compressed file structure is shown in Fig. 7.

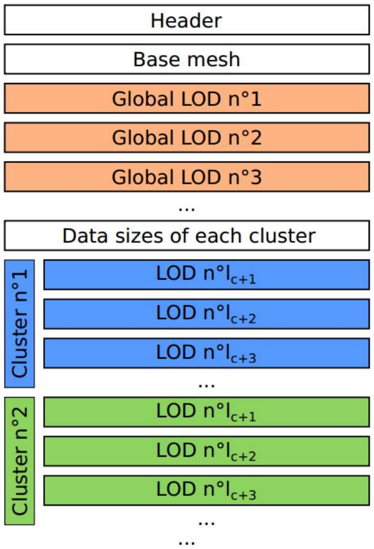


Fig. 7. The file structure of POMAR from [35].

During the decompression, as long as a single cluster on the base mesh is selected, its adjacent cluster will be decompressed at the same time to get a smooth transition. Unlike previous cluster-based approaches, POMAR is able to select any position on the base mesh, thanks to the storage of the data size of each cluster.

5.2. The hierarchical representation methods

Unlike the cluster-based methods which divide a mesh into independently compressed charts and a base coarse mesh, Courbet and Hudelot proposed a hierarchical representation of the mesh in [33]. A boundary-based approach is used to recursively split the mesh into two partitions until each partition contains only one polygon, and each generated submesh can be decompressed independently. It also can compress meshes with arbitrary polygons instead of only triangles and is very simple to implement. The coder finally achieve 14bpv for geometry coding using 12 bits quantification. But the compression efficiency for triangle meshes is inferior to cluster-based methods.

The ChuMI viewer is given by Jamin et al. [34], which combines good performance in the aspect of both the compression rate and the visualization frame rates. To handle meshes with no size limitations and allow local refinements, the mesh bounding box are partitioned into a hierarchical structure called the SP-tree in which the original data structures are embedded in a way that optimizes the bit distribution between geometry and topology. To enable independent decoding of each cell, it duplicates the vertices belong to several SP-cells, and uses the Gandoin and Devilliers [14] algorithm to encode SP-cells.

We summarize relative experiment results and their features in Table 3 for comparison in order to give an intuitive description of these algorithms.

Table 3. Random accessible mesh compression ratios.

Category	Algorithm	Ratio(bpv)	Features
The Cluster-based methods	Choe and Kim[26]	14-22	Decode desirable parts in arbitrary order
	Junho Kim et al.[24]	15-30	Progressivity and random accessibility
	Choe and Kim[29]	8-18	Better compression ratio
	Maglo et al.[30]	8-20	Large mesh visualization and color encoding
	Maglo et al.[35]	13-23	Separate global and Lod clusters
Hierarchical representation methods	Courbet and Hudelot [33]	14-24	Hierarchical representation
ChuMI viewer	Jamin et al.[34]	18-28	Dealing with any n-D simplicial complex

6. Conclusion

After decades of development on mesh compression technique, researchers have proposed many kinds of schemes to solve this problem, and the mesh compression technique is becoming mature gradually. However, there still exists the development space for this topic. On the one hand, the mesh compression technique mainly focuses on processing regular, manifold meshes, and the mesh compression algorithms aiming at dealing with non-manifold meshes are limited. On the other hand, it is necessary to combine the mesh compression techniques with progressivity and random accessibility, as this can highly facilitate users in the application on the internet. For the future development of the mesh compression techniques, we predict that the study of processing irregular, non-manifold mesh models will become popular, since those models are ubiquitous and this is still an open problem. Besides, in most 3D applications on the internet such as online computer games, the precise models usually are irregular and the corresponding mesh compression should be studied to achieve a more feasible pattern, which is not be well dealt with by current algorithms.

Acknowledgment

This work is supported by National High-tech R&D Program (2013AA013701).

References

[1] Peng, Jingliang, Chang-Su Kim, and C-C. Jay Kuo. "Technologies for 3D mesh compression: A survey." Journal of Visual Communication and Image Representation 16.6 (2005): 688-733.

[2] Alliez, Pierre, and Craig Gotsman. "Recent advances in compression of 3D meshes." Advances in Multiresolution for Geometric Modelling. Springer Berlin Heidelberg, 2005. 3-26.

[3] M. Deering, Geometry compression, in: ACM SIGGRAPH, 1995, pp. 13–20.

[4] G. Taubin, J. Rossignac, Geometric compression through topological surgery, ACM Trans. Graph. 17 (2) (1998) 84–115.

[5] C.L. Bajaj, V. Pascucci, G. Zhuang, Single resolution compression of arbitrary triangular meshes with properties, Computer. Geometry. Theory. Appl. 14 (1999) 167–186.

[6] J. Rossignac, Edgebreaker: connectivity compression for triangle meshes, IEEE Transactions on Visualization and Computer Graphics. 5 (1) (1999) 47–61.

[7] H. Hoppe, Progressive meshes, in: Proc. of SIGGRAPH, 1996, pp. 99–108.

[8] J. Popovi´c, H. Hoppe, Progressive simplicial complexes, in: Proc. of SIGGRAPH, 1997, pp. 217–224.

[9] G. Taubin, A. Gu´eziec, W. Horn, F. Lazarus, Progressive forest split compression, in: Proc. of SIGGRAPH, 1998, pp. 123–132.

[10] R. Pajarola, J. Rossignac, Compressed progressive meshes, IEEE Transactions on Visualization and Computer Graphics 6 (2000) 79–93.

[11] Z. Karni, A. Bogomjakov, C. Gotsman, Efficient compression and rendering of multi-resolution meshes, in: Proc. of the conference on Visualization, 2002, pp. 347–354.

[12] J. Li, C.-C. J. Kuo, Progressive coding of 3-d graphic models, in: Proc. of the IEEE, Vol. 86, 1998.

[13] P. Alliez, M. Desbrun, Progressive compression for lossless transmission of triangle meshes, in: Proc. of SIGGRAPH, 2001, pp. 195–202.

[14] P.-M. Gandoin, O. Devillers, Progressive lossless compression of arbitrary simplicial complexes, in: Proc. of SIGGRAPH, 2002, pp. 372–379.

[15] A. Khodakovsky, P. Schr¨oder, W. Sweldens, Progressive geometry compression, in: Proc. of SIGGRAPH, 2000, pp. 271–278.

[16] A. Khodakovsky, I. Guskov, Compression of normal meshes, in: Geometric Modeling for Scientific Visualization, 2003, pp. 189–206

[17] Z. Karni, C. Gotsman, Spectral compression of mesh geometry, in: Proceedings of SIGGRAPH, 2000, pp. 279–286.

[18] K. Mamou, C. Dehais, F. Chaieb, F. Ghorbel, Shape approximation for efficient progressive mesh compression, in: Proc. of the IEEE International Conference on Image Processing, 2010.

[19] C. Touma, C. Gotsman, Triangle mesh compression, in: Graphics Interface 98 Conference Proc., 1998, pp. 26–34.

[20] J. Peng, C.-C. J. Kuo, Geometry-guided progressive lossless 3d mesh coding with octree decomposition, in: Proc. of SIGGRAPH, 2005, pp. 609–616.

[21] Maglo, Adrien, et al. "Progressive compression of manifold polygon meshes."Computers & Graphics 36.5 (2012): 349-359.

[22] Kim, Junho, Changwoo Nam, and Sungyul Choe. "Bayesian AD coder: mesh-aware valence coding for multiresolution meshes." Computers & Graphics 35.3 (2011): 713-718.

[23] Lee, Dae-Youn, Sanghoon Sull, and Chang-Su Kim. "Progressive 3D mesh compression using MOG-based Bayesian entropy coding and gradual prediction."The Visual Computer (2013): 1-15.

[24] J. Kim, S. Choe, S. Lee, Multiresolution random accessible mesh compression, Computer Graphics Forum 25 (3) (2006) 323–331.

[25] J. Kim, S. Lee, Truly selective refinement of progressive meshes, in: Proc. of Graphics interface, 2001, pp. 101–110.

[26] Choe, Sungyul, et al. "Mesh compression with random accessibility." The 5th Korea-Israel Bi-National Conference on Geometric Modeling and Computer Graphics. 2004.

[27] A. Khodakovsky, P. Alliez, M. Desbrun, and P. Schr¨oder. Near-Optimal Connectivity Encoding of 2-Manifold Polygon Meshes. Graphical Models 64(3-4), pages 147–168, 2002.

[28] M. Isenburg and P. Alliez. Compressing polygon mesh geometry with parallelogram prediction. In Proceedings of the IEEE Visualization 2002, pages 141–146, 2002.

[29] Choe, Sungyul, et al. "Random accessible mesh compression using mesh chartification." Visualization and Computer Graphics, IEEE Transactions on 15.1 (2009): 160-173.

[30] Maglo, Adrien, Ian Grimstead, and Céline Hudelot. "Cluster-based random accessible and progressive lossless compression of colored triangular meshes for interactive visualization." Computer Graphics International. Vol. 11. 2011.

[31] Lee, H., Lavou´ e, G., Dupont, F.: New methods for progressive compression of colored 3D Mesh. In: International Conference on Computer Graphics, Visualization and Computer Vision (WSCG) (2010)

[32] Yoon, Sung-eui, and Peter Lindstrom. "Random-accessible compressed triangle meshes." Visualization and Computer Graphics, IEEE Transactions on 13.6 (2007): 1536-1543.

[33] Courbet, Clement, and Celine Hudelot. "Random accessible hierarchical mesh compression for interactive visualization." Computer Graphics Forum. Vol. 28. No. 5. Blackwell Publishing Ltd, 2009.

[34] Jamin, Clément, Pierre-Marie Gandoin, and Samir Akkouche. "CHuMI viewer: Compressive huge mesh interactive viewer." Computers & Graphics 33.4 (2009): 542-553.

[35] Maglo, Adrien, Ian Grimstead, and CéLine Hudelot. "POMAR: Compression of progressive oriented meshes accessible randomly." Computers & Graphics 37.6 (2013): 743-752.

[36] Liang Hu, Pedro Sander, Hugues Hoppe. "Selective refinement of irregular mesh hierarchy using GPU streaming passes." *Symposium on Interactive 3D Graphics and Games 2009*, 169-176.