AR Water Effects Visualization

Ege Dumanlı (e.dumanl@student.tudelft.nl)
Professor: Elmar Eisemann
Supervisor: Mark Winter
CSE3000 20-06-2022

1 - BACKGROUND

Sailing+ application is an **AR/VR app** that aims to engage those interested in the sport of sailing by getting them to experience sailing regattas.

Visually compelling water simulation is a vital component of a sailing visualization application.

A **shader** is a computer program that modifies an image before it is rendered and drawn to the screen. They can be used to create **many visual effects**:

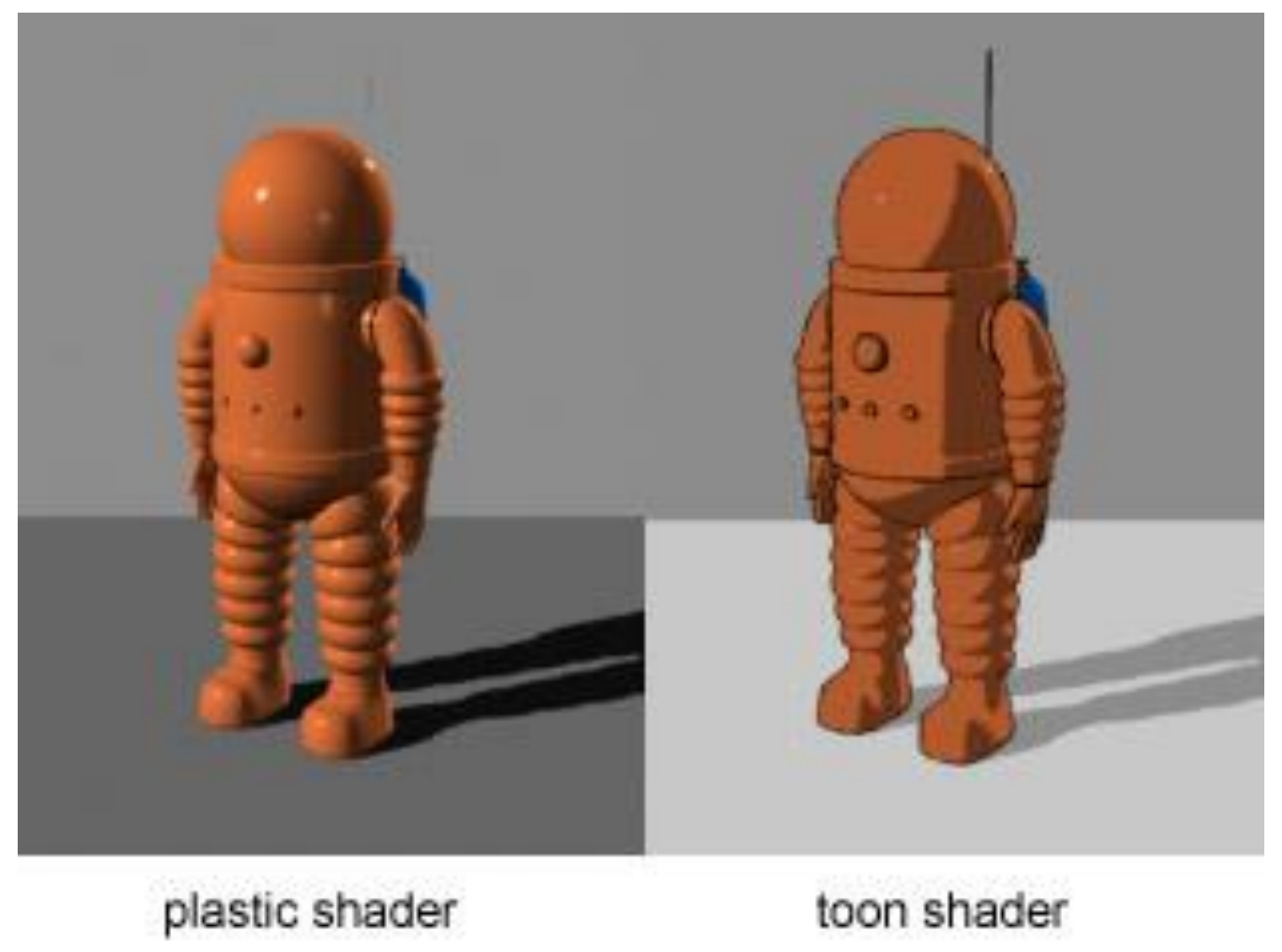

Figure 1 – A classic example of shaders
https://upload.wikimedia.org/wikipedia/commons/b/b7/Toon-shader.jpg

2 - QUESTIONS

- What types of water effect implementations are efficient and effective for AR environments?
- What water effects can be used to enhance the AR experience?
- How will the effects on fluids be implemented to work with AR environments?

3 – METHODS

Planar Reflections:

Environment is dynamically reflected by **treating the water as a plane**.

The camera is **reflected to the other side of the plane** and the render result is used as the reflection texture.

Oblique View Clipping is used to clip objects underwater and only render reflections for things that are above the water plane.

A plane is created in the scene that uses the user's mobile camera output as its texture.

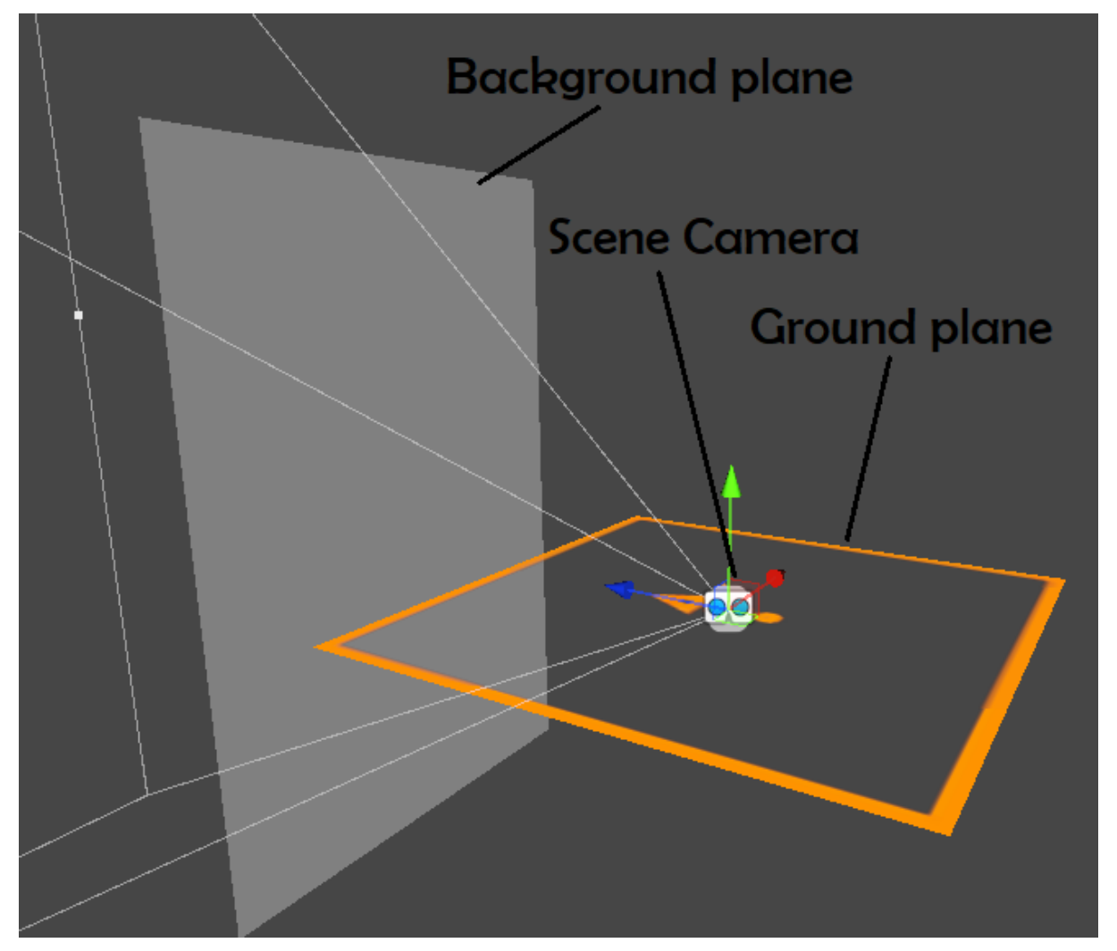

Figure 2 –The background plane

This way, the planar reflection code can also reflect the user's environment. The newly created plane **always faces the camera and moves tangent to a circle covering the ground plane.**

Sampling Refraction Texture with Screen Space UV's:

When the water plane is projected onto a surface, the things that remain under the plane should be distorted as if caused by the refraction of light.

The screen space UV coordinates of the water plane is used while sampling the refraction texture which is the output of the user's mobile camera. This way, only the parts under the water plane are rendered on the water plane.

Finally, this texture can be **distorted** by **multiplying** the normal vectors of the water plane with a refraction value and **adding** it to the sampled value
.

Caustics Approximation:

Caustics are bright parts which are lit by multiple rays. They can be approximated by assuming that the sun is shining from **directly above**. When this assumption is made, the following texture can be used to simulate sun:

Figure 3 – Caustics*

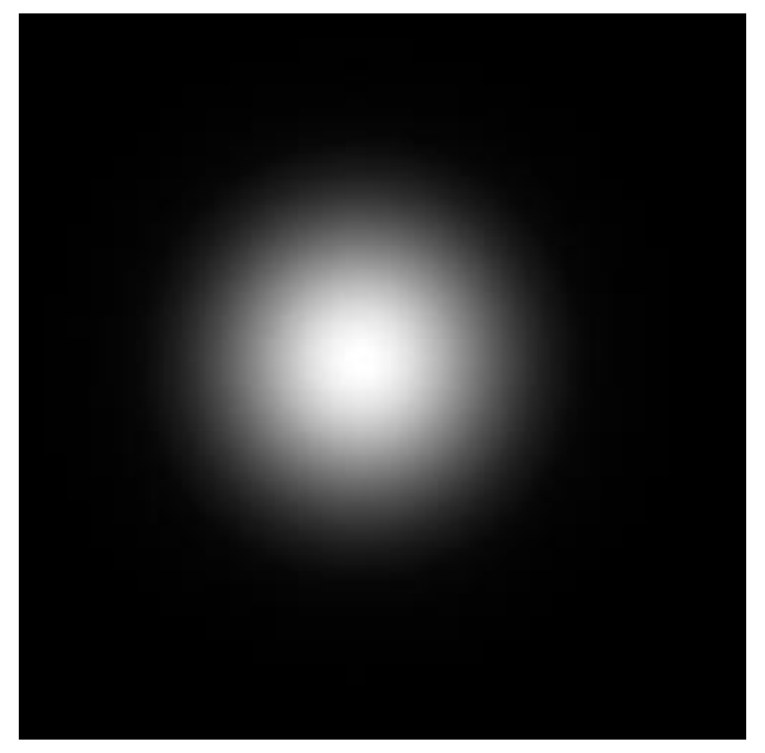

Figure 4 – Sun Texture

This texture can be sampled by using the normal of the water surface. If the normal lands on the bright part of the texture, then the ground is lit, if it lands on the black part, then its not.

Finally, the black parts need to be made **transparent** by using the sum of the red, green and blue values of the current pixel clamped between 0 and 1 as the alpha value of the current pixel.

Texture Distortion:

Texture distortion is used to modify the normal vectors of the water plane in order for the effects described above to appear **jittery** as if caused by the **movement** of the water surface.

This involves the use of a **normal map** and a **flow map**.

4 – RESULT

Reflection:

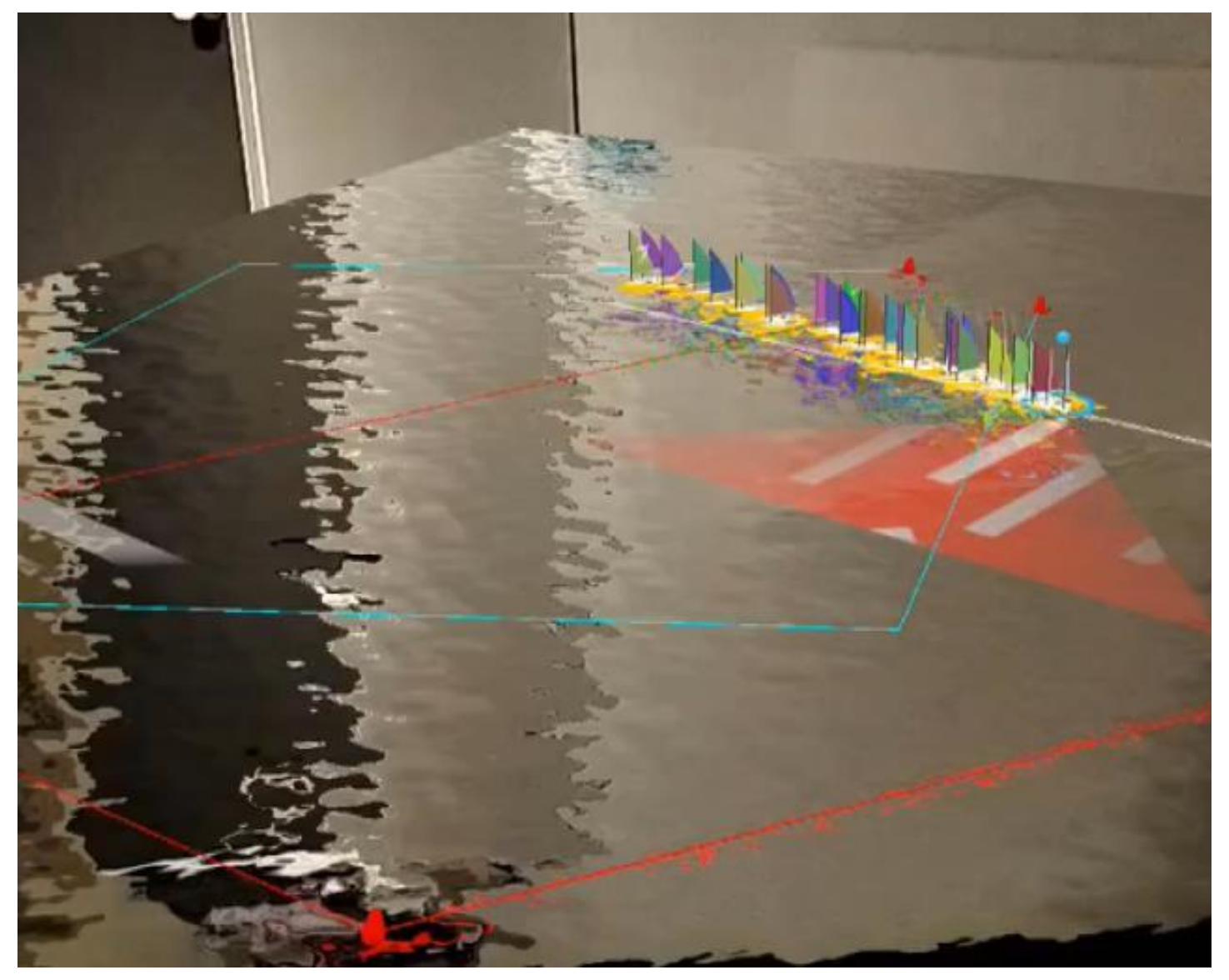

Figure 5 – Reflection in AR

Refraction:

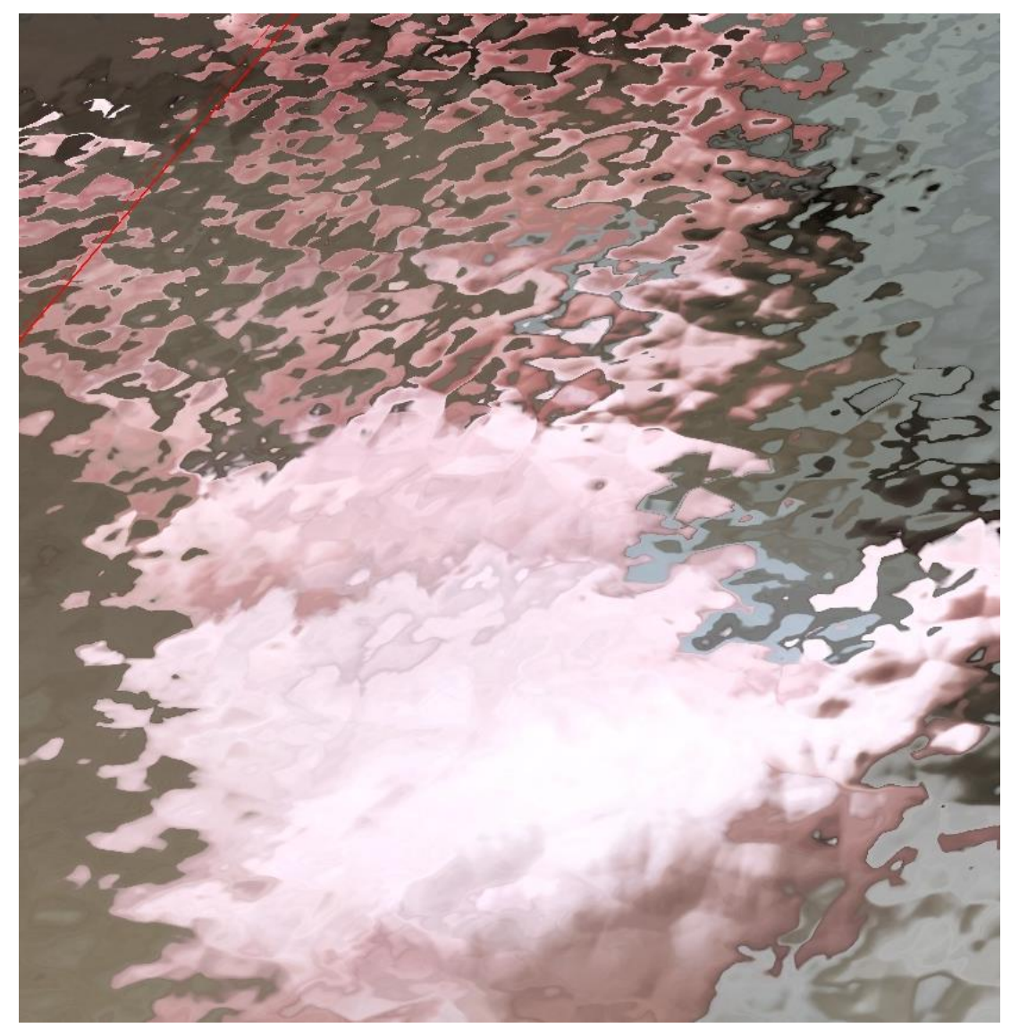

Figure 6 – Refraction in AR

Caustic:

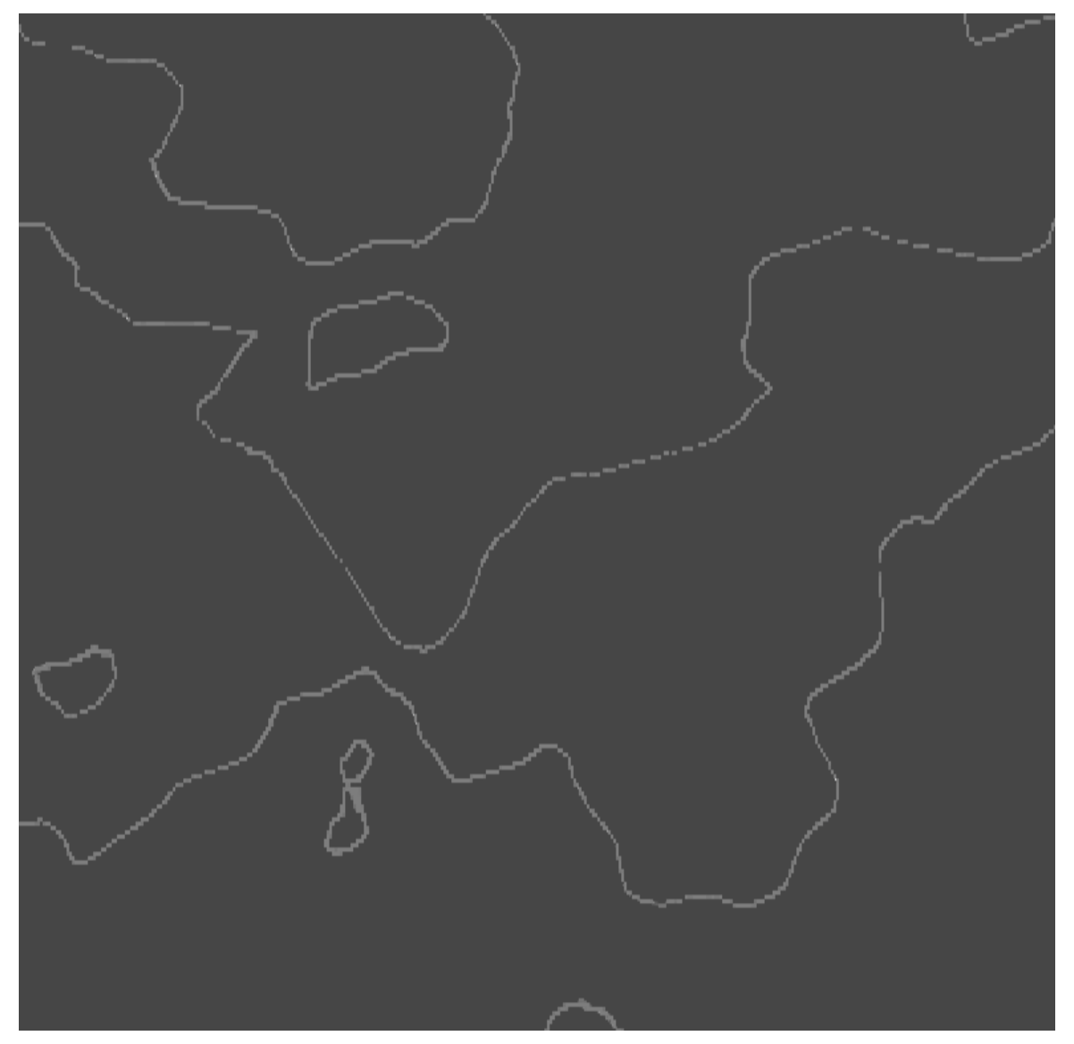

Figure 7 – Caustic texture

Performance:

Effects	Min fps	Max fps	Average fps
Reflection	24	30	27
Refraction	26	30	29
Caustics	28	30	30
Texture Distortion	27	30	29

Table 1: Fps for individual effects.

Together:

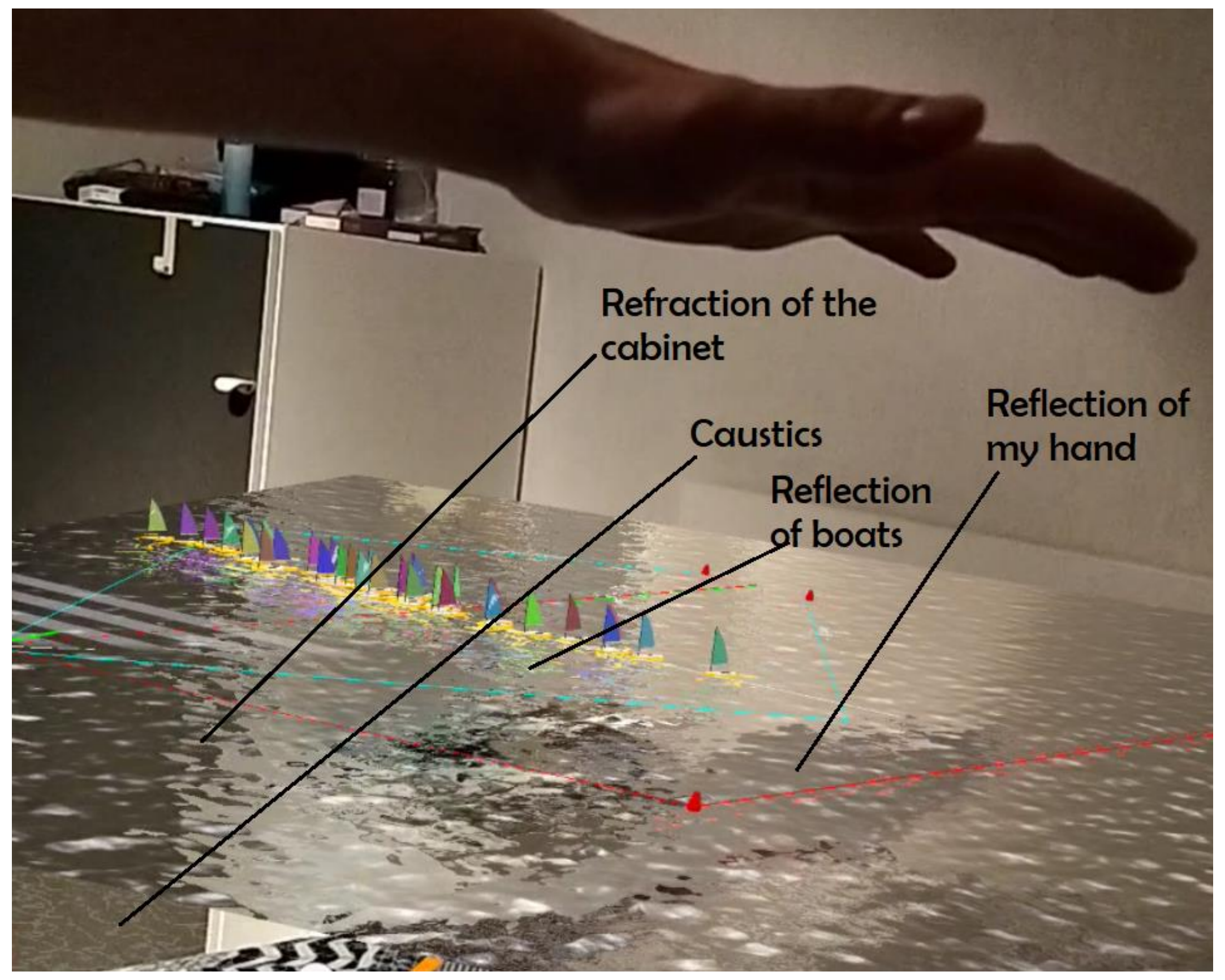

Figure 8 – All effects in AR

5 - DISCUSSION

No use of depth information. Incorporating depth information into the effects described here might require rewriting them as they are not extendible with depth information.

Reflections look worse as the camera looks down at the water plane.

Reflections, refractions and caustics are all physically inaccurate as an accurate implementation would require ray tracing.

Reflections use two cameras. This means two different renders each frame. This can be reduced by rendering the reflections in a lower resolution. This won't be as noticeable as it would be on a smooth plane because the texture will be distorted.

The distortion caused by refraction is the same for all objects without any regards to the angle they are observed from.

6 - CONCLUSION

The new water shader looks visually compelling and interactive.

Every effect mentioned here is implemented without the use of depth information. This is because the water should look visually compelling even on mobile phones that do not support this feature.

Future Work:

To improve the reflections, the user could be asked to take pictures of all 6 directions of their environment or to scan their room with photogrammetry. This could be mapped to a cube map texture and used in combination with reflection probes to calculate reflections which would look more accurate than the current implementation. However, this approach would not reflect anything dynamic such as the user's hand.

*. https://www.georgiaaquarium.org/wp-content/uploads/2018/09/yellowtail-snapper-2-1440x454.jpg