Virtual Reality & Intelligent Hardware

• Review •

October 2023 Vol. 5 No. 5: 395–406
10.1016/j.vrih.2020.02.002

Survey of lightweighting methods of huge 3D models for online Web3D visualization

Xiaojun LIU[1], Jinyuan JIA[2], Chang LIU[3*]

1. *Department of mathematics and Information Engineering, Nanhu College of Jiaxing University, Zhejiang* 314000*, China;*
2. *Department of Softwere Engineeging, Tongji University, Shanghai* 201804*, China;*
3. *School of Information Technology, Nanchang Hangkong University, Nanchang* 330063*, China*

Received 13 January 2020; **Revised** 1 February 2020; **Accepted** 12 February 2020

Abstract: Background With the rapid development of Web3D technologies, the online Web3D visualization, particularly for complex models or scenes, has been in a great demand. Owing to the major conflict between the Web3D system load and resource consumption in the processing of these huge models, the huge 3D model lightweighting methods for online Web3D visualization are reviewed in this paper. **Methods** By observing the geometry redundancy introduced by man-made operations in the modeling procedure, several categories of light-weighting related work that aim at reducing the amount of data and resource consumption are elaborated for Web3D visualization. **Results** By comparing perspectives, the characteristics of each method are summarized, and among the reviewed methods, the geometric redundancy removal that achieves the lightweight goal by detecting and removing the repeated components is an appropriate method for current online Web3D visualization. Meanwhile, the learning algorithm, still in improvement period at present, is our expected future research topic. **Conclusions** Various aspects should be considered in an efficient lightweight method for online Web3D visualization, such as characteristics of original data, combination or extension of existing methods, scheduling strategy, cache management, and rendering mechanism. Meanwhile, innovation methods, particularly the learning algorithm, are worth exploring.

Keywords: Huge 3D model; Lightweighting; Web3D; Visualization; Shape descriptor

Supported by the National Natural Science Foundation of China (U19A2063).

Citation: Xiaojun LIU, Jinyuan JIA, Chang LIU. Survey of lightweighting methods of huge 3D models for online Web3D visualization. Virtual Reality & Intelligent Hardware, 2023, 5(5): 395–406

1 Introduction

Currently, to access or interact with the world at home has become a popular lifestyle, and its great facility has stimulated enormous demands in every industry sector. With various related applications emerging, the Web3D-based systems that allow us to visualize, share, and interact with the 3D virtual environments through the internet have been the most popular due to their possibility of immersing, no installation required, and security of information. However, a significant amount of literature or research has addressed a few issues[1–4].

*Corresponding author, lcsszz@163.com

2096-5796/©Copyright 2023 Beijing Zhongke Journal Publishing Co. Ltd., Publishing services by Elsevier B.V. on behalf of KeAi Communication Co. Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by/4.0/).

The online Web3D visualization of 3D virtual models (or scenes) has become an imperative means of data accessing and sharing in the future, and a wide range of implements are available in online games, tourism, education, engineering, construction, etc. Meanwhile, there is a further increase in demand for more data type and heavier data volume. However, with the user demand for a more refined model and the increasing complexity of the model itself, the resource conflict between the model processing requirement and the capabilities of the Web3D system has evolved into a serious problem. Consequently, the online Web3D visualization has become a tough and challenging task. The huge 3D models (or scenes) are always man-made models developed using the modeling software such as 3ds Max and Solid Works. During the modeling procedure, many operations such as copy/paste and drag and drop can bring numerous repeated patterns into the final model based on translation, rotation, and scaling. Various research aiming at detecting and removing these redundancies based on repeated patterns have been conducted[5,6].

Based on our previous work, the lightweight approach that reduces the amount of data or its consumption for the visualization system using the information hiding and geometric slimming strategy[7] will be reviewed in this paper. Furthermore, considering the main resource limitation of Web3D browsers in the occupied cache buffer, network consumption, and online operation complexity, we will endeavor to present and summarize some of the most popular geometrically lightweighting work for online Web3D visualization. As far as we are concerned, this approach is more effective and functional in practical projects. Meanwhile, owing to our limited attempts and experiments, we believe that there still exist numerous other methods or potential solutions beyond the review presented in this paper.

The remainder of this paper is organized as follows. The popular lightweighting methods are firstly elaborated in section 2. The characteristics of the reviewed method are described in section 3 from a comparative perspective. The conclusion and future work are presented in section 4.

2 Lightweighting methods

The geometrically lightweighting method for processing geometry model, especially the geometric mesh here, has long existed. More specific phases such as “rigid” and “articulated” that represent the model feature have been introduced[8–11]. In this paper, considering the current operational capability of Web3D browsers, we pay more attention to the methods for rigid model and elaborate some of the most popular research: mesh compression, progressive mesh, geometric redundancy removal, and parametric representation, etc.

2.1 Mesh compression

Compression algorithm[12] is a classic method to reduce the size of data file using a coding strategy, and two reverse procedures are required in this algorithm. The earlier survey on mesh compression was introduced by Peng et al. in which this algorithm was classified into single and multiple resolution compression according to the mesh geometry characteristics[13]. In the compression algorithm, an encoding procedure converts raw data into a new one with a smaller amount, and a reverse decoding procedure is employed when raw data is needed to be recovered. The mesh compression method in this subsection is an extension of the compression algorithm in graphics area. The connectivity of vertices with the geometry of the mesh models is selected for the encoding procedure by constructing a code index through spanning tree[14], octree[15], or other data structure[16,17].

Therefore, the mesh compression method can reduce the mesh amount in conventional applications. However, due to its two procedures, encoding and decoding[15], it also requires more device resources that are fatal for the Web3D visualization systems. Meanwhile, these two procedures also raise a time delay problem that cannot be ignored.

2.2 Progressive mesh

Based on the coding strategy of mesh compression, an improved method named progressive mesh (PM) is proposed by Hoppe[18]. The major contribution of PM is that a new mesh simplification algorithm is proposed. With this simplification, an arbitrary mesh can be encoded into a simple base mesh M^0 along with a sequence of detail records based on a streaming idea. First, by applying a sequence of mesh transformation such as edge collapse (represented as *ecol* in Figure 1), the original mesh can be stored as a much coarser mesh, called base mesh M^0 along with a sequence of detail records. Second, by using a sequence of inverse mesh transformation such as vertex split (represented as *vsplit* in Figure 1), the sequence of detail records can then be used to incrementally refine M^0 back into the original mesh when required[18]. It can be observed from the figure that each detail record stores the information of a vertex split and an elementary mesh transformation, which can be used later to add split vertex back to the mesh.

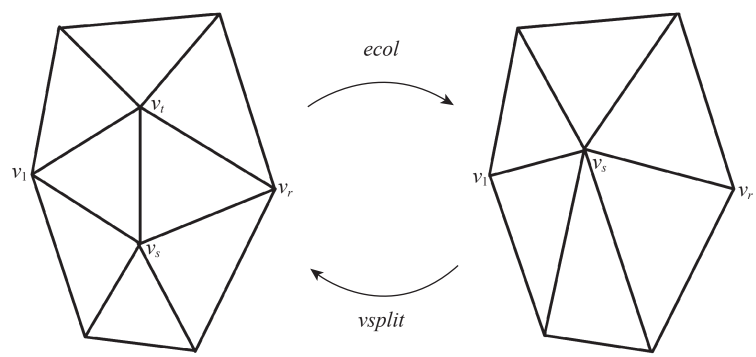


Figure 1 Illustration of edge collapse transformation[18].

Using PM, the complex original mesh can be stored as a simple base mesh along with the sequence of much smaller detail records, and when the mesh is requested in the visualization system, the simple base mesh with small data amount can be first preloaded. Next, as the interaction time increases, the detail records can be added to refine the base mesh progressively until the original mesh is recovered. The progressive lossless mesh representation addresses several practical problems in graphics, especially in the Web3D-based applications, such as smooth geomorphing with a minor time delay, progressive transmission for relatively few network resources, and mesh compression based on a streaming idea. However, we also found that the PM method, based on streaming idea, is more suitable to reduce the load amount at a certain moment. The details such as streaming velocity or interactive control have not been mentioned. Meanwhile, it also creates a new problem that the whole amount of data of base mesh, along with the sequence of detail records, is much larger than the original mesh. Increasing cache data with data loading is very likely to exceed the resource limitation of Web3D systems, resulting in a system crash.

With further improvements, Peng et al. introduced a more upgraded PM method: coding combined with lossless PM[19]. Zheng et al. proposed a non-redundant directed acyclic graph (DAG) hierarchy for view-dependent multiresolution mesh to browse online mesh models, and it enabled the model to be visualized as soon as the viewpoint moves without affecting the quality of user experience[20]. Zhao et al. presented a network congestion aware PM, named bandwidth adaptation PM, which focused on the preview streaming architecture and framework, and suggested five methods, stop-and-wait, reduce-speed, reduce-quality, key-view-aware, and adaptive-zoom, to handle bandwidth variation[21]. Meanwhile, the texture model supported by PM[22,23] combines progressive texture streaming with an adaptive number of refinement levels without pixel retransmission, and the geometric structure simplification-based PM[24] for building models enables us to address the challenges of building a model with various structures. As demand changes, further improvements are being made, and extended PM is being implemented in various industry sectors, particularly in Web3D-based applications.

2.3 Geometric redundancy removal

Based on the geometric redundancy introduced by repeated components, the lightweight goal of the geometric redundancy removal precisely relies on the repeated components in a geometric view. By directly locating and removing the geometric repeated components, the original model is lightweighted using only unique components left. Meanwhile, the geometric transformation between repeated and unique components is recorded to restore the original model and future application.

To detect the repeated components, different components should be first compared and evaluated under the same space to obtain their geometry similarity. Until now, an effective and efficient way for this is to employ a shape descriptor (SD) that can be viewed as a mapping because it can preserve as much information as possible, but with a result vector possibly in a low dimension. As a rapidly developing filed, several SDs are being reported, and the most up-to-date related reviews are provided in various research[9,10,25–28]. In this section, we will roughly present a comparison of existed SDs and their corresponding lightweighted methods in the following categories: 2D image-based method, shape distribution-based method, geometry feature-based method, spatial map-based method, graph-based method, and other recent methods.

2.3.1 2D image-based method

The 2D image-based method is a visual attention aware method in which the similarities between 3D models are obtained by measuring the visual similarities. The main idea of the 2D image-based method is that repeated 3D components appear similar from other viewpoints at different viewing angles. Therefore, hundreds of orthogonal projection images of a 3D component are always captured for descriptor construction. The image processing method, which is developed earlier than 3D technology, is then implemented to analyze the visual similarity of above images. One of the representative works[29–31] in this image processing is to encode both Zernike moments and Fourier descriptors to detect similar 3D components (Figure 2).

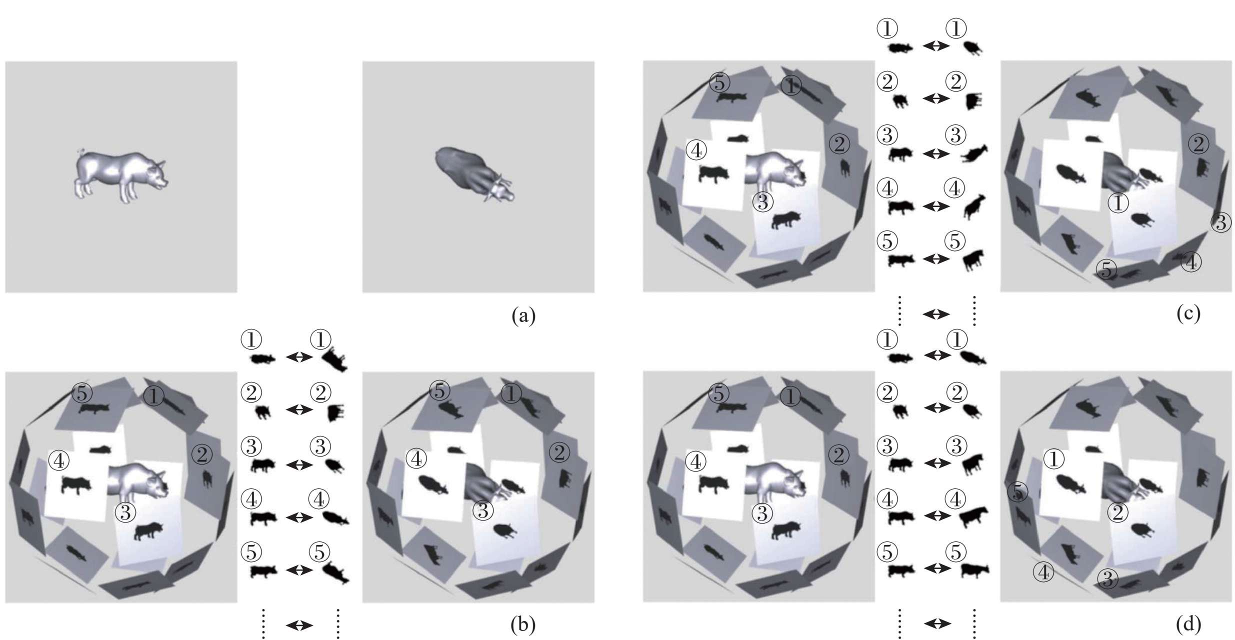


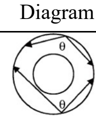
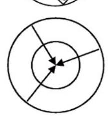
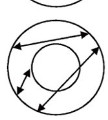
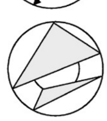
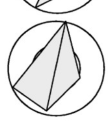
Figure 2 Comparing LightField descriptors of two 3D models[29].

2.3.2 Shape distribution-based method

The shape distribution-based method implemented by Osada et al. aims at computing 3D shape signatures and dissimilarity measures for arbitrary geometry model, including degenerate 3D polygonal models[32]. Using a shape function to measure global geometric properties of a model, a corresponding distribution histogram can be constructed for a geometry model. Furthermore, the difficult comparison between two geometry models is reduced to a simple comparison between two probability distributions. Meanwhile, several predefined shape distribution functions (Table 1) are provided in this paper; early experiences show that these functions are fast to compute, easy to understand, and invariant to rigid motions.

In addition to the above-mentioned studies, there are also other methods with a distribution idea[33]. For instance, Ankerst et al. used shells and sectors around the centroid of a 3D model as a basic space decomposition in which geometric statistical studies are conducted to describe shape distribution using discrete histograms[34]. Körtgen et al. proposed a distribution of sampled points, called the shape context, in which the relative positions and edge angles are summarized as a shape descriptor to measure shape similarity[35]. Akgül et al. also considered the distribution of point features that were represented in three functions, namely the radial shape function, the tangent plane function, and the cross-product function[36].

Table 1 Shape distribution functions[32]

Shape distribution function	Diagram	Function illustration
A3		Measures the angle between three random points on the surface of a 3D model
D1		Measures the distance between a fixed point and a random point on the surface; We use the centroid of the boundary of the model as the fixed point.
D2		Measures the distance between two random points on the surface
D3		Measures the square root of the area of the triangle between three random points on the surface
D4		Measures the cube root of the volume of the tetrahedron between four random points on the surface

2.3.3 Geometry feature-based method

The features utilized in the geometry feature-based method mainly include some basic properties such as position, normal and curvature, or some arithmetic combinational characteristics of these basic ones. A representative work in this category is presented by Mitra et al.[37], in which the geometry signature at each sample point p is computed using the principal curvatures, principal directions $c1$, $c2$, and normal vector $n = c1\times c2$. By estimating the translation between a point pair (p_i, p_j), as shown in Figure 3, a clustering procedure is then employed to map local surface patches onto each other to detect symmetry components.

Meanwhile, the weighted point set (WPS)[38] and heat kernel signature (HKS)[39] methods also employ feature vectors representing geometry properties of points. In WPS, the points with high gauss curvature values are selected to create its shape descriptor, and in HKS, the heat kernels of points are selected based on the concept of heat diffusion.

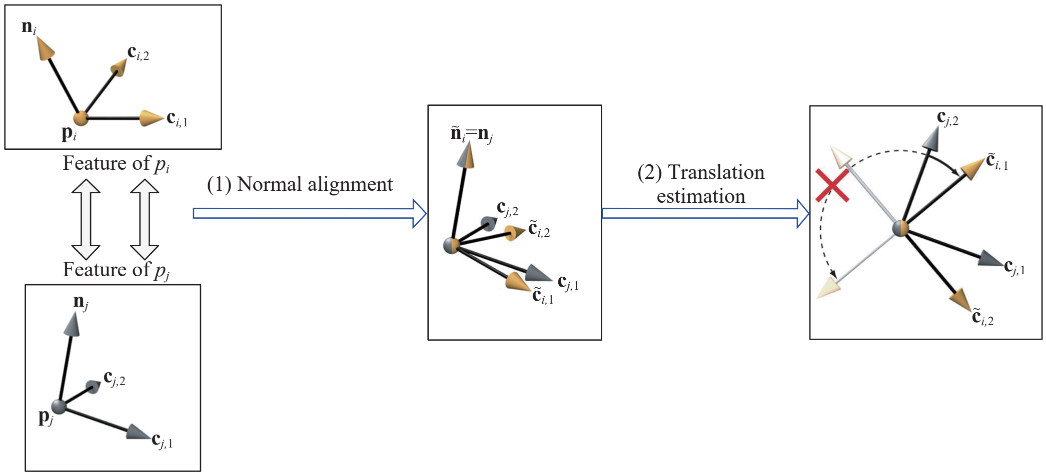


Figure 3 **Idea of translation estimation between a point pair (p_i, p_j)[37].**

2.3.4 Spatial map-based method

Spatial maps are the representations that capture the spatial location of geometry models. Generally, the distinctive features of this method are always within the frequency domain for a simple representation through which the primary characteristics are preserved simultaneously. Thus, the geometry data is first mapped from the spatial domain to the frequency domain.

The spherical harmonics implemented by Kazhdan et al. classified the model similarity using a harmonic shape representation computing idea (Figure 4)[40]. (1) A given geometry model was rasterized into a 64×64 ×64 voxel grid in which the boundary voxel was assigned to 1 and the others to 0; (2) The voxel grid was treated as a function defined in three-space and decomposed into 32 spherical functions by restricting radii 1 to 32; (3) Each sphere was decomposed as a sum of its first 16 harmonic components described in (4), and in (5), a 32×16 signature combined with above functions is obtained to define a computational shape representation. Similar to the spherical harmonic decomposition-based mapping, other spatial map-based methods also existed in 3D Fourier transform[41], spherical wavelet transform[42], etc., and performed well in their experiments.

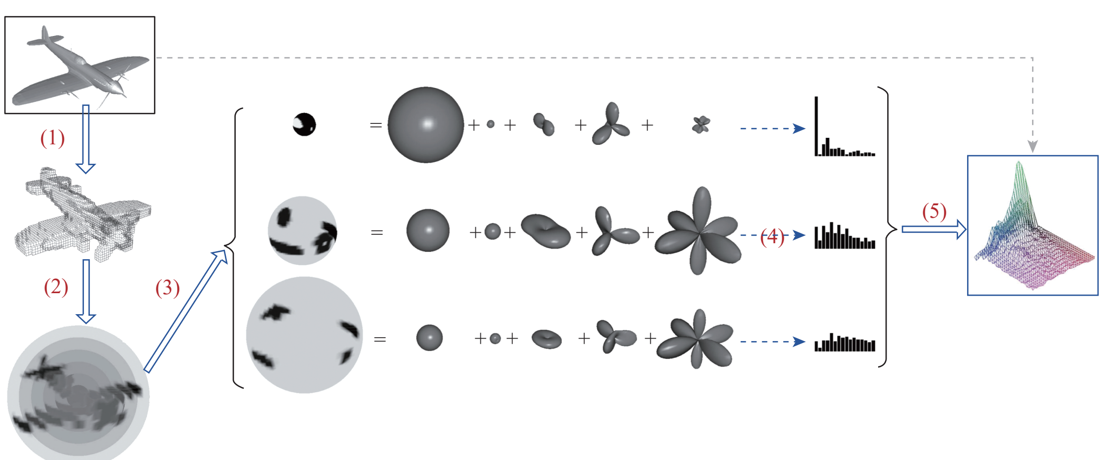


Figure 4 **Workflow of harmonic shape representation computing idea[40].**

2.3.5 Graph-based method

Graph-based method aims to achieve a geometric meaning, e.g., skeleton, topology, or other structure-oriented properties of the original model. The descriptors in this method are considered to be more complex. It can encode geometry data and other deduced topology properties using the graph nodes and edges, and reduces the problem of geometry similarity computing to graph comparison.

Skeleton graph is a common graph used for shape descriptors of geometry models, especially for articulated shapes. A skeleton graph[43] (Figure 5) is always obtained from a voxelized solid model and described using a set of the most representative voxels. The Reeb graph presented by Hilaga et al.[44] also represented a topological skeleton under a scalar function of a geometry model, and a series of cross sections of the model were employed to determine the nodes and arcs of the graph.

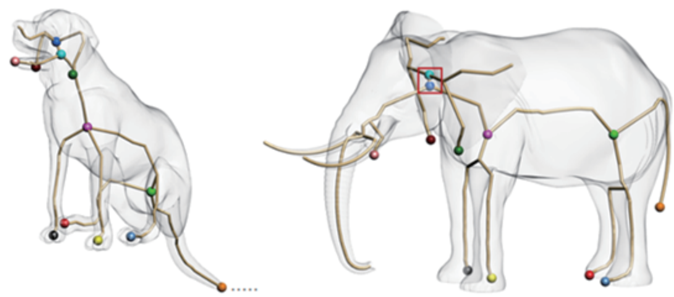
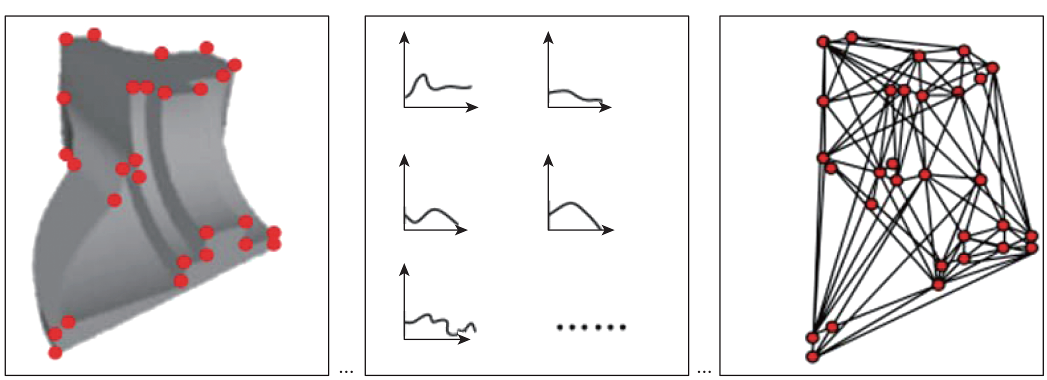


Figure 5 3D skeleton graph examples[43].

With further research, graphs with more information are designed for some special tasks or models[45,46]. The polynomial local shape descriptor proposed by Quan et al.[47] provided an interesting case in which the idea of the attributed relational graph (ARG) was implemented. Initially, many specified points were extracted as the graph nodes using the idea of minimum geodesic distance, and edges were added for every two nodes. Meanwhile, a piecewise polynomial of the geodesic iso-contours for the node point, called the local shape polynomial (LSP), is computed as the property of this node and the Euclidean distance of two nodes for the edge between them (Figure 6). The repeated model search is then realized using an ARG matching procedure.



(a) Interest points (b) Local shape polynomials (c) ARG

Figure 6 Polynomial local shape descriptor of interest points[47].

2.4 Parametric representation

Mesh parameterization is a powerful geometry processing method used in numerous computer graphics applications[48]. In the current industrial design field, it is widely utilized in geometry modeling. An appropriate parametric representation of geometry mesh can greatly reduce the amount of model data. However, despite of many extended parametric surfaces, such as explicit surfaces[49,50], the traditional mesh parameterization is only applicable for some special features, such as primitive surface and isometric mesh[51], due to the complexity of the model in practical cases.

Another method based on parameterization is procedural modeling[52,53] in which a special language, named computer generated architecture (CGA), and language grammar were designed to describe the modeling procedure. With the recorded modeling parameter, the resulting geometry model in CGA language is more analogous to a computer program. Meanwhile, we can create many different geometry models using the same model in CGA language purely by modifying its parameters. In other words, a CGA model can generate several different geometry models that represent an obvious lightweight idea. Owing to its background, the CGA model is more implemented to describe building models. However, other than CGA, there also exists another notable language, named L-system[54], which can be employed to construct tree models.

2.5 Others

In addition to the above-mentioned methods, there also exist many other geometry model lightweighting methods such as semantic information involved[55], prior knowledge driven[56], mesh segmentation preprocessed[57–60], and shape part-based[61]. These methods always make it possible to make the most of model information for feature extraction. For instance, the mesh segmentation preprocessed method tends to be sensitive to small perturbations in the model, and the model segment can always be treated as a shape part (or component) to detect repeated components. Moreover, with the passage of time, new types of methods based on some innovation approach have also appeared such as hybrid feature and leaning based methods. For the content-based 3D shape retrieval, Tangelder et al.[30] constructed a hybrid feature and 3D geometry learning[58–60,62–64] methods, which are still improving being a recent hot research trend, and have also been extended for 3D geometry model processing. Our previous work[1,65] also concentrated on some hybrid features to build models, particularly for building information modeling (BIM) models for Web3D visualization.

3 Results and comparisons

As elaborated above, the huge 3D model lightweighting methods for online Web3D visualization aim to reduce the resource consumption without or with a small amount of impact on the quality of user experience. Generally, each method reviewed above has its own characteristics, particularly for online Web3D visualization (Table 2).

Table 2 Comparison of huge 3D model lightweighting methods

Category of methods	Occupied cache buffer	Network consumption	Online operation	Operation challenge	Time delay for first sight
Mesh compression	★★☆☆☆	★★☆☆☆	Compression decoding	★★★★★	★★★★★
Progressive mesh	★★★★☆	★★★☆☆	Mesh refinement	★☆☆☆☆	★☆☆☆☆
Geometric redundancy removal	★★★☆☆	★★☆☆☆	Instanced rendering	★☆☆☆☆	★★☆☆☆
Parametric representation	★☆☆☆☆	★☆☆☆☆	Parametric parsing	★★★★☆	★★★☆☆

In comparison, the mesh compression method that can directly encode the original model without adding additional auxiliary data makes it possible to reduce the amount of data at a high level, resulting in an effective transmission and data caching. However, data that have been received online need additional decoding process before online Web3D visualization, and many unstable events, such as lack of resources, serious time delays, and even system crash, can occur.

The biggest advantage of the PM method is the promptness for the first sight because a coarse geometry model, called base mesh, can be transmitted first. However, in the subsequent streaming transmission, the vertex split containing auxiliary data should also be transmitted to restore the original model. With all these data accumulated to a certain amount, the amount of the transmitted data also exceeds, resulting in an overload risk to the web browsers. While dealing with the vertex split packet loss and efficient stream control, procedures such as view-dependent approach should be comprehensively considered.

In geometric redundancy removal, numerous lightweight methods are proposed based on the geometry characteristics of a model. Each repeated component is substituted with a 4×4 transform matrix from the unique component. Meanwhile, the instanced rendering function of rendering engine could be directly called to save resource using the transform matrix and unique component. Therefore, from our perspectives, despite difficulties in finding appropriate methods based on the geometry characteristics of a model, the geometric redundancy removal is an effective lightweight method for online Web3D visualization.

For parametric representation, although work on CGA and L-system has been implemented in some fields, the automatic inverse parametric procedure for different types of geometry models still possesses a great challenge. Moreover, online parametric parsing, which is resource consuming, will be required.

The learning algorithm in other methods, which is still in the improvement period, is our expected future research for geometry model lightweighting. In an era of artificial intelligence, we believe that it can automatically recognize redundancies efficiently and provide us with adequate information for the geometry model lightweighting.

4 Conclusion and future work

With the increasing volume of geometry models or scenes for online Web3D visualization, several popular lightweighting methods to reduce the amount of data have been reviewed in this paper. To relieve the conflict between the resource consumption of geometry model visualization and the resource limitation of web browsers, we mainly categorized the lightweighting-related research into mesh compression, progressive mesh, geometric redundancy removal, and parametric representation. Meanwhile, the geometric redundancy removal is illustrated in five methods. Based on the characteristics of geometry redundancy due to man-made operations in the modeling procedure, it is summarized that the geometric redundancy removal that achieves the lightweight goal by detecting and removing the repeated components is an appropriate method for current online Web3D visualization. Meanwhile, the learning algorithm that is being applied in more practical fields is our future research topic for lightweighting.

With this comparison and summary, we conclude that various aspects should be considered in an efficient lightweight method for online Web3D visualization, including the characteristics of the original data, combination or extension of existing methods, scheduling strategy, cache management, and rendering mechanism. Moreover, some new advanced innovation methods, particularly the learning algorithm, are worth exploring.

Declaration of competing interest

We declare that we have no conflict of interest.

References

1 Liu X, Xie N, Tang K, Jia J. Lightweighting for Web3D visualization of large-scale BIM scenes in real-time. Graphical Models, 2016, 88: 40–56
DOI: 10.1016/j.gmod.2016.06.001

2 Rautenhaus M, Bottinger M, Siemen S, Hoffman R, Kirby R M, Mirzargar M, Rober N, Westermann R. Visualization in meteorology—a survey of techniques and tools for data analysis tasks. IEEE Transactions on Visualization and Computer Graphics, 2018, 24(12): 3268–3296
DOI: 10.1109/tvcg.2017.2779501

3 Violante M G, Vezzetti E. Design of web-based interactive 3D concept maps: A preliminary study for an engineering drawing course. Computer Applications in Engineering Education, 2015, 23(3): 403–411

DOI: 10.1002/cae.21610
4 Violante M G, Vezzetti E. Design and implementation of 3D Web-based interactive medical devices for educational purposes. International Journal on Interactive Design and Manufacturing (IJIDeM), 2017, 11(1): 31–44
DOI: 10.1007/s12008-015-0277-0
5 Sipiran I, Gregor R, Schreck T. Approximate symmetry detection in partial 3D meshes. Computer Graphics Forum, 2014, 33(7): 131–140
DOI: 10.1111/cgf.12481
6 Cai K Y, Wang W C, Chen Z B, Chen Q Q, Teng J. Exploiting repeated patterns for efficient compression of massive models. In: Proceedings of the 8th International Conference on Virtual Reality Continuum and its Applications in Industry. Yokohama, Japan, ACM Press, 2009, 145–150
DOI:10.1145/1670252.1670283
7 Liu W, Niu Q, Zhou X H, Zhang X B. SCIENTIA SINICA Informationis, 2014, 44(11): 1422–1431
DOI: 10.1360/N112014-00201
8 Tang M, Kim Y J. Interactive generalized penetration depth computation for rigid and articulated models using object norm. ACM Transactions on Graphics, 2014, 33(1): 1–15
DOI: 10.1145/2517108
9 Kazmi I K, You L H, Zhang J J. A survey of 2D and 3D shape descriptors. In: 2013 10th International Conference Computer Graphics, Imaging and Visualization. Los Alamitos, CA, USA, IEEE, 2013, 1–10
DOI:10.1109/cgiv.2013.11
10 Zhang L, da Fonseca M J, Ferreira A. Survey on 3D shape descriptors. Technical Report, DecorAR, 2007, 3
11 Tam G K L, Zhi-Quan Cheng G K L, Yu-Kun Lai G K L, Langbein F C, Yonghuai Liu F C, Marshall D, Martin R R, Xian-Fang Sun R R, Rosin P L. Registration of 3D point clouds and meshes: a survey from rigid to nonrigid. IEEE Transactions on Visualization and Computer Graphics, 2013, 19(7): 1199–1217
DOI: 10.1109/tvcg.2012.310
12 Le Gall D J. The MPEG video compression algorithm. Signal Processing: Image Communication, 1992, 4(2): 129–140
DOI: 10.1016/0923-5965(92)90019-c
13 Peng J, Kim C S, Jay Kuo C C. Technologies for 3D mesh compression: A survey. Journal of Visual Communication and Image Representation, 2005, 16(6): 688–733
DOI: 10.1016/j.jvcir.2005.03.001
14 Aviles M, Moran F. Static 3D triangle mesh compression overview. In: 2008 15th IEEE International Conference on Image Processing. San Diego, CA, USA, IEEE, 2008, 2684–2687
DOI: 10.1109/icip.2008.4712347
15 Peng J, Kuo C C J. Geometry-guided progressive lossless 3D mesh coding with octree (OT) decomposition. ACM Transactions on Graphics, 2005, 24(3): 609–616
DOI: 10.1145/1073204.1073237
16 Maglo A, Lavoué G, Dupont F, Hudelot C. 3D mesh compression. ACM Computing Surveys, 2015, 47(3): 1–41
DOI: 10.1145/2693443
17 Pavez E, Chou P A, de Queiroz R L, Ortega A. Dynamic polygon clouds: representation and compression for VR/AR. APSIPA Transactions on Signal and Information Processing, 2018, 7: e15
DOI: 10.1017/atsip.2018.15
18 Hoppe H. Progressive meshes. In: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques. New York, USA, ACM Press, 1996, 99–108
DOI:10.1145/237170.237216
19 Peng J, Huang Y, Kuo C C J, Eckstein I, Gopi M. Feature Oriented Progressive Lossless Mesh Coding. Computer Graphics Forum, 2010, 29(7): 2029–2038
DOI: 10.1111/j.1467-8659.2010.01789.x
20 Zheng Z, Chan T K Y. View-dependent progressive mesh using non-redundant DAG hierarchy. In: Proceedings of the 3rd international conference on Computer graphics and interactive techniques in Australasia and South East Asia. Dunedin, New Zealand, ACM Press, 2005
DOI:10.1145/1101389.1101470
21 Zhao S H, Ooi W T, Carlier A, Morin G, Charvillat V. Bandwidth adaptation for 3D mesh preview streaming. ACM Transactions on Multimedia Computing, Communications, and Applications, 2014, 10(1s): 1–20
DOI: 10.1145/2537854
22 Lim N K, Kim D Y, Lee H. Interactive progressive image transmission for realtime applications. IEEE Transactions on Consumer Electronics, 2010, 56(4): 2438–2444
DOI: 10.1109/tce.2010.5681125
23 Englert M, Jung Y, Klomann M, Etzold J, Grimm P. Instant texture transmission using bandwidth-optimized progressive interlacing images. In: Proceedings of the Nineteenth International ACM Conference on 3D Web Technologies. Vancouver, British Columbia, Canada, ACM Press, 2014, 144

DOI:10.1145/2628588.2628606

24 Li Q, Sun X, Yang B, Jiang S. Geometric structure simplification of 3D building models. ISPRS Journal of Photogrammetry and Remote Sensing, 2013, 84: 100–113
DOI: 10.1016/j.isprsjprs.2013.07.006

25 Lara López G, Peña Pérez Negrón A, De Antonio Jiménez A, Ramírez Rodríguez J, Imbert Paredes R. Comparative analysis of shape descriptors for 3D objects. Multimedia Tools and Applications, 2017, 76(5): 6993–7040
DOI: 10.1007/s11042-016-3330-5

26 Narottambhai M, Tandel P. A survey on feature extraction techniques for shape based object recognition. International Journal of Computer Applications, 2016, 137(6): 16–20
DOI: 10.5120/ijca2016908782

27 Hamad M, Thomassey S, Bruniaux P. A new sizing system based on 3D shape descriptor for morphology clustering. Computers & Industrial Engineering, 2017, 11: 3683–692
DOI: 10.1016/j.cie.2017.05.030

28 Biasotti S, Moscoso Thompson E, Barthe L, Berretti S, Giachetti A, Lejemble T, Mellado N, Moustakas K, Manolas I, Dimou D. Recognition of geometric patterns over 3D models. In: Proceedings of the 11th Eurographics Workshop on 3D Object Retrieval. Eurographics Association, 2018, 71–77
DOI: 10.2312/3dor.20181055

29 Chen D Y, Tian X P, Shen Y T, Ouhyoung M. On Visual Similarity Based 3D Model Retrieval. Computer Graphics Forum, 2003, 22(3): 223–232
DOI: 10.1111/1467-8659.00669

30 Tangelder J W H, Veltkamp R C. A survey of content based 3D shape retrieval methods. Multimedia Tools and Applications, 2008, 39(3): 441–471
DOI: 10.1007/s11042-007-0181-0

31 Eitz M, Richter R, Boubekeur T, Hildebrand K, Alexa M. Sketch-based shape retrieval. ACM Transactions on Graphics, 2012, 31(4): 1–10
DOI: 10.1145/2185520.2185527

32 Osada R, Funkhouser T, Chazelle B, Dobkin D. Shape distributions. ACM Transactions on Graphics, 2002, 21(4): 807–832
DOI: 10.1145/571647.571648

33 Zhang T, Peng Q. Shape Distribution-Based 3D Shape Retrieval Methods: Review and Evaluation. Computer-Aided Design and Applications, 2009, 6(5): 721–735
DOI: 10.3722/cadaps.2009.721-735

34 Ankerst M, Kastenmüller G, Kriegel H P, Seidl T. 3D shape histograms for similarity search and classification in spatial databases// Advances in Spatial Databases. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, 207–226
DOI: 10.1007/3-540-48482-5_14

35 Körtgen M, Park G J, Novotni M, Reinhard K. 3D shape matching with 3D shape contexts. In: The 7th central European seminar on computer graphics. Budmerice, Slovakia, 2003, 3: 5–17

36 Akgül C B, Sankur B, Yemez Y, Schmitt F. 3D model retrieval using probability density-based shape descriptors. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2009, 31(6): 1117–1133
DOI: 10.1109/tpami.2009.25

37 Mitra N J, Guibas L J, Pauly M. Partial and approximate symmetry detection for 3D geometry. ACM Transactions on Graphics, 2006, 25 (3): 560–568
DOI: 10.1145/1141911.1141924

38 Tangelder J W H, Veltkamp R C. Polyhedral model retrieval using weighted point sets. International Journal of Image and Graphics, 2003, 03(1): 209–229
DOI: 10.1142/s021946780300097x

39 Sun J, Ovsjanikov M, Guibas L. A concise and provably informative multi-scale signature based on heat diffusion. Computer Graphics Forum, 2009, 28(5): 1383–1392
DOI: 10.1111/j.1467-8659.2009.01515.x

40 Kazhdan M, Funkhouser T. Harmonic 3D shape matching. In: ACM SIGGRAPH 2002 conference abstracts and applications. San Antonio, Texas, ACM Press, 2002, 191
DOI:10.1145/1242073.1242204

41 Vranic D, Saupe D. 3D shape descriptor based on 3D Fourier transform. In: Proceedings of the EURASIP Conference on Digital Signal Processing for Multimedia Communications and Services. 2001, 271–274

42 Laga H, Takahashi H, Nakajima M. Spherical wavelet descriptors for content-based 3D model retrieval. In: IEEE International Conference on Shape Modeling and Applications 2006 Matsushima, Japan, IEEE, 2006
DOI:10.1109/smi.2006.39

43 Kin-Chung Au O, Tai C L, Cohen-Or D, Zheng Y, Fu H. Electors Voting for Fast Automatic Shape Correspondence. Computer Graphics Forum, 2010, 29(2): 645–654

DOI: 10.1111/j.1467-8659.2009.01634.x
44 Hilaga M, Shinagawa Y, Kohmura T, Kunii T L. Topology matching for fully automatic similarity estimation of 3D shapes. In: Proceedings of the 28th annual conference on Computer graphics and interactive techniques. New York, USA, ACM Press, 2001, 203–212
DOI: 10.1145/383259.383282
45 Mitra N, Wand M, Zhang H R, Cohen-Or D, Kim V, Huang Q X. Structure-aware shape processing. In: SIGGRAPH Asia 2013 Courses on-SA '13. Hong Kong, China. New York, USA, ACM Press, 2013
DOI:10.1145/2542266.2542267
46 Balashova E, Singh V, Wang J P, Teixeira B, Chen T, Funkhouser T. Structure-aware shape synthesis. In: 2018 International Conference on 3D Vision (3DV). Verona, IEEE, 2018, 140–149
DOI:10.1109/3dv.2018.00026
47 Quan L, Tang K. Polynomial local shape descriptor on interest points for 3D part-in-whole matching. Computer-Aided Design, 2015, 59: 119–139
DOI: 10.1016/j.cad.2014.09.005
48 Hormann K, Bruno Levy, Sheffer A. Mesh parameterization: theory and practice. ACM Siggraph Asia Courses, 2008
DOI: 10.1145/1508044.1508091
49 Li Q, Tian J. Partial shape-preserving splines. Computer-Aided Design, 2011, 43(4): 394–409
DOI: 10.1016/j.cad.2011.01.007
50 Hong Q Q, Li Q D, Tian J. Implicit reconstruction of vasculatures using bivariate piecewise algebraic splines. IEEE Transactions on Medical Imaging, 2012, 31(3): 543–553
DOI:10.1109/tmi.2011.2172455
51 Pietroni N, Tarini M, Cignoni P. Almost isometric mesh parameterization through abstract domains. IEEE Transactions on Visualization and Computer Graphics, 2010, 16(4): 621–635
DOI: 10.1109/tvcg.2009.96
52 Chen G N, Esch G, Wonka P, Müller P, Zhang E. Interactive procedural street modeling. In: ACM SIGGRAPH 2008 papers on-SIGGRAPH '08. Los Angeles, California, ACM Press, 2008, 27(3): 103
DOI:10.1145/1399504.1360702
53 Kelly T, Wonka P. Interactive architectural modeling with procedural extrusions. ACM Transactions on Graphics, 2011, 30(2): 1–15
DOI: 10.1145/1944846.1944854
54 Šťava O, Beneš B, Měch R, Aliaga D G, Krištof P. Inverse procedural modeling by automatic generation of l-systems. Computer Graphics Forum, 2010, 29(2): 665–674
DOI: 10.1111/j.1467-8659.2009.01636.x
55 Tasse F P, Dodgson N. Shape2vec. ACM Transactions on Graphics, 2016, 35(6): 1–12
DOI: 10.1145/2980179.2980253
56 van Kaick O, Tagliasacchi A, Sidi O, Zhang H, Cohen-Or D, Wolf L, Hamarneh G. Prior knowledge for part correspondence. Computer Graphics Forum, 2011, 30(2): 553–562
DOI: 10.1111/j.1467-8659.2011.01893.x
57 Golovinskiy A, Funkhouser T. Consistent segmentation of 3D models. Computers & Graphics, 2009, 33(3): 262–269
58 Lv J, Chen X, Huang J, Bao H. Semi-supervised mesh segmentation and labeling. Computer Graphics Forum, 2012, 31(7): 2241–2248
DOI: 10.1111/j.1467-8659.2012.03217.x
59 Huang L, Yang J, Tian Z. Unsupervised co-segmentation of 3D shapes via functional maps. Journal of Physics: Conference Series, 2019, 1168
DOI: 10.1088/1742-6596/1168/3/032046
60 George D, Xie X, Lai Y K, Tam G. A deep learning driven active framework for segmentation of large 3D shape collections. 2018
61 Theologou P, Pratikakis I, Theoharis T. A review on 3D object retrieval methodologies using a part-based representation. Computer-Aided Design and Applications, 2014, 11(6): 670–684
DOI: 10.1080/16864360.2014.914385
62 Fang Y, Xie J, Dai G X, Wang M, Zhu F, Xu T T, Wong E. 3D deep shape descriptor. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Boston, MA, USA, IEEE, 2015, 2319–2328
DOI:10.1109/cvpr.2015.7298845
63 Xie J, Zhu F, Dai G, Shao L, Fang Y. Progressive shape-distribution-encoder for learning 3d shape representation. IEEE Transactions on Image Processing, 2017, 26(3): 1231–1242
DOI: 10.1109/tip.2017.2651408
64 Bronstein M M, Bruna J, LeCun Y, Szlam A, Vandergheynst P. Geometric deep learning: going beyond euclidean data. IEEE Signal Processing Magazine, 2017, 34(4): 18–42
DOI: 10.1109/msp.2017.2693418
65 Liu X, Jia J. Mobile web-based lightweight and real-time roaming algorithm for large-scale WebBIM scenes. SCIENTIA SINICA Informationis, 2018, 48(3): 274–292
DOI: 10.1360/N112017-00240