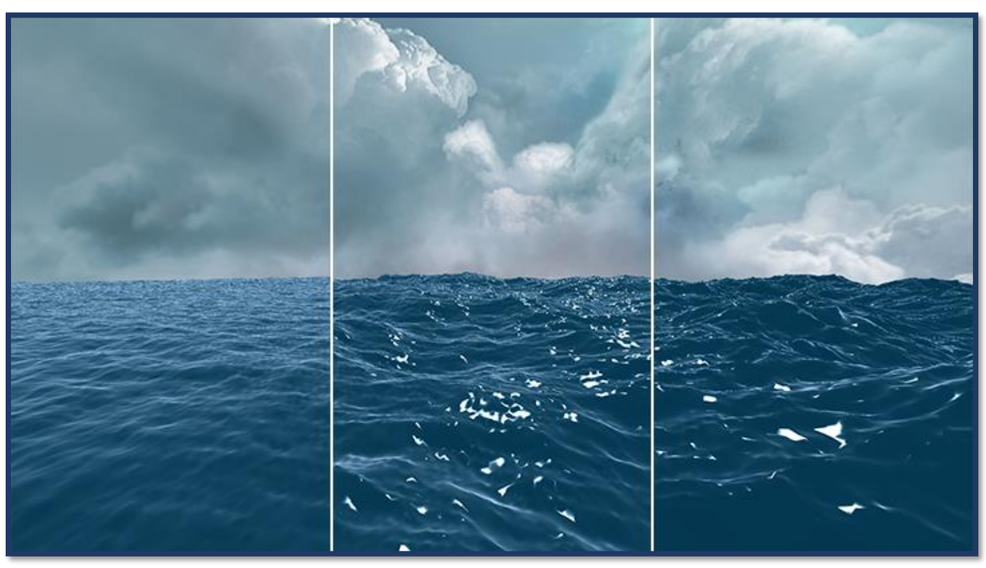
Large Scale Ocean Rendering and Simulation

Spencer McKean

Project Description

Inspired by the large and magnificent ocean in the game Sea of Thieves, I wanted to recreate the same feeling of awe in my own game engine.

Using the Fast Fourier Transform to simulate the movement of water and utilizing the DirectX11 shader pipeline to recreate the appearance of water, I achieved something close to what Sea of Thieves created.



$$h(\vec{x},t) = \sum_{\vec{k}} \tilde{h}(\vec{k},t)\, e^{i\vec{k}\cdot\vec{x}}$$

$$\tilde{h}(\vec{k},t) = \tilde{h}_0(\vec{k})e^{i\omega(k)t} + \tilde{h}_0^*(-\vec{k})e^{-i\omega(k)t}$$

$$\tilde{h}_0(\vec{k}) = \frac{1}{\sqrt{2}}(\xi_r + i\xi_i)\sqrt{P_h(\vec{k})}$$

$$P_h(\vec{k}) = A\frac{e^{-1/(kL)^2}}{k^4}\left|\hat{k}\cdot\hat{\omega}\right|^2$$

Fast Fourier Transform Wave Surface Simulation

- Based on ocean surface statistical models
- Allows for more complex surfaces than other methods currently used in the industry
- Exposure to different real-world variables allows for easy modification from external sources

Rendering: Reflectivity and Transmissivity

- Light interaction with water is complex
- Every time light hits the water a portion of it is reflected and another is transmitted
- Refraction of light as it passes through water is dictated by the lights angle of incidence