Name : Sanketh Karuturi

Email ID: karutusa@oregonstate.edu

CS 557 *Computer Graphics Shaders:* Final Project

Dynamic Ocean Wave Simulation
Final Project Proposal

Title: Dynamic Ocean Wave Simulation Using GLSL Shaders
Name: Sanketh Karuturi
Email Address: karutusa@oregonstate.edu

Project Objective:
The goal of this shaders final project is to develop a realistic and interactive ocean wave simulation using the OpenGL Shading Language (GLSL). This simulation will aim at replicating the dynamic and complex nature of ocean waves in the real world, including reflection, refraction, and the interaction of light with the water surface. The project will leverage vertex and fragment shaders to generate and render the wave animations in real-time.

Technical Approach:

- **Vertex Shader :** Will be responsible for simulating the wave motion. This will involve implementing algorithms for wave generation, such as the Gerstner wave model, to produce realistic wave patterns. The vertex shader will calculate vertex positions and normals to simulate the undulating surface of the ocean.
- **Fragment Shader :** Will handle the rendering aspects, including the water's colour, transparency, and the effects of light on the water surface. It will implement lighting models to simulate sunlight reflection and the water's diffuse and specular reflections.
- **GLIB File :** this file will contain setup configurations, shader program initialization, and utility functions to support the shader operations. It will ensure the shaders are correctly compiled, linked, and applied to the geometry.

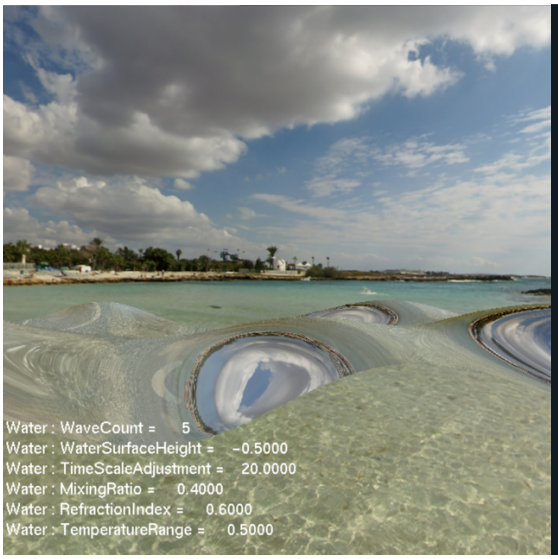
Learning Goals:
Through this project, I aim to deepen my understanding of:

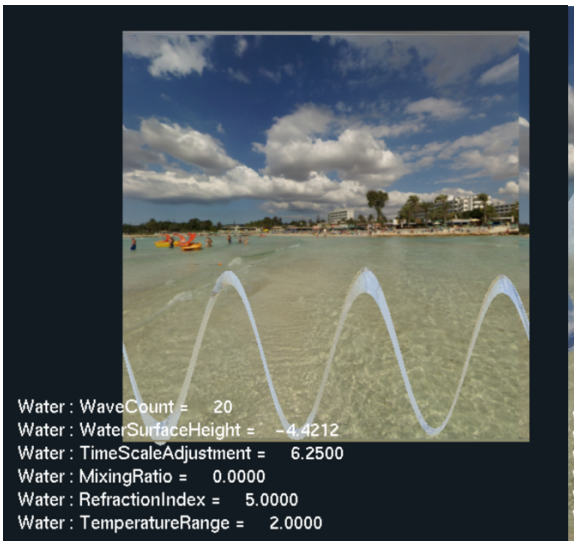
- Advanced shader programming techniques and their optimization for real-time rendering.
- Complex water simulation algorithms and their implementation in GLSL.
- The interaction of light with dynamic surfaces and how to realistically render these effects using shaders.

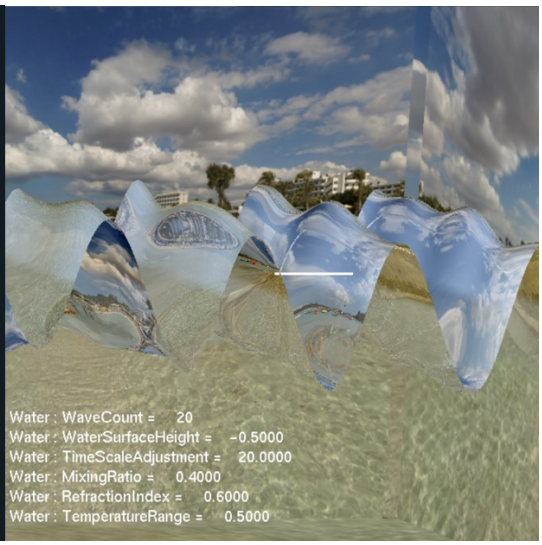
Conclusion:

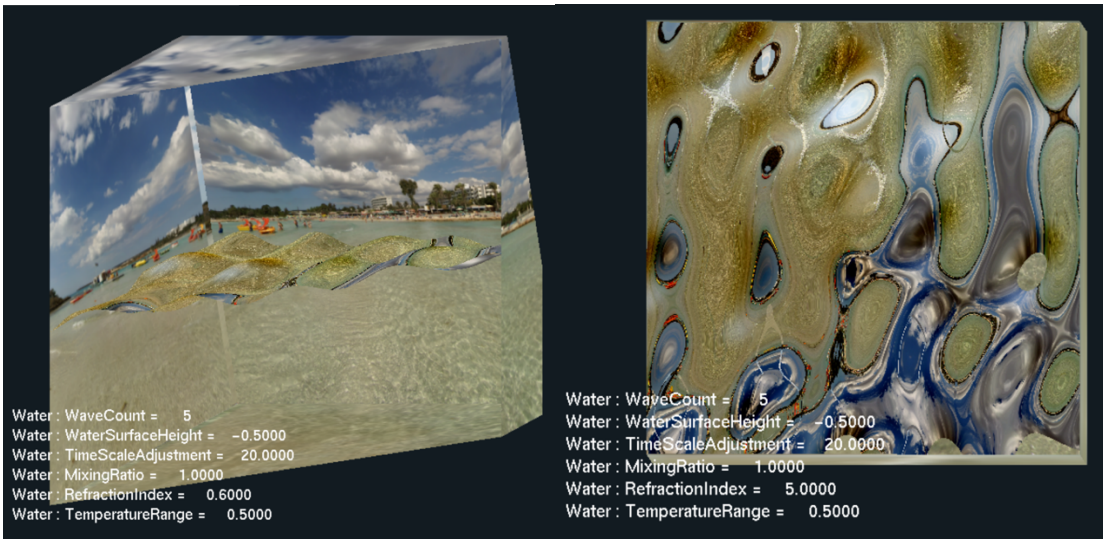
This project represents a challenging but achievable goal within the one-week timeframe. It will not only demonstrate my ability to apply and extend my knowledge of GLSL shaders but also result in a visually impressive simulation that could serve as a foundation for future projects or applications requiring realistic water rendering.

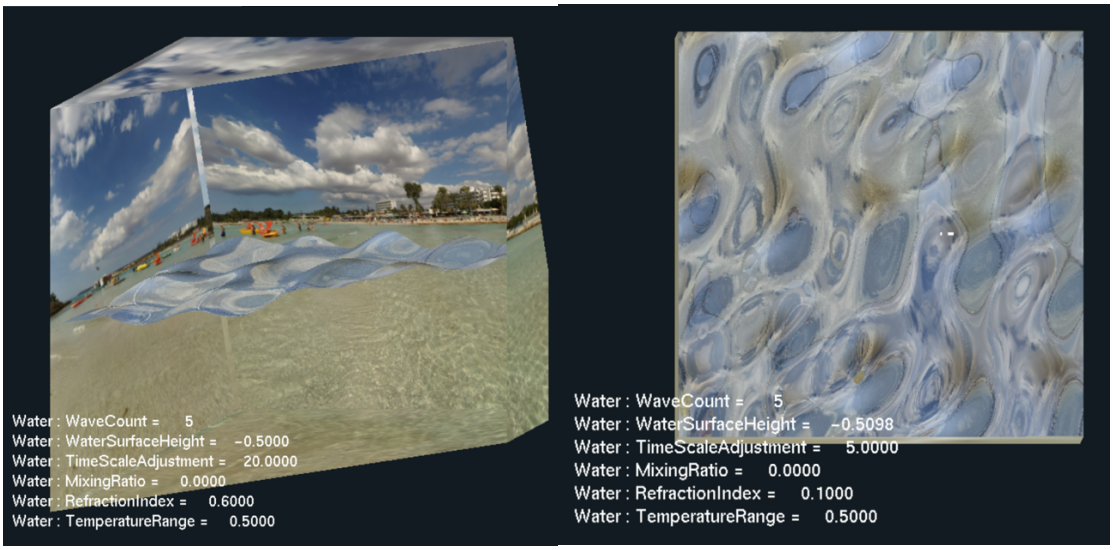
1) Project Screenshots

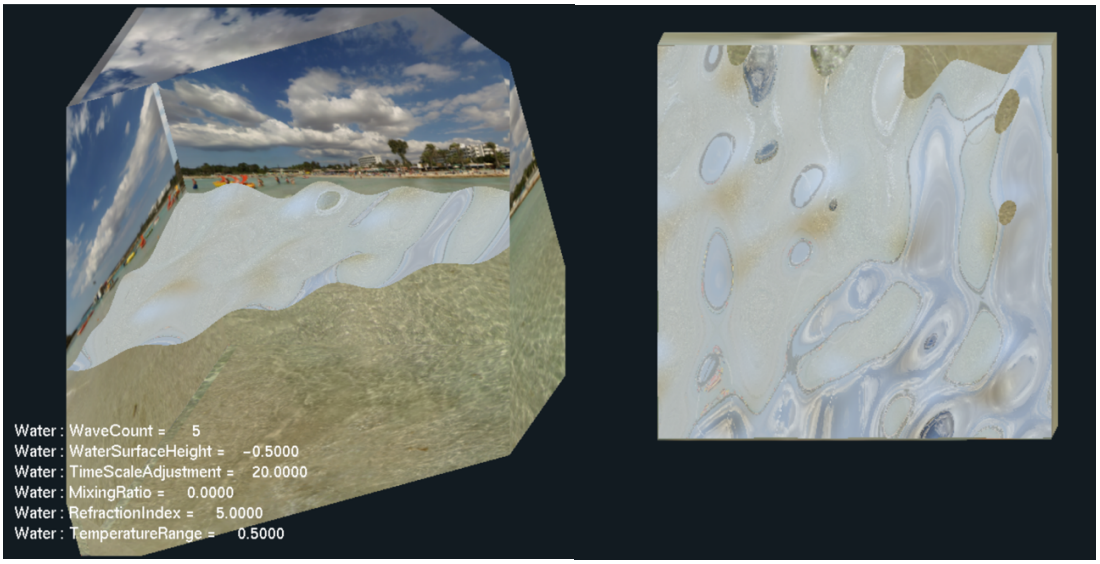







Water : WaveCount = 5
Water : WaterSurfaceHeight = −0.5000
Water : TimeScaleAdjustment = 20.0000
Water : MixingRatio = 1.0000
Water : RefractionIndex = 0.6000
Water : TemperatureRange = 0.5000
Water : WaveCount = 5
Water : WaterSurfaceHeight = −0.5000
Water : TimeScaleAdjustment = 20.0000
Water : MixingRatio = 1.0000
Water : RefractionIndex = 5.0000
Water : TemperatureRange = 0.5000


Water : WaveCount = 5
Water : WaterSurfaceHeight = −0.5000
Water : TimeScaleAdjustment = 20.0000
Water : MixingRatio = 0.0000
Water : RefractionIndex = 0.6000
Water : TemperatureRange = 0.5000
Water : WaveCount = 5
Water : WaterSurfaceHeight = −0.5098
Water : TimeScaleAdjustment = 5.0000
Water : MixingRatio = 0.0000
Water : RefractionIndex = 0.1000
Water : TemperatureRange = 0.5000


Water : WaveCount = 5
Water : WaterSurfaceHeight = −0.5000
Water : TimeScaleAdjustment = 20.0000
Water : MixingRatio = 0.0000
Water : RefractionIndex = 5.0000
Water : TemperatureRange = 0.5000

2) Project Description

- **Implemented Vertex Shader (OceanWave.vert):**
    - Simulated wave motion using algorithms inspired by the Gerstner wave model, calculating realistic wave patterns.
    - Calculated vertex positions and normals to replicate the undulating surface of the ocean, adjusting for wave amplitude, wavelength, direction, and phase.
    - Incorporated dynamic adjustments based on the **TemperatureRange** uniform, allowing for the simulation of environmental effects on Ocean Wave dynamics, such as freezing or increased wave activity.
- **Developed Fragment Shader (OceanWave.frag):**
    - Rendered the ocean's appearance by controlling the water's color, transparency, and light interactions, including reflection and refraction.
    - Utilized cube map textures for realistic environmental reflections on the water surface.
    - Implemented lighting models to simulate sunlight reflection and the water's diffuse and specular reflections.
    - Adjusted the visual effects based on water temperature, simulating transitions between liquid water and ice for different environmental conditions.
- **Configured GLIB File (OceanWave.glib):**
    - Created cube map textures in texture units for environment mapping, crucial for achieving realistic reflections and refractions.
    - Defined a "OceanWave" shader program with uniform parameters for detailed control over reflections, refractions, water surface height, wave count, and dynamic effects like temperature adjustments.
    - Mapped shaders onto a large quad in the XZ plane to simulate the water surface, ensuring a broad area for displaying water effects.
- **Shader Program Integration and Testing:**
    - Compiled and linked vertex and fragment shaders within the OpenGL context, ensuring they were correctly applied to the geometry.

    - Conducted extensive testing to fine-tune wave dynamics, lighting effects, and performance optimizations for real-time rendering.
- **Dynamic Environment Interaction:**
    - Added functionality to modify wave properties in real-time based on environmental factors, such as temperature changes, to enhance realism and interactivity.
    - Implemented user inputs to adjust viewing angles, wave parameters, and environmental settings, allowing for an interactive simulation experience.
- **Optimization and Performance Tuning:**
    - Optimized shader code for efficiency to achieve smooth, real-time rendering of complex wave animations on consumer-grade hardware.
    - Profiled and adjusted rendering settings to balance visual fidelity with performance, ensuring a broad range of systems could run the simulation effectively.

3) **Media Link :** https://media.oregonstate.edu/media/t/1_2wyhseew

4) **If your final result is different from what you proposed, explain how it is different and why.**

   → Yes , My project is different from what I proposed, I implemented minor improvements

- **Enhanced Environmental Dynamics:**
    - **Initial Proposal:** The project aimed to simulate ocean waves with basic interaction of light, including reflection and refraction.
    - **Final Implementation:** Added dynamic environmental factors such as temperature, which affects the wave motion and transitions between liquid water and ice, enhancing realism beyond the original scope.

- **Advanced Lighting Models:**
    - **Initial Proposal:** Planned to implement basic lighting models to simulate sunlight reflection and water's diffuse and specular reflections.
    - **Final Implementation:** Integrated more complex lighting techniques to achieve a more realistic rendering of sunlight and its interaction with the water surface.
- **Temperature-Based Visual Effects:**
    - **Initial Proposal:** Focused on the dynamic nature of ocean waves without considering the impact of environmental temperature.
    - **Final Implementation:** Introduced temperature as a variable affecting the water's visual properties, such as color and transparency, to simulate freezing and melting effects, adding an extra layer of interaction.
- **User Interaction Enhancements:**
    - **Initial Proposal:** Aimed for a realistic wave simulation with limited user interaction.
    - **Final Implementation:** Developed an interactive interface allowing users to modify wave parameters, environmental conditions, and view perspectives in real-time, significantly increasing the simulation's interactivity.
- **Expanded Shader Functionality:**
    - **Initial Proposal:** Planned on utilizing standard shader functionalities for water simulation.
    - **Final Implementation:** Extended shader capabilities to include custom functions for wave height calculation, surface perturbation based on noise textures, and dynamic adjustments based on environmental factors, providing a more versatile and realistic simulation toolkit.