

COMP5411 Rendering Project Proposal: Rendering Thin Film Interference on Soap Bubbles

Mucong DING, 20323458
mcding@connect.ust.hk
Hong Kong University of Science and Technology

Abstract—The computer graphics simulation of real-world optical effects is of great interest. In this project, I focus on the thin film interference of soap bubbles. Soap bubbles have fascinating optical properties which are difficult to be simulated by simple shading algorithm. Although generally, a shader which takes reflection, refraction and Fresnel effect into considerations can already produces acceptable results[1], there are many other interesting effects which are omitted. Among them, the dominant one is the thin film interference. Since the thickness of soap bubble film is around hundreds nanometers, thin film interference greatly affect the appearance of a soap bubble. In this proposal, we introduce a formula of the thin film reflectance $R(\theta, \lambda, d)$ which not only depends on incident angle θ , but also varies with wave length λ and thin film thickness d . Base on that, we can implement a shader which also handles interference effects, using *webgl* and *three.js*. Moreover, we propose some approaches to approximate the dynamics of bubble thickness d , especially for the draining and sloshing effects. This makes the interference effect more obvious to see and more lifelike.

I. A SKELETON EXAMPLE

three.js provide a very simple example which use a self defined shader to render soap bubbles [1]. As we can see from the Fig. 1, that this shader use slightly different dynamic texture mappings to approximate the spectral splitting effects due to refraction.

Although this dynamic texturing already makes those spheres look like bubbles, it is very inadequate and not lifelike. Apart from the



Fig. 1. Soap bubble Demo Example[1]

fact that it does not simulate real reflection and refraction with Fresnel effect, it mistakes the origin of "spectral texture" on the bubble surface. In general, since the thickness of bubble is not larger than 1 micrometer, the dominant effect of refraction is only a small displacement. The real cause of "spectral texture" is the thin film interference.

II. THIN FILM INTERFERENCE

A complete model which simulate interference must takes the wave length λ and thickness d as parameters. In general, the radiance at point P still consists of the transmitted and reflected components, as follows,

$$L_P(\lambda) = (1 - R(\lambda, \theta, d))L_{it}(\lambda) + R(\lambda, \theta, d)L_{ir}(\lambda) \quad (1)$$

But the formula of reflectance look more complex [2],

$$R(\lambda, \theta, d) = 2R_{\perp}^2 \frac{1 - \cos \delta}{1 + R_{\perp}^4 - 2R_{\perp}^2 \cos \delta} + 2R_{\parallel}^2 \frac{1 - \cos \delta}{1 + R_{\parallel}^4 - 2R_{\parallel}^2 \cos \delta} \quad (2)$$

Where δ is also a function of λ , θ and d ,

$$\delta(\lambda, \theta, d) = \frac{4\pi}{\lambda} nd \cos \theta \quad (3)$$

With these equations, we are able to render the complete refraction, reflection and interference effects.

III. DYNAMICS OF THICKNESS d

In order to make the interference effect more obvious and lifelike, we should also consider the non-uniform distribution and dynamics of thickness d on the bubble surface. In general, two dominant effects should be considered.

The first one is the vertical gradient of thickness due to draining and earth gravity. This makes the thickness distribution look like Fig. 2.

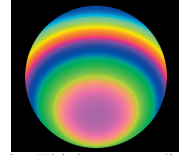


Fig. 2. Thickness gradient [3]

The second one is due to the perturbation of the environment around, there are random sloshing on the surface. When taking this into considerations, the thickness distribution is more like Fig. 3.

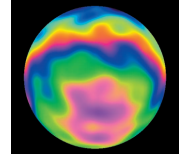


Fig. 3. Thickness sloshing [3]

IV. OBJECTIVE AND EXPECTED RESULTS

The main objective is to extend the shader in the skeleton example described in Sec. I using the radiance formula described in Sec. II and also simulate the distribution and dynamics of thickness d on the bubble surface according to Sec. III. The original shader will be completely changed to implement the ideas described in this proposal, while the rest parts, which make extensive use of *three.js* library, will remain the same.

The expected rendering result should exhibit lifelike interference textures on the surface, similar to the Fig. 4. I hope that we can optimize the algorithm and still achieve real-time rendering as in the original example. However, it is likely that we have to reduce the number of bubbles in the scene to speed up the rendering.



Fig. 4. Expected rendering results [1]

REFERENCES

- [1] three.js, "three.js bubble demo with fresnel effect," https://threejs.org/examples/webgl_materials_shaders_fresnel.html, 10 2017, [Online; accessed 09-Nov-2017].
- [2] K. Iwasaki, K. Matsuzawa, and T. Nishita, "Real-time rendering of soap bubbles taking into account light interference," in *Proceedings Computer Graphics International, 2004.*, June 2004, pp. 344–348.
- [3] A. Glassner, "Soap bubbles. 2 [computer graphics]," *IEEE Computer Graphics and Applications*, vol. 20, no. 6, pp. 99–109, Nov 2000.