Project Report

Chi-Wei Tseng; Muhammad Ahmed Riaz

March 21, 2015

1 ABSTRACT

We implement a real-time rendering system for Lambertian and glossy surfaces. Our system is capable of handling soft shadows and radiance self-transfer under dynamic environment lighting. In order to achieve real-time rendering, we use a representation which captures an object's response to distant illumination. The response signal, being pre-computed and stored densely over the surface of the object, encodes how light from the environment is redistributed by the geometry into exitance radiance. At run-time, we apply the response to a parameterized environment radiance map, and obtain the transferred radiance over the object surface. The computation is done under spherical harmonic basis for enhanced performance. This allows us to switch between different environment maps and rigidly rotate objects with full global illumination effects without substantial computational overhead.

2 SPHERICAL HARMONICS REPRESENTATION

The surface light field of a Lambertian surface is always slowly varying. Ramamoorthi et al.[1] show that three bands of spherical harmonics are sufficient to approximate surface light field to obtain a final render within a 3% pixel error on average. Glossy surfaces produce responses of relatively higher frequency, but can be well handled by including 2 additional bands of spherical harmonic basis.

Functions represented with spherical harmonics manifest good mathematical properties, which makes their use convenient. These includes rotational invariance (analogous to shift invariance of Fourier basis), and capabilities of carrying out convolution, projection and rotation operations directly in its parameter space.

3 Rendering Pipeline

Transfer vectors and matrices represent how incident radiance of a vertex is transferred to the exitance radiance. Diffuse surfaces transfer radiance uniformly in all directions, thus a coefficient for each basis is enough to encode its response. However, glossy surfaces transfer radiance inhomogeneously to each direction. Therefore, a transfer matrix is needed instead of just a vector. Our approach to pre-compute transfer vectors and matrices is in section 4.

3.1 Diffuse Surface

We represent the environment map with 9 spherical harmonic basis coefficients. As such, the pre-computed transfer vector for each vertex is an 9-dimensional entity that holds the response stimulated by an input of a spherical harmonic basis. Computing radiance from a surface point becomes a dot product between the environment map and the transfer vector.

$$L_o^{(p)} = \sum_{n=1}^{9} M_n^{(p)} \cdot (L_{env})_n \tag{3.1}$$

where $L_o^{(p)}$ is the exitance radiance from vertex p, $\mathbf{M}^{(p)}$ is the transfer vector at this vertex, and $\mathbf{L}_{env}$ is the spherical harmonics projected environment map.

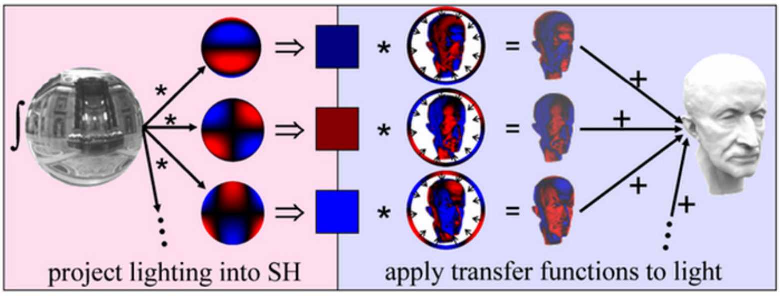

Figure 3.1: Spherical harmonic lighting coefficients (left) are modulated by the transfer vectors over the surface(middle), to produce the final shading (right).[2]

3.2 Glossy Surface

A pre-computed transfer matrix is a 9-by-9 matrix which produces the transferred surface radiance field when left-multiplied onto an environment map. To produce the final shading, the radiance in a specific viewing direction is computed by evaluating the spherical function at this direction.

$$(L_o'^{(p)})_m = \sum_{n=1}^{9} M_{m,n}'^{(p)} \cdot (L_{env})_n \tag{3.2}$$

where $\mathbf{L_o}'^{(p)}$ is the exitance radiance field from vertex p, $\mathbf{M}'^{(p)}$ is the transfer matrix at this vertex. In order to evaluate $\mathbf{L_o}'^{(p)}$ at the viewing direction $\mathbf{R}$, we use

$$L_o^{\prime(p)}(\mathbf{R}) = \sum_{n=1}^{9} (L_o^{\prime(p)})_n Y_n(\mathbf{R}) \tag{3.3}$$

where $Y_n(\mathbf{R})$ is the spherical harmonics basis n evaluated at the $\mathbf{R}$ direction.

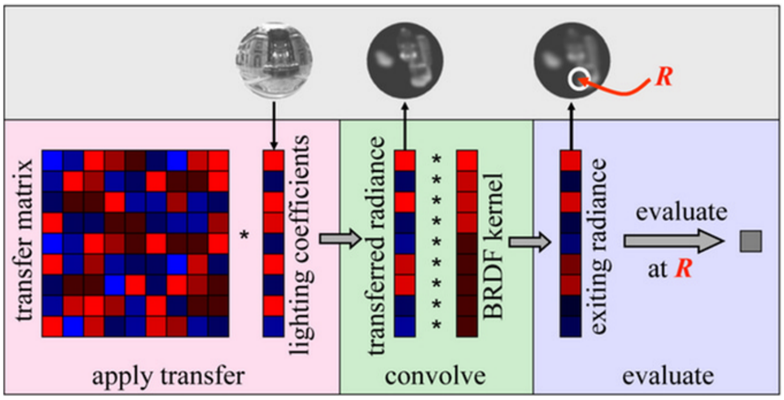

Figure 3.2: Stored on each vertex (left), the transfer matrix modulated the spherical harmonic lighting coefficients to give the transferred radiance field. The transferred radiance is then convolved with a BRDF kernel (middle) and evaluated in the viewing direction (right). [2]

3.3 Run-time Rendering Steps

We now describe the full run-time rendering steps of our system.

1. Rotate the environment map coefficients $\mathbf{L}_{env}$ to compensate the object's and camera's rotation [3].
2. Perform the linear transformation specified in previous two subsections at each vertex to obtain exit radiance.
3. For glossy surface, at each pixel, evaluate the exit radiance in the current viewing direction.
4. Sum the radiance contribution from diffuse and glossy transfer according to specified weights.

In our implementation, for diffuse surfaces, light transfer is evaluated in the fragment shader. For glossy surfaces, due to the memory limitations of the GPU, we perform light transfer on the CPU instead. To achieve high frame rates on CPU, we are constrained to using only 3 bands of spherical harmonics.

4 Radiance Transfer Pre-computation

Transfer vectors and matrices depend only on the geometry. Therefore, for our rigid body system, we pre-compute them to be used in the rendering pipeline. To integrate the response for each spherical harmonic basis at the vertex, we run a path tracer which samples the hemispherical domain of incident directions. For unoccluded rays, we sample the spherical harmonic basis and accumulate its value into the response. For samples rays that hit an object, we incorporate specific radiance transfer policies, which are explained below.

4.1 Diffuse Surface Self-Transfer

We implemented 3 radiance transfer policies that achieve different levels of photorealism.

- Local illumination. We always sample the spherical harmonic basis, regardless of any detected ray-object occlusion. There is no self-shadowing nor inter-reflection under this policy.
- Direct illumination. For sample rays that get occluded by objects, it is regarded that zero radiance is being transmitted from this direction. This policy enables self-shadowing effects.
- Full diffuse self-transfer. On hitting an object, the next level of sample rays are generated in the positive hemisphere from the intersection. This policy enables inter-reflection effects.

4.2 Glossy Surface Self-Transfer

Due to the directional dependence of radiance transfer on a glossy surface, in addition to record the total intensity of incoming light, we also need to record its direction. The directional distribution is represented as another set of spherical harmonic basis. For each sample ray, we compute its response for all spherical harmonic basis and accumulate them separately. This gives us a 9-dimensional vector for every spherical harmonic basis over the domain of the environment map.

5 Discussion and Conclusion

Pre-computed radiance transfer systems essentially trade off the flexibility of the scene for generic high quality real-time global illumination effects. In order to achieve this goal, it has to perform large amount of computations in able to produce all possible appearance variationS at run-time. This also leads to the side effect of taking up large amount of storage on disk or in memory. All of these aspects have limited the applicability of pre-computed radiance systems and also make them more difficult to implement.

Our pre-computed radiance system achieves real-time performance. It is able to produce high quality results under basic assumption of relatively rough surface or low-frequency environment map. Accurate global illumination can be observed in our system in the forms of

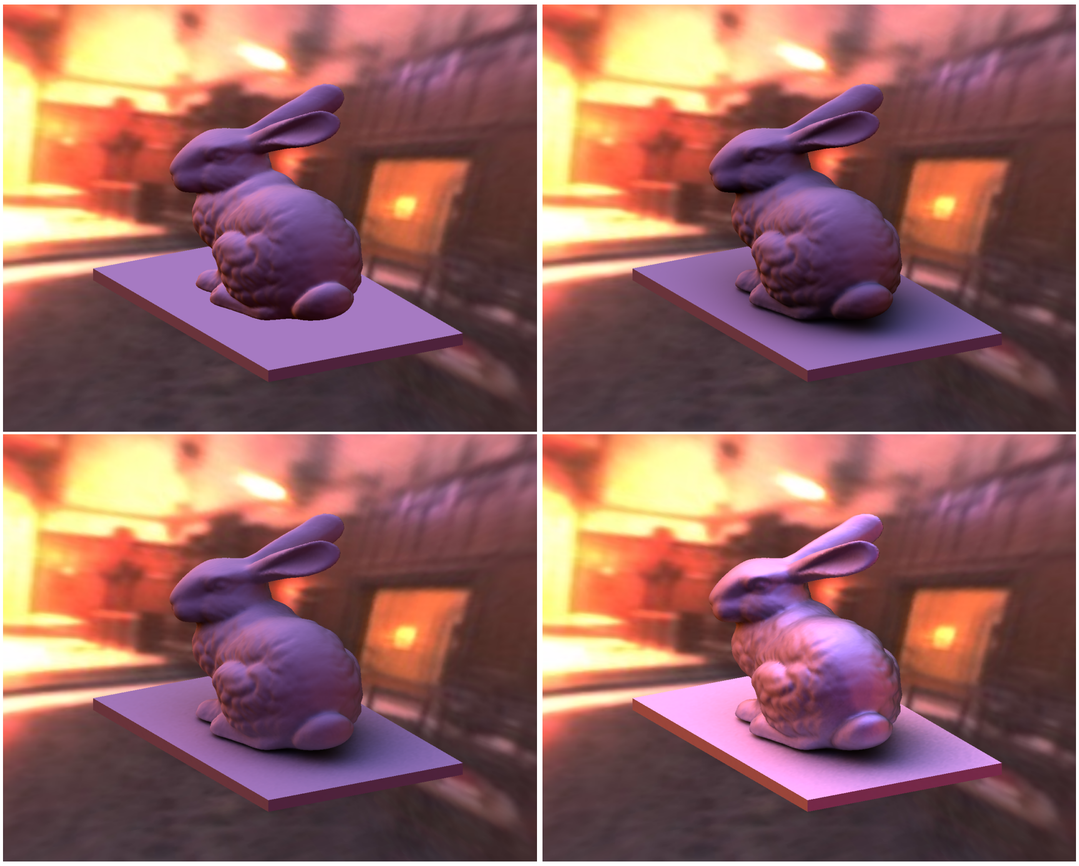

Figure 4.1: Radiance transfer policies. Top left: Local illumination. Top right: Direct illumination. Bottom left: Full diffuse self-transfer. Bottom right: Full glossy self-transfer.

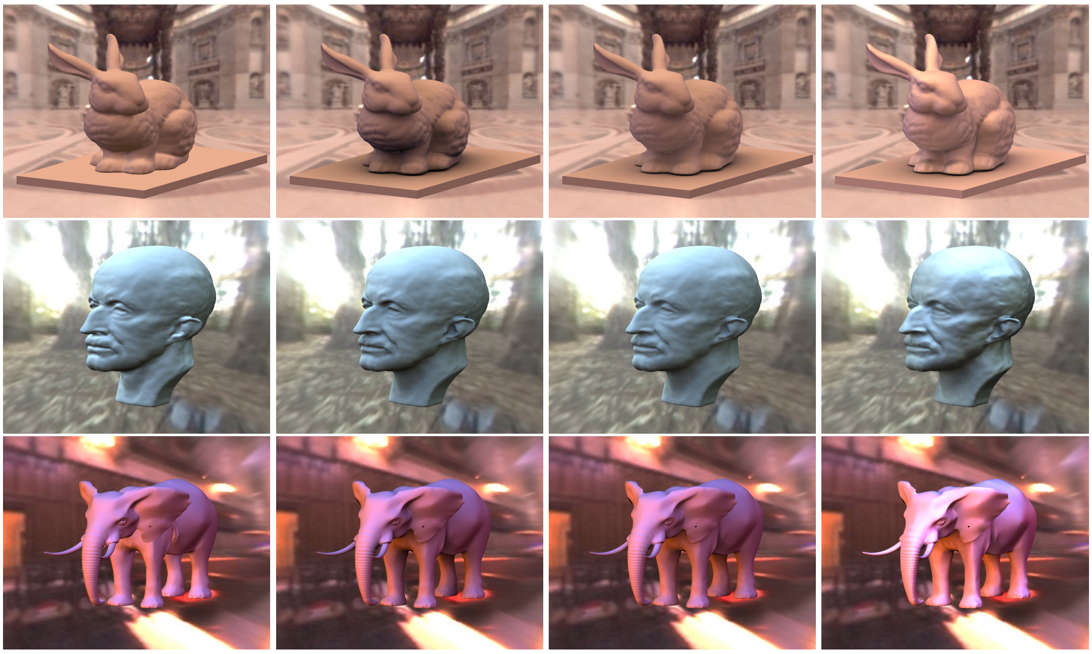

Figure 5.1: Radiance transfer policy comparison. From left to right: Local illumination, Direct illumination, Full diffuse transfer, Full glossy self-transfer

dynamic soft shadows, inter-reflections and caustics.
A video demonstration can be found at https://youtu.be/xNmlTkO0ki4

REFERENCES

[1] Ravi Ramamoorthi , and Pat Hanrahan. An efficient representation for irradiance environment maps. *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM, 2001.

[2] Peter-Pike Sloan, Jan Kautz, and John Snyder. Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. *ACM Transactions on Graphics (TOG). Vol. 21. No. 3.* ACM, 2002.

[3] Derek Nowrouzezahrai, Patricio Simari, and Eugene Fiume. Sparse zonal harmonic factorization for efficient sh rotation. *ACM Transactions on Graphics (TOG) 31.3 (2012): 23.*