A 3D Visualization Method of Global Ocean Surface Based on Discrete Global Grids

Shuxiang Wang[1,2], Kaixiang Wen[1,3](✉), Li Liu[4], Chen Zhang[1],
and You Li[5]

[1] Guangdong Key Laboratory of Geospatial Information Technology and Application, Guangzhou Institute of Geography, Guangzhou 510070, China
`690612731@qq.com`
[2] Shandong University of Technology, Zibo 255000, China
[3] Guangdong University of Technology, Guangzhou 510000, China
[4] Ludong University, Yantai 264001, China
[5] State Grid Hunan Electric Power Limited Company Maintenance Company, Changsha 410004, China

Abstract. Ocean surface 3D visualization is an important aspect in marine information research and is of great significance to the construction of digital ocean. Due to the limitation of rendering efficiency and loading speed, the existing 3D visualization of ocean surface cannot meet the construction of ocean surface environment in a large area. Therefore, how to realize the global 3D visualization of the ocean surface is one of the key issues to be solved. For this, the paper proposes a 3D visualization method based on discrete global grids. Firstly, a multi-scale grid model for GPU based on the existing global discrete grid model is designed. Secondly, a dynamic wave simulation method based on GPU shader technology is proposed in order to realize the dynamic wave simulation. Finally, the effectiveness of the method is verified by comparative experiments. The experimental results show that the method of the paper achieves the effect of pixel-level ocean-land boundary and supports the dynamic simulation and real-time updating of the wave driven by the wind field. Compared with the existing 3D visualization method of ocean surface, the loading speed is faster and the rendering efficiency is higher. Therefore, the method is more suitable for the application needs of environment construction of the global ocean surface.

Keywords: 3D visualization · Ocean surface · Global discrete grid · Wave rendering

1 Introduction

The ocean covers 70% of the global surface and is an important part of the geographical environment. Ocean surface visualization uses computer graphics to reproduce the real ocean surface environment. It is of great significance to Digital Ocean. With the development of computer graphics technology, the traditional ocean surface visualization based on 2D graphics system has been unable to meet need of visualization and application due to problems such as abstract expression and lack of sense of

Y. Xie et al. (Eds.): GSES 2019/GeoAI 2019, CCIS 1228, pp. 19–32, 2020.
https://doi.org/10.1007/978-981-15-6106-1_2

reality. Ocean surface visualization based on 3D graphics technology has become the mainstream of research of ocean surface visualization. However, concerning efficiency, the existing 3D visualization method of ocean surface is mainly constructed for a small-scale regional ocean environment. The limitations of spatial scale and scope limit the development of 3D visualization systems of ocean surface. Therefore, how to build an ocean surface environment of global scale and realize global 3D visualization of the ocean surface has become a key issue to be solved.

Aiming at the above problems, this paper proposes a method for 3D visualization of ocean surface based on global discrete grids. The method includes two parts: First, in order to effectively organize and manage the global ocean surface grid, and make full use the ability of GPU parallel computing to improve the efficiency of ocean surface rendering, a multi-scale grid model of ocean surface for GPU rendering is designed, based on the existing global discrete grid model. Secondly, based on the tessendorf model, a method of wave dynamic simulation with GPU shader technology as the core is proposed in order to realize the function of dynamic simulation and real-time update of ocean wave driven by wind field. At the end of the paper, the effectiveness of the method is verified by comparative experiments.

2 Related Works

The existing research on 3D visualization of ocean surface mainly focuses on three aspects: ocean surface grid organization, wave dynamic simulation and the optimization of ocean surface rendering efficiency.

2.1 Ocean Surface Grid Organization

In order to realize the visualization of the ocean surface by computer, it is necessary to abstract the real ocean surface into an ocean surface grid. For large-scale ocean surface visualization, it is essential to efficiently organize and manage the ocean surface grid. Now, the grid organization method of ocean surface based on the projection grid model is the most widely used. This method realizes the infinite extension effect of the ocean surface by constructing an ocean surface grid in the projection space so that the ocean surface grid can automatically move and zoom with the angle of view. However, since all ocean surfaces are represented by a grid, it is difficult to reflect the characteristics of different ocean areas (such as boundaries of land and ocean, etc.) [1, 2]. In addition, there is a method of ocean surface grid organization based on the LOD grid model. This method does not load all ocean surface into the memory at one time but dynamically performs the loading and unloading of the ocean surface grid according to the viewpoint [3, 4]. However, since the ocean surface in different regions is represented by different grids, the actual application requires grid splicing in real time and the ocean surface rendering efficiency is inevitably affected during the splicing process [4–6].

2.2 Wave Dynamic Simulation

The ocean wave simulation simulates the ocean surface spray by modifying the vertex position of the ocean surface grids. The sense of reality, instantaneity and interactivity directly affect the realistic degree of the ocean surface scene. The wave dynamic simulation is relatively mature. The research in this aspect can be traced back to the 1980s. Reeves, Bailey et al. proposes a method of wave simulation based on the Gerstner model [7, 8]. According to the parameters of the orbit, the method can help to produce realistic wave shapes and subdivide waves of different characteristics through ray tracing and adaptive methods. Later, He and Gary improves it on the basis of this method [9, 10]. Through the A buffer technique, different waveforms are generated according to the steepness and depth of the waves, and the simulation and rendering of the broken waves on the inclined beach are realized. In addition, Paul, Chen et al. proposes a real-time method of wave simulation based on texture mapping [11, 12]. Through random analog sine, wave shape is simulated and mapped, the velocity of wave simulation is faster. Tessendorf and Schlitzer proposes a method of wave simulation based on ocean wave spectrum [13, 14]. The method simulates height of the wave by 2D fast Fourier inverse transform, and simulates the undulating feature of the wave in vertical direction by the form of a height field sequence. And the dynamic simulation of ocean surface spray is simulated. Due to operability and efficiency, the Tessendorf method is the most widely used.

2.3 The Optimization of Ocean Surface Rendering Efficiency

Low ocean surface rendering efficiency can seriously affect the user's visual experience. Especially when performing large-scale ocean surface visualization the ocean surface rendering efficiency directly determines the availability of ocean surface visualization methods. In recent years, scholars have begun a lot of research on how to improve the efficiency of ocean surface rendering. At present, there are mainly two ways to improve the efficiency of ocean surface rendering. One way is to update in parallel the position of the ocean grid point based on the GPU shader which loads the offset value of the wave motion into the GPU for the parallel sampling by the shader in a displacement texture so as to reduce CPU load and improve ocean surface rendering efficiency [15]. Another way is to build an ocean surface visualization system based on the CUDA (Compute Unified Device Architecture). For example, Su et al. achieves parallel reading of ocean surface data based on the CUDA architecture [16]. Zhang et al. realizes the parallel calculation of wave shape based on the CUDA architecture [17]. To some extent, the above research has solved the problem of ocean surface rendering efficiency. If the ocean surface grid is complicated to construct or the memory occupancy is large, it will still affect the overall ocean surface rendering efficiency [18].

In summary, in order to realize the 3D visualization of the ocean surface facing the world, it is necessary to solve the problems of ocean surface grid organization, wave dynamic simulation and ocean surface rendering efficiency optimization. For this, this paper proposes a 3D visualization method of ocean surface based on global discrete grid which organizes the global ocean surface through a global discrete grid model, and uses the GPU shader in combination with the method of wave simulation named Tessendorf to achieve efficient rendering of ocean surface waves.

3 Methods

3.1 Global Multi-scale Ocean Surface Grid Model for GPU

In order to effectively organize and manage the global ocean surface grid and make full use of the GPU parallel computing ability to improve the ocean surface rendering efficiency, this paper explores on the basis of the global longitude and latitude discrete grid model and designs a global multi-scale ocean surface grid model of GPU-oriented rendering. The multi-scale ocean surface grid model is shown in Fig. 1:

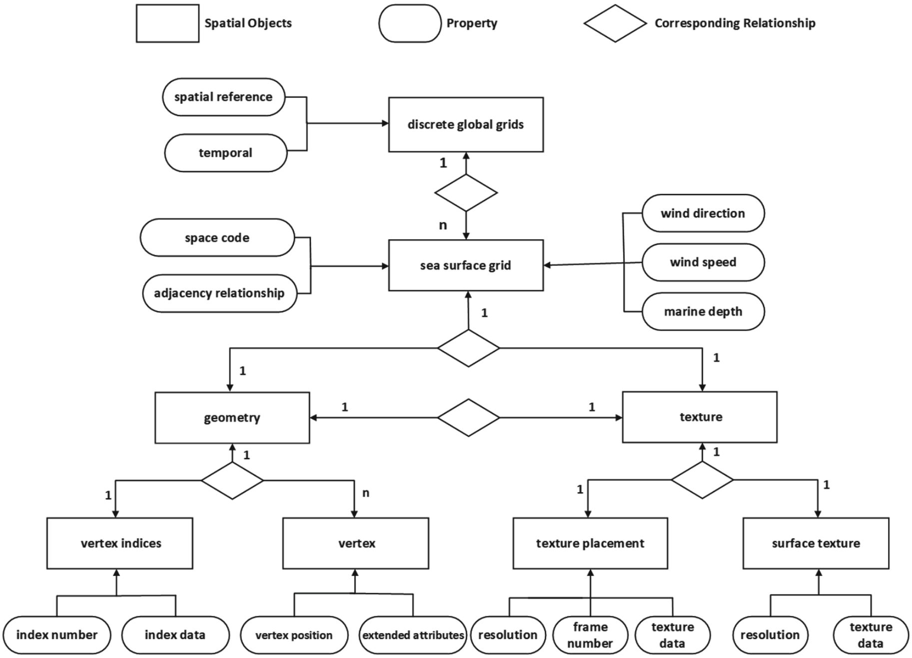


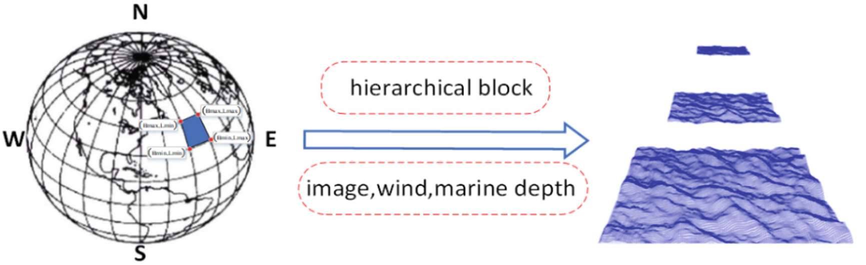
Fig. 1. GPU-oriented multi-scale ocean surface grid model

The ocean surface grid is the basic unit of ocean surface data organization and rendering. It is organized and constructed according to the rules of global latitude and longitude grid splitting. Thus, unique spatial code of each ocean surface grid corresponds to a global discrete grid, which records the number of stages the row number and the column number of the current grid. Spatial coding can be used to retrieve information such as wind speed, wind direction and ocean depth, ocean area distribution, and the adjacency relationship between the grids. The geometric object of the ocean surface grid describes the geometric information of the ocean surface, which consists of vertex and vertex indices. The vertex is the grid point, which is considered as the minimum unit of ocean level fluctuation. In addition to the basic position information, in order to realize the ocean surface dynamic simulation based on GPU, the vertex attribute is extended. Other attributes including latitude and longitude and vertex type are added. The vertex index records the organization relationship among discrete vertices for the transformation of point-to-face primitives.

In order to realize the dynamic simulation of the ocean wave, the texture object of the ocean surface grid is divided into displacement texture and surface texture. The displacement texture records the dynamic fluctuation process of the ocean surface wave. The current wind direction and wind speed are used as parameters and then generated based on the ocean wave modeling algorithm. Finally, it is loaded into the GPU in the form of 3D texture. The surface texture records the color and transparency of the current ocean surface grid, which can be generated by seawater depth or mask data of the recorded ocean area. The depth of the color can reflect the water body shading effect caused by the depth of the ocean. In addition to reflecting the texture of the water, the degree of transparency can also be used to hide the epicontinental water to achieve the effect of ocean and land boundary.

The advantage of this model structure is that the geometry of the ocean surface grid and its specific morphological features (recorded in the displacement texture) are separated from each other and only correlated by the GPU when rendering. When the ocean surface grid is first loaded, only the most basic geometry structure needs to be constructed, and the more complex morphological features can be reused by multiple ocean surface grids by means of resource sharing. Thus the construction speed of the gird is effectively improved.

Figure 2 shows the multi-scale ocean surface rendering process based on the multi-scale ocean surface grid model. The division of latitude and longitude of the ocean area in the virtual earth is carried out. With the transformation of the viewpoint, the spatial data of the corresponding scale in the visible ocean area is dynamically scheduled, and the geometry and texture of the ocean surface grid are constructed by spatial data. The ocean surface grids of different scales in the same ocean area are organized in a top-down quadtree pyramid. As the viewpoint goes forward and the resolution of the grid increases exponentially, the undulating effect of ocean waves is gradually obvious.



(a) virtual earth model (b) spacial database (c) grid pyramid of sea surface

Fig. 2. Flow chart of multi-scale mapping of global ocean surface. a, latitude longitude division of the virtual earth model. b, dynamic scheduling of spatial data of corresponding scales in the visible ocean area. c, organization of the ocean surface grid by top-down quadtree pyramid.

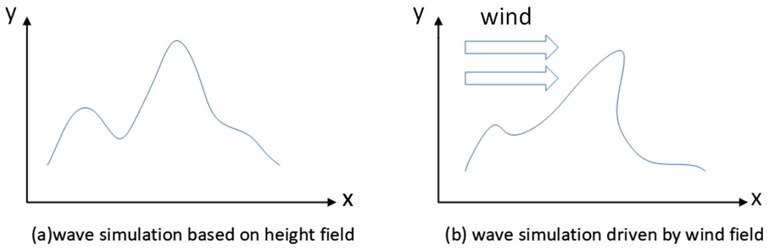
3.2 Wave Dynamic Simulation Method Driven by Wind Field

To a large extent, the sense of reality in ocean surface visualization depends on the dynamic fluctuation of ocean surface spray, especially the dynamic simulation of ocean surface spray driven by wind field, which not only meets the visual needs of users, but also matches the real geographical environment and more in line with the requirements of geographic information simulation applications. In this paper, based on the multi-scale ocean surface grid model, a dynamic simulation method of ocean wave driven by wind field is proposed.

In order to realize the dynamic simulation of ocean surface waves, we need to model the motion pattern of ocean surface waves. This paper uses a wave modeling method based on ocean wave spectrum proposed by Tessendorf [13]. The method considers that the height of the ocean surface wave is superimposed by a series of sine and cosine waves, so it can be solved by inverse transform of 2D fast Fourier. The formula is as follows:

$$h(m,t) = \sum_{k} \tilde{h}(k,t) \cdot \exp(ik \cdot m) \tag{1}$$

Where, $\mathrm{m} = (\mathrm{x}, \mathrm{y})$, which represents the horizontal position of the ocean level point. $\mathrm{t} = \frac{2\pi * \mathrm{frame}}{\mathrm{totalframe}}$, frame represents the current number of frames, totalframe represents the total number of frames of an ocean wave cycle and $\mathrm{k} = \left(\mathrm{k_x}, \mathrm{k_y}\right)$, which is a two-dimensional vector. In order to make the generated ocean surface height field have the characteristics of cyclic repeating of up, down, left and right, $\mathrm{m} = \left(\frac{\mathrm{aS}}{\mathrm{R}}, \frac{\mathrm{bS}}{\mathrm{R}}\right)$, $\mathrm{k} = (\frac{2\pi\mathrm{c}}{\mathrm{S}}, \frac{2\pi\mathrm{d}}{\mathrm{S}})$, $-\frac{\mathrm{R}}{2} < a, \mathrm{b}, \mathrm{c}, \mathrm{d} < \frac{\mathrm{R}}{2}$, S is a numerical constant representing the height field resolution, and R is the number of Fourier samples in one dimension. The influence of wind field conditions on Fourier amplitude changes with time, which is the key to calculate the wave height. However, the literature [13], which describes the law of wave motion in the form of height field sequence, can only simulate the wave fluctuation characteristics in the vertical direction, as shown in Fig. 3(a). In a real geographical environment, the waves tend to tilt under the effect of the wind field, as shown in Fig. 3(b).



(a)wave simulation based on height field

(b) wave simulation driven by wind field

Fig. 3. Horizontal wave profile. (a), wave simulation based on height field. (b), wave simulation driven by wind field. y represents the vertical direction of the ocean surface, x represents the horizontal direction and the blue arrow represents the wind field. (Color figure online)

In this paper, the Tessendorf method is improved. According to the current fluctuation state of the ocean wave and the current ocean surface wind field condition, the offset value of each wave point in the horizontal direction can be calculated to achieve more realistic wave simulation.

The main steps are as follows: First, find the gradient of formula (1):

$$G_x(m,t) = \sum_k \tilde{h}(k,t) \cdot ik_x \cdot \exp(ik \cdot m) \tag{2}$$

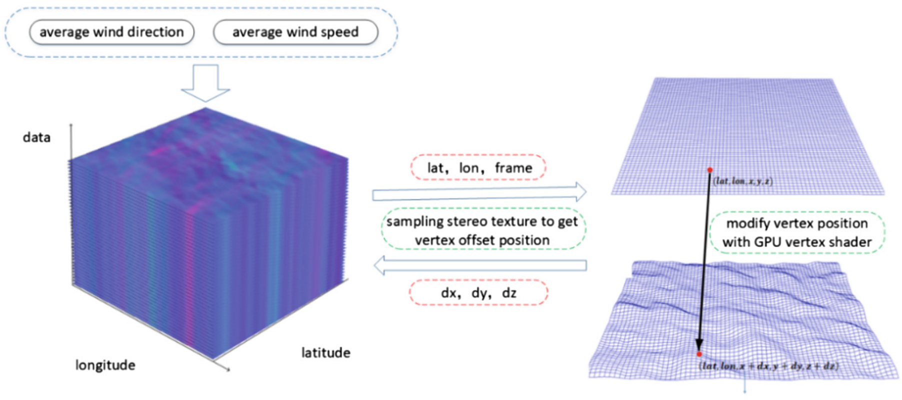
$$G_y(m,t) = \sum_k \tilde{h}(k,t) \cdot ik_y \cdot \exp(ik \cdot m) \tag{3}$$

$\mathrm{G_x(m,t), G_y(m,t)}$ represent the gradient values of the height field in the horizontal and vertical directions, which is solved based on two dimensional inverse fast Fourier transform. For the ocean surface height field, since the gradient points to the direction in which the height field grows fastest, the unit vector of the gradient vector can be regarded as the direction in which the ocean wave is most inclined. Then the offset value of the wave in the horizontal direction is:

$$d = \left(d_x \cdot d_y\right) = f \cdot \left(g_x \cdot g_y\right) \tag{4}$$

f is the dot product of the gradient direction $\left(\mathrm{g_x, g_y}\right)$ and the current wind field vector $\left(\mathrm{w_x, w_y}\right)$, reflecting the degree of influence of the wind field on the waves. Therefore, an ocean wave model of describing the current wave motion in the vertical, horizontal and vertical directions is established. The dynamic simulation of the ocean surface can be achieved by applying three directions of offset values to the initial position of the ocean level grid point. In the dynamic simulation mode, the conventional method of modifying the position of the grid point based on the CPU in real time has a large calculation amount and affects the rendering efficiency. The method of pre-building a multi-temporal grid will increase the memory footprint and affect the loading efficiency of the grid which is not suitable for dynamic visualization of the large-scale ocean surface. What's worse, the grid is difficult to modify after construction, and cannot match the real-time wind field. Based on the multi-scale ocean surface grid model, this paper proposes a dynamic simulation method of ocean surface on grounds of GPU which can update the grid point position of ocean surface in parallel through GPU, reduce CPU calculation, and load the offset value of wave motion into GPU for shader sampling in the way of displacement texture. When the wind field conditions change, only the displacement texture needs to be reloaded instead of any modification to the constructed ocean grid. The detailed algorithm flow is shown in Fig. 4:

1. Traverse the virtual earth tile quadtree to determine the current visible ocean area and construct the ocean surface grid of the current ocean area.
2. According to the spatial coding of the grids, obtain the wind direction and wind speed of the nearest ocean surface grids. The average value is taken as the parameter. The offset value of the ocean wave in the horizontal, vertical and vertical directions is calculated according to the ocean wave modeling algorithm. The



（a）current sea area displacement texture （b）a certain moment of the sea grid

Fig. 4. Ocean surface dynamic simulation process

values are stored respectively in the three channels (r, g and b) of displacement texture. The displacement textures at different moments are combined to obtain the stereo texture and then loaded into the vertex shader. If the wind speed and direction of the current scene do not change much from that of the previous frame, the displacement texture does not need to be updated, which is conducive to reducing the amount of calculation.

3. In the vertex shader, the texture coordinates are determined according to the coordinates (lat,lon) of latitude and longitude of the grid vertex and the current frame number.

$$texcoord = vec3\left(\frac{lat}{hres}, \frac{lon}{hres}, frame\%(totalframe - 1)\right) \tag{5}$$

4. hres is the span of latitude and longitude of the wave height field, m is the surface area of the earth ($m^\circ \times m^\circ$) corresponding to the ocean surface height field, totalframe is the total number of frames of a wave motion cycle, and % is the operation of get the remainder.
5. The obtained texture coordinates are used to sample the stereo texture of the current ocean area to get the offset value D of the vertex and overlay the offset value on the original vertex position.

$$position = ModelViewProjectionMatrix * \left(gl_{vertex} + D_x + D_y + D_z\right) \tag{6}$$

$D_x = D.r*N_x$, $D_y = D.g*N_y$ and $D_z = D.b*gl\_normal$. N_x, N_y are the horizontal and vertical vectors of the vertex when it cuts the earth's surface.gl_position is the vertex coordinates of the final output, gl_ModelViewProjectionMatrix is the vertex transformation matrix, gl_vertex is the original vertex coordinates and gl_normal is the vertex normal vector.

4 Experiment and Discussion

In order to verify the validity and feasibility of the proposed method, the experiment is implemented based on the osgEarth. The software environment includes Windows 7, Visual Studio 2010 and OpenGL. The hardware configuration is Intel(R) Core(TM) i3-2100 dual-core 3.1GHZ CPU, NVIDIA Quadro 600 graphics card, 1 GB video memory and 8 GB memory.

The image in the experimental data uses the ESRI_Imagery_World_2D published by ESRI. Terrain uses the Terrain Tile Service published by ReadMap. The land part uses STRM data with resolution of 90 meters, and the ocean bottom part uses GEBCO (General Bathymetric Chart of the Oceans) to provide global deep ocean measurement data ($30'' \times 30''$). The wind field data is simulated and interpolated according to the wind speed and wind direction information acquired by the ship sensor in real time. The ocean area distribution data (images of the ocean and the land are respectively represented by different colors) are generated and edited by the shoreline data in the electronic chart. According to the multi-scale ocean surface grid model before the experiment, the data is organized in multiple scales to construct a tile pyramid of a quadtree structure. Figure 5, Fig. 6 and Fig. 7 show the comparison of the ocean surface effects of the proposed method at different scales.

Fig. 5. Ocean surface global view of the method

Fig. 6. Ocean surface partial view of this method

Comparing the three pictures, it can be found that with the progressive view, the ocean surface is gradually clear, the ocean water tone transition is smooth, the effect of macroscopical ocean-land boundary and the microscopic ocean surface wave can be effectively reflected, basically satisfying the application need of global multi-scale ocean surface seamless browsing. Figure 7 and Fig. 8 show the difference between this method and the projected grid method for the ocean close-up view.

Fig. 7. Ocean surface close-up view of the method

Fig. 8. Ocean surface close-up view of projection grid method

It can be seen from the figure that the method of this paper can quickly call the ocean area distribution data of the current area with the support of the discrete grid framework to achieve the fine boundary between the land and ocean. The projection grid method can only achieve the rough ocean-land boundary effect (the ocean-land boundary scheme provided by Trit) through the surface elevation acquired by RTT technology in real time, which tends to cause the problem of seawater covering the real surface.

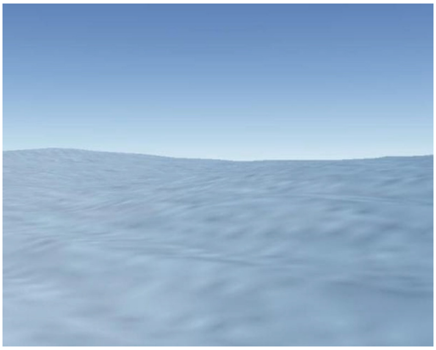


Fig. 9. Ocean wave effect with wind speed of 5 m/s

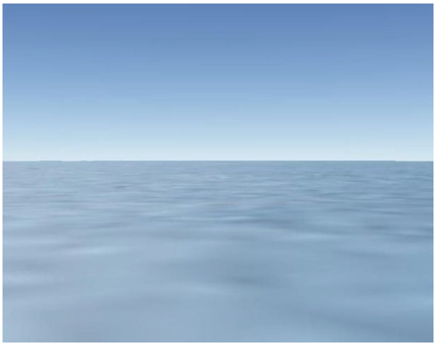


Fig. 10. Ocean wave effect with wind speed of 24 m/s

Figure 9 and Fig. 10 show the comparison of ocean surface wave effects that achieved in different wind field conditions (In Fig. 9, wind speed is breeze of 5 m/s and in Fig. 10 wind speed is gentle breeze of 24 m/s). When the wind speed is high, the undulating effect of the ocean surface is more obvious.

In order to verify the rendering and loading efficiency of the proposed method, the method is compared with the ocean surface visualization method based on projection grid and the existing ocean surface visualization method based on discrete grid under the same experimental conditions. The experimental results are shown in Fig. 11 and Fig. 12:

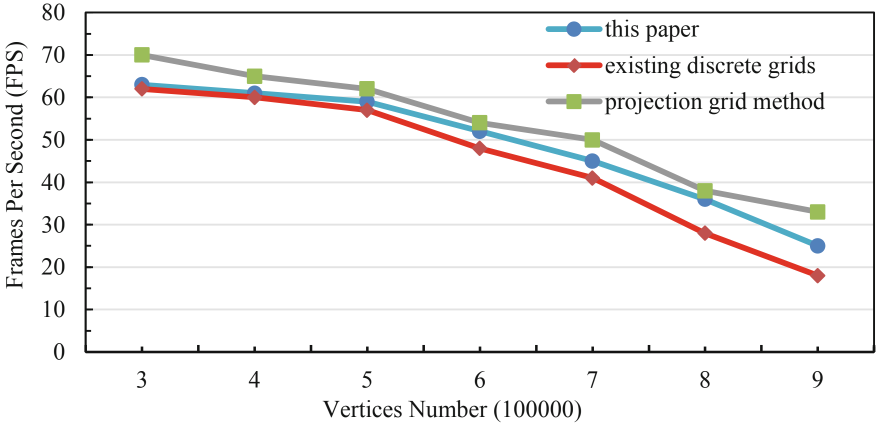


Fig. 11. Comparison of frame rate display of three methods

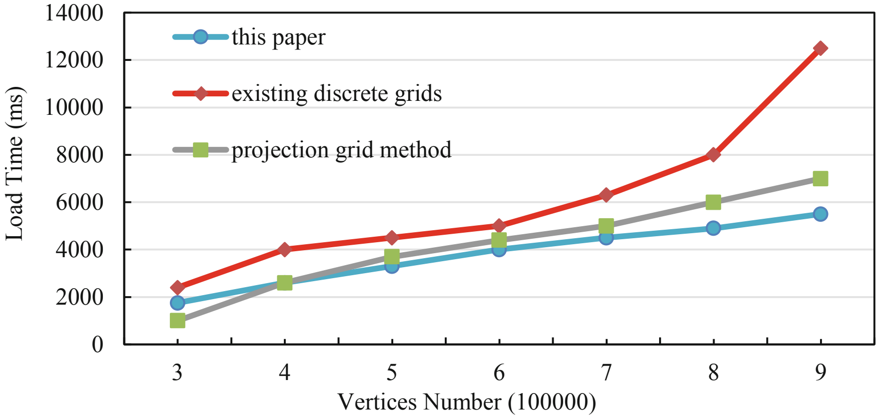


Fig. 12. Comparison of three methods in terms of loading time

It can be seen from Fig. 11 that the ocean surface visualization method based on projection grid implemented by commercial software Triton has obvious advantages in rendering efficiency. The main reason is that the ocean surface visualization method based on the discrete grid needs to simulate the ocean wave when rendering every frame so it has a certain influence on the frame rate. Because of the GPU technology, the improved discrete grid method implemented in this paper is much more efficient than the existing discrete grid method in terms of ocean wave simulation efficiency. Compared with the projected grid method, the method can maintain the frame rate difference within 6 frames/second.

Figure 12 shows that this method is faster than the other two methods. With the increase of the number of grids, the existing discrete grid method has an exponential increase in loading time. The projection grid method does not need to reconstruct a new grid, but its ocean-land boundary method has efficiency problems, which ultimately affects the overall loading speed. In contrast, the method in this paper uses tile quadtree to organize and manage the grids. There is no constraint among the peer grids so it can make full use of multi-threading for fast parallel loading. When 2 threads are enabled, the load of 900,000 grid vertices can be completed in about 5 s.

In order to verify the ability that the ocean surface visualization method can quickly respond to the change of real-time wind field, the update time of the wave effect is compared among the three methods when the wind field conditions change. As shown in Fig. 13:

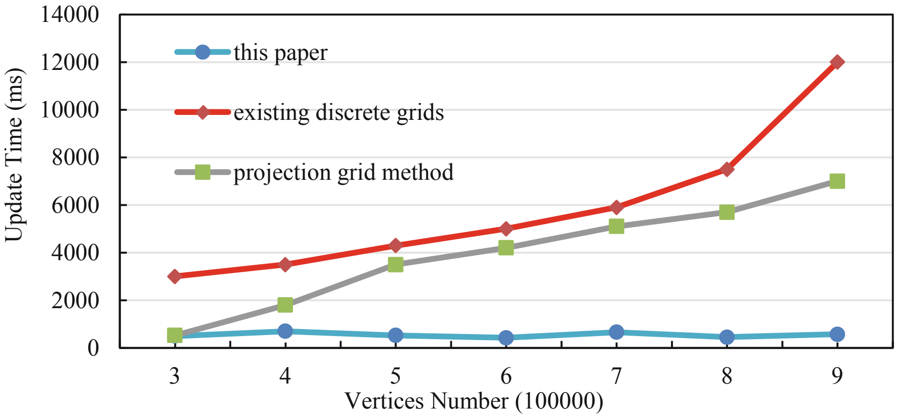


Fig. 13. Comparison of the three methods in terms of update time

As can be seen from the figure above, as the method in this paper can update the whole wave effect only by modifying the displacement texture when the wind field conditions change, the update time can be stable at about 0.5 s and the wave currently being simulated will not be affected because this part of calculation can be performed in the background thread. In the other two methods, the update of the ocean wave is equivalent to reloading the ocean surface grid, which cannot meet the needs of real-time dynamic wind field update in terms of update time and visual experience.

Compared with the existing 3D visualization method of ocean surface, the loading speed is faster and the rendering efficiency is higher. Therefore, the method is more suitable for the application needs of environment construction of the global ocean surface.

5 Conclusion and Future Work

In order to solve the problems of poor rendering efficiency and slow loading speed in the existing ocean surface 3D visualization method based on global discrete grid, this paper proposes an optimization method, which mainly includes two parts: Firstly, this paper expands on the traditional latitude and longitude discrete grid, and designs a global multi-scale ocean surface grid model based on GPU in order to effectively organize and manage the global ocean surface grids, while making full use of GPU parallel computing capabilities. Secondly, this paper proposes a real-time updating method of wave dynamic simulation based on multi-scale ocean grid model, considering that the effect of wave fluctuation driven by wind field is more in line with the needs of geographic information simulation application. Finally, the method was verified by comparison experiments. The experiment results show that the proposed method has stable rendering efficiency, fast loading speed, and has the functions of reflecting the characteristics of different ocean areas, realizing the fine land-ocean boundary and supporting the dynamic update of wind field conditions, which is more in line with the application needs of global ocean surface environment simulation. The next step will be to discuss how to achieve real-time interactive dynamic simulation between global ocean surface and ship navigation, laying the foundation for specific applications of the global ocean.

Acknowledgments. This study was jointly supported by China Southern Power Grid Guangzhou Power Supply Bureau Co., Ltd. Key Technology Project (0877002018030101SRJS0 0002); Guangdong Provincial Science and Technology Program (2017B010117008); Guangzhou Science and Technology Program (201806010106, 201902010033); the National Natural Science Foundation of China (41976189,41976190); the Guangdong Innovative and Entrepreneurial Research Team Program (2016ZT06D336); the Southern Marine Science and Engineering Guangdong Laboratory (Guangzhou) (GML2019ZD0301); the GDAS's Project of Science and Technology Development (2016GDASRC-0211,2018GDASCX-0403,2019GDASYL-0301001, 2017GDASCX-0101,2018GDAS CX-0101,2019GDASYL-0103003).

References

1. Yoo, J.S., Min, K.J., Ahn, J.W.: Concept and framework of 3D geo-spatial grid system. In: Kawai, Y., Storandt, S., Sumiya, K. (eds.) W2GIS 2019. LNCS, vol. 11474, pp. 136–149. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-17246-6_11

2. San, O., Staples, A.E.: An efficient coarse grid projection method for quasigeostrophic models of large-scale ocean circulation. Int. J. Multiscale Comput. Eng. **11**(5), 463–495 (2013)
3. Lv, C., Yu, F.: The application of a complex composite fractal interpolation algorithm in the seabed terrain simulation. **2018**, 1–6 (2018)
4. Li, C., Li, T., Huang, Q.: Research status and prospect for maritime object monitoring technology. J. Phys: Conf. Ser. **1288**, 012064 (2019)
5. Wang, Q., et al.: Grid evolution method for DOA estimation. IEEE Trans. Sig. Process. **66** (9), 2374–2383 (2018)
6. Hong, X., et al.: Simulating and understanding the gap outflow and oceanic response over the Gulf of Tehuantepec during GOTEX. Dyn. Atmos. Oceans **82**, 1–19 (2018)
7. Reeves, W.T.: A Simple Model of Ocean Waves (1986)
8. Bailey, R.J., Jones, I.S.F., Toba, Y.: The steepness and shape of wind waves. J. Oceanogr. Soc. Japan **47**(6), 249–264 (1991)
9. Huaiqing, H., et al.: A way to real-time ocean wave simulation. In: International Conference on Computer Graphics, Imaging and Visualization (CGIV 2005) (2005)
10. Mastin, G.A., Watterberg, P.A., Mareda, J.F.: Fourier synthesis of ocean scenes. IEEE Comput. Graph. Appl. **7**(3), 16–23 (1987)
11. Chapman, P., et al.: Seabed visualization. In: Proceedings Visualization 1998 (Cat. No. 98CB36276) (1998)
12. Chen, H., Li, Q., Wang, G., Zhou, F., Tang, X., Yang, K.: An efficient method for real-time ocean simulation. In: Hui, K.-C., et al. (eds.) Edutainment 2007. LNCS, vol. 4469, pp. 3–11. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-73011-8_3
13. Tessendorf, J.: Simulating Ocean Water. SIGGRAPH2001 Course notes, pp. 47–58. Addison Wesley, Boston (2001)
14. Schlitzer, R.: Interactive analysis and visualization of geoscience data with Ocean Data View. Comput. Geosci. **28**(10), 1211–1218 (2002)
15. Li, H., Quan, W., Xu, C., Wu, Y.: A GPU-based mipmapping method for water surface visualization. In: Proceedings of SPIE 10610, MIPPR 2017: Parallel Processing of Images and Optimization Techniques; and Medical Imaging, 1061003, 6 March 2018
16. Su, T., et al.: Multi-dimensional visualization of large-scale marine hydrological environmental data. Adv. Eng. Softw. **95**, 7–15 (2016)
17. Zhang, F., et al.: Spatial and temporal processes visualization for marine environmental data using particle system. Comput. Geosci. **6**(1), 53–54 (2019)
18. Liu, S., et al.: A framework for interactive visual analysis of heterogeneous marine data in an integrated problem solving environment. Comput. Geosci. **7**(1), 20–28 (2017)