Physically Correct - Part II

Assigned: 1/29/15
Due: 10:30am, 2/5/15

In this assignment, you will be implementing a shader that renders a water droplet on a flat surface. Here are some reference images (best viewed on a computer monitor):

Grading:

For a grade of a C (75):
Implement a fragment shader that performs physically correct raytracing with the following features:

- Your scene must contain at least 1 light source, at least 1 water droplet, at least 1 surface (that the water drop is resting on).
- Handle Fresnel reflection, refraction.
- Correct bumping during raytracing, i.e. does not intersect the same geometry it was traced from.
- Handle attenuation (as light passes through a glass sphere, it gets less and less bright).

For a grade of a B (85):

Accurately reproducing certain physical phenomena (soft shadows, caustics, scattering) is too computationally expensive for a shader. Enhance your image by adding physical or psychovisual hacks, i.e. any means to fake better light transport, or improve the water drop shape, or aesthetic of the scene.

- Watch this video [http://youtu.be/sgo_M0_Zwik](http://youtu.be/sgo_M0_Zwik) ("How to paint a water drop in Acrylics"). Despite huge simplifications on how light transport works, the artist is still able to render what is unmistakably a water droplet.
- Extend your scene to render caustics and soft shadowing.
- Create an interesting surface that the water droplet rests on (graph paper, denim, sampler2D texture).
- Please provide macros at the top of your shader code for enabling / disabling these features.

For a grade higher than a B (85+), implement something that impresses us. We encourage you to think of something creative that no one else in the class will try.

Here are some ideas to get you thinking (you certainly don’t need to do all of them):

- Implement the following phenomenon to influence the shape of your water droplet.

![Figure 1](http://example.com/figure1.png)  

*Figure 1:* Shown are a droplet on a flat hydrophobic surface; a drop on a rough hydrophobic surface (full contact of water with substrate); and a superhydrophobic droplet (both air-liquid and substrate-liquid contacts).
• Clever tricks to improve the quality of the image or reduce render times (without sacrificing much quality).
• Discard all the physics of raytracing and go full NPR - render a painterly shader of your water drop.

As a hint, IQ’s article on multires ambient occlusion discusses low-frequency AO, which has a pretty cool “color key-like” effect.

• Add camera motion to the scene, such that the NPR features still looks good while the camera moves.
• Depth of field.
• Motion Blur.
• Do something like this

• How many points you receive will be at the discretion of the staff, depending on how cool & unique your implementation is. It is possible to receive a grade higher than 100/100 on this assignment.

Support Code & Hints:
- No support code is provided.
- For the NPR component of this project, we encourage you to look online for examples of psychovisual “hacks” that artists use to fool the human brain.
  - A yellow sun tends to cast purple shadows - why?
  - Seemingly random strokes of oil paint in an impressionist painting come together at a macroscopic level to form material & texture.
  - Stars are blotted out by the light of the moon.
- Read 32.3 in the book for how to get the bumping right.
- Area Lights - look at section 32.4 in the book for some hints.
- Attenuation of Light - radiance $R$ at one point becomes $Re^{-dk}$ after travelling a distance of $d$ meters through the material, where $K = \text{extinction coefficient of the medium, in units of “1/meters”}$. For opaque materials, $K$ is infinite, but it’s irrelevant because you should never have transmitted rays in an opaque material anyway. What’s a reasonable value for kappa? A value of 1 per meter means that light traveling through a 1 meter piece of the material will only have about $\frac{1}{3}$ the original radiance when it exits. That’s probably a reasonable value to use for ordinary glass, although perhaps 0.5 or 0.25 might be better.

**Handing In:**

Your handin must contain the following:
- README, containing a link to your ShaderToy URL and a list of things you implemented. If you implemented any non-physical hacks (B grade requirements), explain to us how they work and why they approximate the real thing. If you don’t tell us why it works, we won’t give you points for it.
- A copy of the source code for your fragment shader. We will paste this into shadertoy for grading, unless you specify explicitly in your README that you wish to have it graded using glsl-playground.
- Provide a reference image of how your scene is inspired.