Realistic Real-Time Rendering of Eyes and Teeth

Matt Chiang and Graham Fyffe

ICT Technical Report No. ICT-TR-01-2010. September, 2010.

1 Real-time rendering of faces

We demonstrate realistic real-time renderings of a static face along with integrated eyeballs and teeth under both point lighting and image based lighting. Example renderings are shown in Figure 6. The software prototype runs on nVidia graphics cards with 1 GB of GPU memory. This includes a Quadro FX 3800 used for development, as well as the consumer-level GeForce GTX 285 used for testing.

1.1 Environment Lighting

The rendering software supports direct point lighting and also environment lighting. We approximate environment lighting based on captured lighting environments. We use an image-based environment illumination technique to light the face with a specified environment map. An example is shown in Figure 1.The environment map is *pre-filtered* with several blur kernels, for diffuse integration, glossy specular integration, and semi-glossy specular integration.

1.2 Diffuse Shading Component

1.2.1 Soft Shadow Terminator

For directional light sources, diffuse shading is commonly performed using the Lambertian shading model: $D = \alpha_d \max(0, n_d \cdot l)$ where α_d is the diffuse albedo, n_d is the diffuse surface normal, and l is the direction towards the light. However, this model has a hard shadow terminator. The materials of a human face do not exhibit hard shadow terminators, due to their subsurface scattering properties. To approximate this effect, we replace the

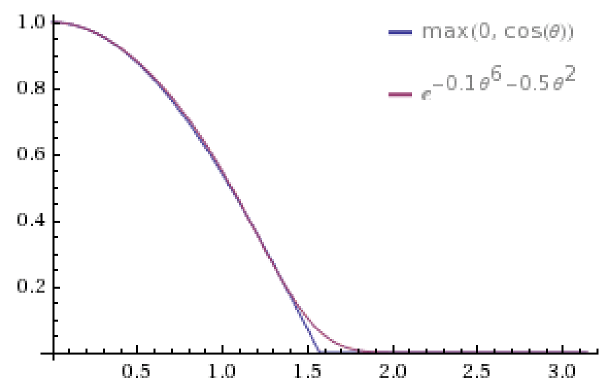

Figure 2: Lambertian shading model versus diffuse shading model with soft terminator.

Lambertian shading model with an alternate model having a soft shadow terminator:
$D = \alpha_d \exp\left(-\frac{1}{2}(\cos^{-1}(n_d \cdot l))^2 - \frac{1}{10}(\cos^{-1}(n_d \cdot l))^6\right)$.
This model is nearly equal to the Lambertian model, but it approaches zero more smoothly (see Figure 2).

For environment light sources, diffuse shading is performed using a normal-based look-up into the environment map that has been pre-filtered with a diffuse blur kernel: $D = \alpha_d E_d(n_d)$, where E_d is the diffuse-filtered environment map.

1.2.2 Hybrid Normal Mapping

The rendering software adapts the ICT Graphics Laboratory's *hybrid normal map* rendering technique [2] to the GPU in order to provide photoreal renderings of human faces in real-time. Essentially, a different diffuse normal map is provided for each color channel, and the diffuse

Figure 1: The real-time rendering software demonstrating environment illumination using the Uffizi Gallery environment map from www.debevec.org.

lighting component is computed independently for each color channel. Additionally, there is a separate specular normal map for the specular lighting component. Hybrid normal maps approximate the effects of ambient occlusion and subsurface scattering, which are effectively *baked in*. If hybrid normal maps are not available, they may be computed by the *Virtual Light Stage* technique (described later), or by a combination of ambient occlusion and subsurface scattering rendering (described next).

1.2.3 Ambient Occlusion

Ambient occlusion is the soft shadowing that occurs on surfaces by the partial occlusion of ambient illumination. We implemented a real-time screen-space ambient occlusion method to compute occlusion on the eyeballs, and in and around the teeth and the surrounding mouth region. The final pixel color is computed by multiplying the shaded color of the material by the ambient occlusion term in a compositing fragment shader.

1.2.4 Subsurface Scattering Approximation

Since hybrid normal maps are not available for eyes and teeth, we investigated and implemented methods to approximate subsurface scattering in the sclera (the white of the eye) and the teeth.

For eyes, we assume 10% of the light entering through the eye socket is scattered inside the eyeball and re-emitted evenly from any point on the sclera. Thus we blend the geometry normal with the direction from the front of the face:

$$n_d = \frac{0.9n_g + 0.1n_f}{|0.9n_g + 0.1n_f|}$$

where n_g is the sclera geometry normal and n_f is the direction pointing out the front of the face. For the iris, we used the *Virtual Light Stage* technique to compute hybrid normals as observed through the refractive cornea.

For teeth, we assume 90% of the light scatters by diffuse normals computed from an approximation of the teeth model geometry as a cylindrical surface:

$$n_d = \frac{0.9n_c + 0.1n_g}{|0.9n_c + 0.1n_g|}$$

where n_g is the teeth geometry normal and n_c is the cylinder geometry normal. We also computed hybrid normal maps for the teeth using the *Virtual Light Stage* approach, however these normals were not used in the software.

1.3 Specular Shading Component

1.3.1 Blinn-Phong with Fresnel term

For directional light sources, specular shading is performed using the normalized Blinn-Phong shading model with a Fresnel term based on Schlick's approximation:

$$S = \alpha_s \frac{(k+2)(k+4)}{8\pi(2^{-k/2}+k)} \max(0, n_s \cdot h)^k F;$$

$$F = \frac{(i-1)^2 + 4i(1-\max(0, n_s \cdot v))^5}{(i+1)^2}$$

where α_s is the specular albedo, n_s is the specular surface normal,$h = \frac{l+v}{|l+v|}$, v is the direction towards the camera, k is the specular exponent (800 for teeth, 50 for eyes, and a blend of 40 and 100 for skin), and i is the index of refraction of the material (1.33 for eyes, 1.4 for teeth and skin).

For environment light sources, we use a reflection-direction-based look-up into the environment map that has been pre-filtered with a blur kernel based on the specular exponent: $S = \alpha_s E_k(2(n_s \cdot v)n_s - v)F$, where E_k is the filtered environment map for specular exponent k.

1.4 Skin

We scanned a face using the technique of [2] to obtain the skin geometry of the face and the diffuse and specular normal and albedo maps. We use the hybrid normal mapping method for diffuse illumination. We do not use ambient occlusion for the skin because these are already baked into the hybrid normals, so we render a mask to exclude the skin from the ambient illumination compositing step. Specular illumination is computed with the Blinn-Phong model and Schlick Fresnel term, using the captured specular normal map and specular albedo map.

1.5 Eyeball

The eyeball is modeled as generic eye geometry. The texture map of the whole eye (sclera and iris) is based on a number of photographs of the eye with the subject looking in different directions, with and without polarization. The

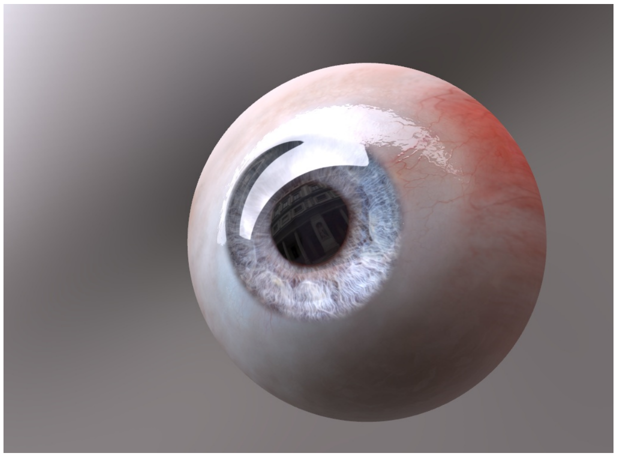

Figure 3: Real-time eyeball rendering, incorporating environment map illumination, subsurface scattering approximation in the sclera, initial blood vessels texture, and specular reflections off of the bulbar conjunctiva.

photographs are loaded into Photoshop, and each one is aligned and warped to place the iris as a circle at the center of the image, and then the photographs are composited together. The iris texture is cut out from the center of the whole eye texture.

We use ambient occlusion and hybrid normal mapping for diffuse illumination on the eyeball, using the approximated hybrid normals described earlier. Specular illumination is computed with the Blinn-Phong model and Schlick Fresnel term, using the eye geometry normals and a specular albedo of 1. Figure 3 shows a close-up of an eyeball rendered with our implementation, under image-based illumination.

Eyeballs present additional challenges for photoreal rendering, so we implemented the following features:

Control of pupil radius. The radius of the pupil adapts automatically to the illumination incident on the pupil.

Corneal refraction. We implemented a shader to approximate the refraction of light at the surface of the cornea. The refraction is computed according to formulae described in [1], using corneal geometry characterized in [3].

Iris caustics approximation. The surface normal of the iris is obtained using the Virtual Light Stage technique. However, to obtain an effect that looks like caustics, the normal is adjusted half-way towards a concave normal. This effect is based on artistic observation: the concave normal shifts the brightest part of the diffuse illumination to the side of the iris that is opposite the direction of the light source, which is qualitatively similar to the behavior of caustics. The concave normal is obtained by inverting the corneal geometry normal in x and y, and scaling down the corneal geometry normal in z.

Iris darkening from refraction. We apply an artistically designed iris darkening function to darken the iris around its edges, approximating the "refraction function" in [1].

Color texture for blood vessels in the conjunctiva. We included a color texture map for the blood vessels in the conjunctiva, modeled from a set of photographs through artistic manipulation.

Bump mapping on the conjunctiva. We apply a bump map to the specular normals of the conjunctiva, combining structured noise with blood vessel relief extracted from the sclera color texture. The structured noise map was created in Photoshop and the bumps from the sclera texture are computed using a standard method for bump mapping based on a height image.

1.6 Teeth

The geometry for the teeth was obtained by scanning a dental cast of real teeth. We scanned plaster casts of the upper and lower teeth made using standard dental casting techniques. We scanned the casts with a structured-light scanning system to produce a 600,000-triangle mesh from 16 merged scans for each set. We manually remeshed the teeth scans to have far fewer triangles and better topology. We included color texture maps for the teeth and gums, based on artistically aligning photographs of the real teeth to the scanned geometric model in Maya.

We use ambient occlusion and hybrid normal mapping for diffuse illumination on the teeth, using the approximated

Figure 4: Real-time teeth rendering, incorporating image-based lighting, specular normals based on dental cast geometry, diffuse normals based on a cylindrical surface, and screen-space ambient occlusion computation.

hybrid normals described earlier. Specular illumination is computed with the Blinn-Phong model and Schlick Fresnel term, using the dental cast geometry normals and a specular albedo of 1. Figure 4 shows a set of teeth rendered with our implementation, under point light illumination.

2 Virtual Light Stage

We developed a technique called *Virtual Light Stage* for computing hybrid normal maps from virtual objects, by simulating a Light Stage within a physically-based rendering application, and performing a virtual Light Stage capture session. We perform offline renderings (these can include sophisticated light transport effects such as occlusion, interreflections, and subsurface scattering) under gradient illumination conditions. Then we compute diffuse photometric normals from these images in the same way as [2]. These diffuse normals can then be used together with specular normals from the object geometry to perform the same hybrid normal lighting calculation used for the face scan data from real photographs. We applied this technique to the teeth and mouth regions in the rendering software.

2.1 Implementation Details

In our virtual light stage rendering development we explored physically-based subsurface scattering simulation. We used the Mental Ray physical subsurface scattering shader in Maya to apply measured optical properties of objects to generate physically-based renderings (Figure 5).

We also made use of rendering into an object's texture space, rather than a typical camera projection space. One of the side benefits of the virtual light stage approach is that it need not be constrained by the limits of what devices can be built, but rather any representation we can compute.

We integrated the subsurface scattering shader setup with the image-based lighting approach for gradient illumination for virtual light stage acquisition in MAYA 2010, rendering into the object's texture space. We applied this to the teeth with surrounding mouth area, to generate normal maps for the hybrid normal map rendering approach. These hybrid normal maps incorporate baked-in light transport effects such as ambient occlusion, interreflection, and subsurface scattering (Figure 6), and incur only a minimal additional rendering cost in a real-time rendering system.

References

[1] Guillaume Francois, Pascal Gautron, Gaspard Breton, and Kadi Bouatouch. Image-based modeling of the human eye. *IEEE Transactions on Visualization and Computer Graphics*, 15(5):815–827, 2009.

[2] Wan-Chun Ma, Tim Hawkins, Pieter Peers, Charles-Felix Chabert, Malte Weiss, and Paul Debevec. Rapid acquisition of specular and diffuse normal maps from polarized spherical gradient illumination. In *Rendering Techniques (Proceedings of Eurographics Symposium on Rendering 2007)*, pages 183–194, 2007.

[3] Ko Nishino and Shree K. Nayar. Eyes for relighting. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, pages 704–711, New York, NY, USA, 2004. ACM.

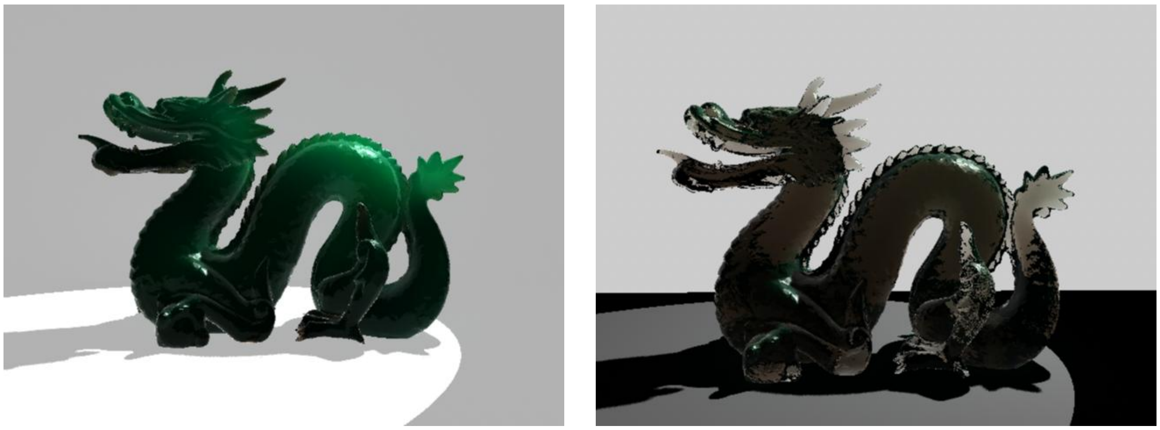

Figure 5: Rendering tests of subsurface scattering using two different sets of physically measured optical properties (left and right).

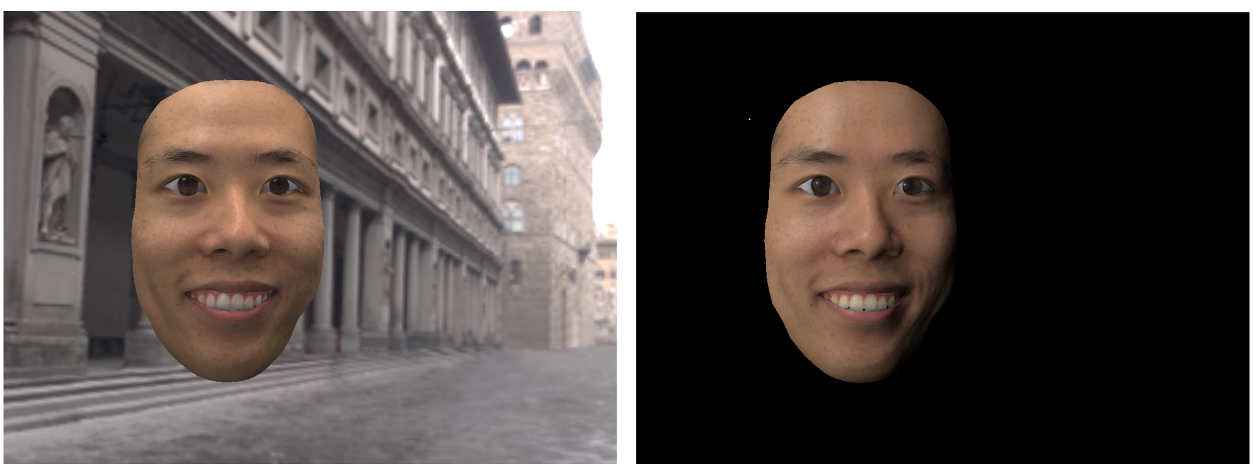

Figure 6: Example real-time face renderings with eyes and teeth, under image based lighting and point lighting.