Ocean wave simulation by the mix of FFT and Perlin Noise

LI TIAN

Institute of Literature, Chinese Academy of Social Sciences
China, Beijing
skyhumxp@163.com

ABSTRACT

For the application of ocean wave, a new height-field simulation method is proposed by the mix of FFT and Perlin Noise, and OpenSceneGraph (OSG) and VC++ 2008 are used to simulate realistic ocean wave. The method takes wind effects into consideration, and uses Philips model and Gauss random numbers to produce ocean wave spectrum, which is then transformed to wave height-field by FFT. Perlin Noise is overlaid to disturb the wave height to generate a vivid and random sea surface. Simulation results show the effectiveness of the proposed method.

Keywords

Ocean wave spectrum, Fast Fourier Transform (FFT), Perlin Noise, Height-field, OpenSceneGraph (OSG)

1. INTRODUCTION

The ocean covers 71 percent of the Earth's surface, and plays a significant role in the Earth's environment. Ocean wave simulation has attracted a lot of attention since the appearance of computer graphics. It can be used in virtual reality, digital Earth, film art and so on [Dar11a]. For example, in Titanic and The Day After Tomorrow, many sea water scenes were simulated by computer [Tes01a].

Nowadays there have been several ocean wave simulation methods, which can be classified into five different types. The first type is based on physical model, the second is based on geometry model, the third is based on wave spectrum, the fourth is based on particle system, and the fifth is based on Perlin Noise.

The first type uses the classical fluid mechanics and solves Navier-Stokes equations to get realistic wave shape, but it is too time-consuming [Fos00a]. The second simulates wave shape by constructing a mathematical function, and its effect is highly dependent on the function itself [Fou86a] [Tso87a]. The third assumes that the ocean wave is composed by many sine waves with different amplitudes, different frequencies and different propagating directions, and FFT are used to transform wave spectrum to the corresponding ocean wave [Tes01a] [Mit05a]. The fourth supposes the water surface is made up of a lot of moving particles, which follow certain rules. Large number of particles will formulate realistic wave, but also cause heavily computation burden [Pea86a]. The fifth uses the summation of coherent noises to construct sea surface height field, the trouble is that it can't account for wind effect [Per85a].

As statistical ocean wave spectrum models are adopted, FFT method can generate high realistic wave. But it supposes the whole sea surface is impacted by the same wind, hence the generated ocean wave has some similarity and repeatability, especially for a large area of sea. The paper proposes a new ocean wave simulation method, which introduces Perlin Noise into FFT method to improve vitality and reality of ocean wave. Also in Windows platform, an example of realistic ocean is simulated by OSG and VC++ 2008.

2. HEIGHT-FIELD GENERATION

FFT method

With the assumption of no surface tension and uniform seawater density, the solution of simplified Navier-Stokes equations could be expressed as follows

$$\mathrm{h}(x,y,t)=\iint \overline{h}(\mathrm{k}_x,k_y,t)\exp\{i(k_x x+k_y y)\}\,dk_x dk_y \quad (1)$$

$$\overline{h}(\vec{k},t)=\overline{\mathrm{h}}_0(\vec{k})\exp\{-i\omega_0(\vec{k})t\}+\overline{\mathrm{h}}_0^*(-\vec{k})\exp\{i\omega_0(\vec{k})t\} \quad (2)$$

$$\omega_0(\vec{k})=\sqrt{\mathrm{g}\left|\vec{k}\right|} \quad (3)$$

Where $\mathrm{h}(x,y,t)$ is the wave height at time t and position (x,y), $\bar{h}(\vec{k},t)$ is the spatial spectrum of $\mathrm{h}(x,y,t)$, and $\vec{k}=(k_x,k_y)$ is the two-dimensional ocean wave vector.

For a sea surface with area $L_\mathrm{x}\times L_\mathrm{y}$ and grid number $N\times M$, the digitized (1) would be

$$\mathrm{h}_{\mathrm{fft}}(x,y,t)=\sum_{\mathrm{n}=N/2}^{N/2-1}\sum_{m=M/2}^{M/2-1}\bar{h}(\vec{k},t)\exp\{i(k_x x+k_y y)\} \quad (4)$$

And

$$\mathrm{k}_x=2\pi n/L_x,\ \mathrm{n}\in[-N/2,N/2),\quad \mathrm{k}_y=2\pi m/L_y,\ \mathrm{m}\in[-M/2,M/2) \quad (5)$$

The equation (4) can be computed by 2D FFT to improve its efficiency. In (2), Phillips spectrum is often used to calculate ocean spatial spectrum, so

$$\bar{\mathrm{h}}_0(\vec{k})=\frac{1}{\sqrt{2}}(\xi_\mathrm{r}+i\xi_i)\sqrt{P_h(\vec{k})} \quad (6)$$

$$\sqrt{P_h(\vec{k})}=A\frac{\exp\left(-1\Big/\left(\left|\vec{k}\right|L\right)^2\right)}{\left|\mathrm{k}\right|^4}\left|\vec{k}\cdot\vec{\omega}\right|^2 \quad (7)$$

Where ξ_r and ξ_i are Gaussian number with mean 0 and variance 1, A is the wave amplitude, $\vec{\omega}$ is the wind vector, and $L=\left|\vec{\omega}\right|^2\Big/\mathrm{g}$ is the maximal wave length under $\vec{\omega}$.

In Philips model, the cosine factor $\left|\vec{k}\cdot\vec{\omega}\right|$ eliminates the energy vertical to wind direction, and makes the main direction of generated wave consistent with wind direction. The maximal wavelength L is determined by wind speed, the larger the wind speed is, and the higher the generated wave will be. When the wavelength is very short ($l<<L$), the convergence of Philips model will be very band, and the following formula can be multiplied to improve it.

$$\exp\left(-\left|\vec{k}\right|^2 l^2\right) \quad (8)$$

Perlin Noise method

In 1985, Ken Perlin developed Perlin Noise function, which used fractal principle to generate coherent noise [Per85a]. Perlin Noise is rather successful to simulate landscapes, clouds and many different sorts of textures.

The ocean wave simulation method by Perlin Noise could be shown as

$$\mathrm{h}_{\mathrm{perlin}}(x,y)=\sum_{i=0}^{P-1}\mathrm{a}^i\cdot noise\left(2^i x,2^i y\right) \quad (9)$$

Where P denotes the number of octaves to be added, 2^i and a^i are the frequency and amplitude of the octave i respectively, and $0<\mathrm{a}<1$. The more octaves, the more details will be.

Fig. 1 gives a typical example of Perlin Noise. The left image is the basic noise image, the next three images are its octave 2, 4, and 8, and the right image is the coherent summation of all left four images. Fig. 1 shows that special texture can be generated by the summation of multiple fractal images with different frequencies and different amplitudes.

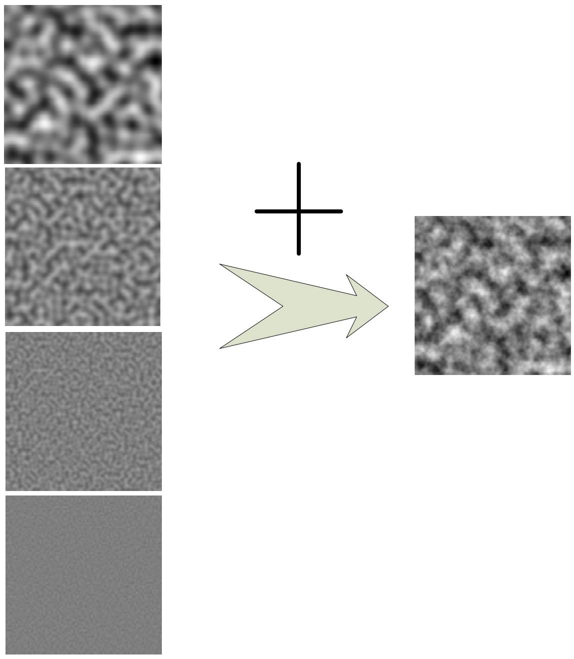


Figure 1 Perlin Noise

Mixed Perlin noise and FFT method

Wind is one of the most important reasons for ocean wave. As statistical and empirical spectrum models are adopted, FFT method can generate wave height-field with wind effect. But the behavior of wind is very complex, even in a small area, the wind speed and direction are not the same. When we see sea surface from near to far, clear wave direction by wind can be seen at near distance, but in the far distance, there is only noise like surface. Perlin Nose can also be used to generate sea surface, but it can’t consider the effect of wind. Therefore the fusion of FFT method and Perlin Noise will be an effective way to produce much more realistic wave.

The mixed height-field could be expressed as following

$$\mathrm{h}_{\mathrm{mixed}}(x,y,t)=\alpha(x,y)\sum_{\mathrm{n}=N/2}^{N/2-1}\sum_{m=M/2}^{M/2-1}\bar{h}(\vec{k},t)$$
$$\bullet\exp\{i(k_x x+k_y y)\}+\beta(x,y) \quad (10)$$
$$\bullet\sum_{i=0}^{P-1}\mathrm{a}^i\cdot noise\left(2^i x,2^i y\right)$$

Where $\alpha(x,y)$ and $\beta(x,y)$ are the weight coefficients of FFT method and Perlin Noise respectively. As the output of Perlin Noise is always between -1 and +1, its coefficient should be $\beta(x,y)\approx \mathrm{h}_{0\max}\left[1-\alpha(x,y)\right]$, where $\mathrm{h}_{0\max}$ is the maximal value of $\bar{\mathrm{h}}_0(\vec{k})$. By choosing different $\alpha(x,y)$ and $\beta(x,y)$, the proportion of FFT method and Perlin Noise can be controlled, therefore much more realistic ocean surface can be simulated.

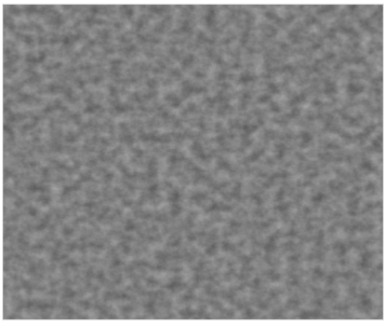
3. OCEAN SURFACE SIMULATION

OpenSceneGraph (OSG) is an open-source 3D graphics engine. It provides many scene management and graphics rendering functionalities to the development of graphics programs, and has been wide-spreading applications in visual simulation, game, virtual reality, high technology research. OSG is programmed based the industrial standard OpenGL and ANSI C++, so it has the feature of cross-platform. By providing an object oriented framework using the concept of scene graph, OSG can free developers from the underlying invocation of OpenGL API.

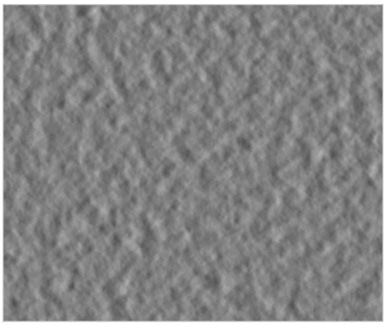
OSG library and VC++ 2008 are used to simulate ocean scene in this paper. Triangular-patch model of ocean surface is constructed based by the height-field data, and then realistic ocean surface can be simulated by color texture and its mapping. A simulation example will be given in the next section.

4. EXAMPLES

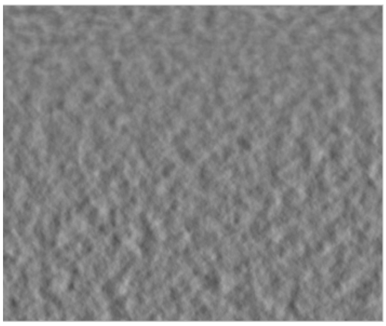
Fig. 2(a), (b) and (c) shows three ocean height-field images generated by Perlin Noise method, FFT method and the proposed method respectively. In these images, simulated ocean size is $512m\times 512m$, grid size is $1m\times 1m$, and pixel grayscale denotes the corresponding wave height. The brighter the pixel is, the higher the wave height is. For FFT method, wind speed and direction are 7m/s and 0 degree (from left to right) respectively. In order to show the wind effect in near field and noise effect in far field, $\alpha(x,y)=1-\left(y/L_y\right)^3$ is used in the proposed method to mix the FFT and Perlin Nose. Furthermore fig. 3(a), (b) and (c) depicts wind effect of 45^o, 90^o and 135^o respectively. Fig. 4 is a rendering example of fig. 3(b) by OSG and VC++ 2008.



（a）Perlin Noise method

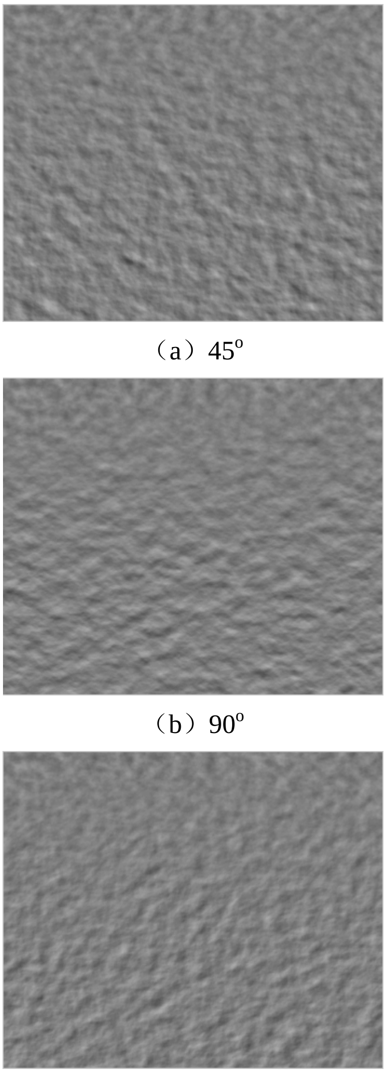


（b）FFT method



（c）mixed method

Figure 2 Sea height-field simulations by different methods



（a）45^o

（b）90^o

（c）135^o

Figure 3 Wind direction effect

5. CONCLUSION

As the development of computer graphics, ocean simulation has been acquiring more and more application. The paper proposes a new ocean simulation method, which fuses FFT and Perlin Noise to generate a more vivid and random sea height-field. Simulation results are given to validate the proposed method; however the realistic rendering of large ocean scene and real-time accelerated processing will be the next study focus.

6. REFERENCES

[Dar11a] Darles, E., Crespin B., Ghazanfarpour, D., et al. A survey of ocean simulation and rendering techniques in computer graphics. Computer Graphics Forum 30, 1, pp.43-60, 2011.

[Tes01a] Tessendorf J. Simulating ocean water, SIGGRAPH 2001: Inspire Interaction and Digital Images, pp.301-318, 2001..

[Fos00a] Foster N., Metaxas D. Modeling water for computer animation. Commun ACM, 43(7), pp.60-67, 2000.

[Fou86a] Fournier A., Reevesw T. A simple model of ocean waves. Computer Graphics, 20(4), pp.75-84, 1986.

[Tso87a] Ts O P., Barskey B. Modeling and rendering wave: wave-tracking using beta-splines and reflective texture mapping. ACM Transaction on Graphics, 16(3), pp.191-214, 1987.

[Mit05a] Mitchell J. Real-time synthesis and rendering of ocean water. Marlborough: Array Technology Industry Technologies inc., 2005.

[Pea86a] Peachy D. Modeling wave and surf. Computer Graphics, pp.65-74, 1986.

[Per85a] Perlin K. An image synthesizer. Computer Graphics, 19 (3), pp.287-296, 1985.

[Rui12a] Rui W., Xuelei Q. OpenSceneGraph 3 Cookbook. Birmingham: Packt Publishing, 2012.

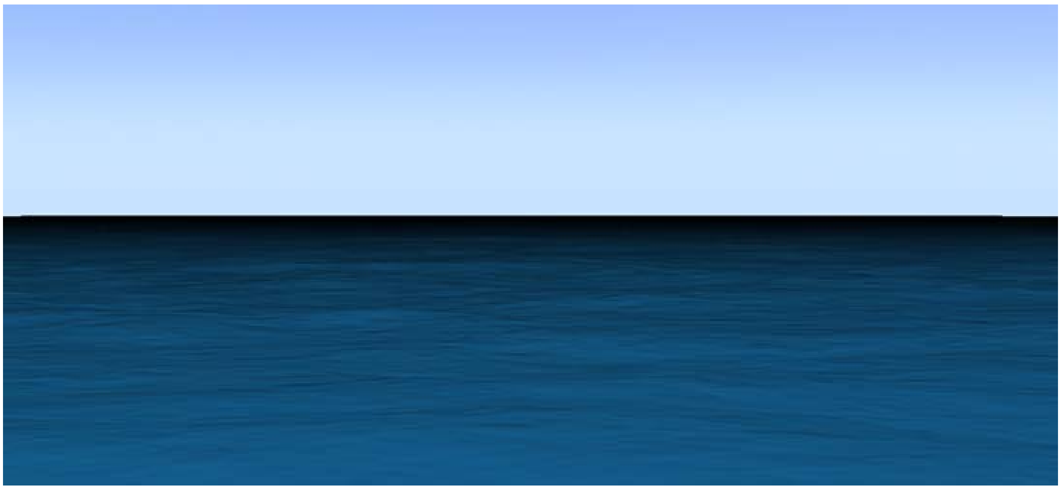


Figure 4 Simulation example