New and Used Cars

Philip Child*
Pixar Animation Studios

Junyi Ling†
Pixar Animation Studios

Alex Seiden‡
Pixar Animation Studios

1 Introduction

We faced several challenges shading the characters for the movie Cars 2. The cars needed a rich, vibrant look. Yet the shading model had to be flexible enough to provide our lighters with non-realistic controls when needed. And it had to mimimize the impact of compute-intensive ray tracing.

This talk presents three shading innovations developed for the film: a layering method that is efficient and flexible, a procedural metal-flake shading model that approximates the complex properties of modern carpaint, and a new model for rendering micro-scratches.

Figure 1: Finn and Mater in *Cars 2*.

2 Shader Layer Consolidation

Most surfaces consist of several different materials layered over each other. For example, a car's body surface may contain many materials like steel, paint, rust, dirt, etc. with independent illumination properties. For Cars 2, this was a prohibitive expense because of the complexity of the shaders and the amount of ray tracing required. Traditionally, we would calculate the illumination parameters for each layer, then run the illumination model on every layer, and composite the results. It would be more efficient to calculate the illumination parameters, composite those parameters, then illuminate the composited parameters. This way, the illumination model is run only once.

*e-mail: jling@pixar.com
†e-mail: philipchild@pixar.com
‡e-mail: aseiden@pixar.com

We could have asked our shading TDs to completely change the way they author shaders: instead of using abstractions like "rust" or "dirt" or "chipped paint", they could manually manage the generation and combining of the parameters. But this would have made shader authoring and maintenance much more complex. We also wanted a technique that would be minimally invasive to our existing shading code libraries.

The two methods do not usually give identical results, but we found ways to make the differences minimal. Additionally, we allow the shading artist to easily switch consolidation on and off, and to consolidate certain layers while allowing others to call their own illumination loop. Render time savings was significant. Depending on the lighting complexity of the scene, and the layering complexity of the surfaces, time savings ranged from 10% to over 50%.

3 CarPaint Shading Model

Metallic car paint is composed of a clear coat lacquer layer on top of a fine metal flake layer. It is non-trivial to reproduce because the individual flakes are often far smaller than a single pixel, and yet cannot be allowed to filter away.

We model the clear coat as a traditional dielectric surface. For the metal flake layer, we designed a procedural flake signal generator that closely couples with the illumination model. It interpolates two underlying BRDFs, eaching cooresponding to a different level of detail. At close-range, a highly specular BRDF approximates the response of individual metal flakes. As the viewer moves farther away from the surface, a secondary illumination model is added. The combined model approximates the integrated reflectance under the calculated filter region, until it converges into a secondary high-roughness specular BRDF which is distance-invariant. We were able to achieve "metal flake" looks, even when the individual flakes are much smaller than a pixel.

4 Scratches Shading Model

Shiny surfaces covered by fine multi-directional scratches can give the appearance of concentric circles, perfectly aligned around an isotropic specular highlight. Rather than simulating this effect in a physically accurate manner, our approach involves defining a set of textures featuring straight scratches. Of these textures, only a weighted subset (those which are perfectly aligned to the gradient of the specular highlight's fall-off) are evaluated.

We achieve this by computing two vectors in tangent space. The first is a specular gradient aligned vector, derived from the surface's reflection vector. The second is the orientation of the textures' alignment. The angle between the two vectors dictates the weighting of an individual texture's contribution. In addition, the first vector is used in conjunction with a surface tangent to define a polar coordinate space around the center of the highlight. An added advantage of this technique is that having a specular-relative coordinate space we can apply patterns to our specular falloff.

SIGGRAPH 2011, Vancouver, British Columbia, Canada, August 7 – 11, 2011.
ISBN 978-1-4503-0921-9/11/0008