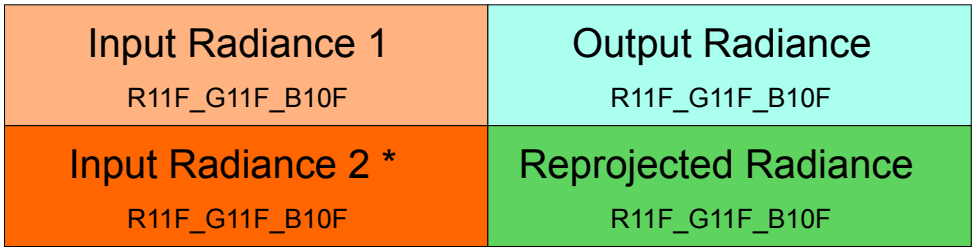
Deep G-Buffer Global Illumination Implementation Details

1. Memory Layout

1.1 Deep G-Buffer

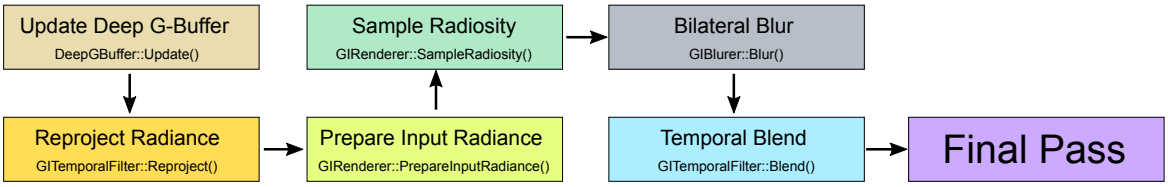
Totally **176 bits per pixel** are used by **2-Layer Deep G-Buffer**. Normal used **oct16 encoding**, which is mentioned in the origin paper. **Last Depth** stores the depth value of last frame, which is essential for single pass **Deep G-Buffer** generation.

1.2 Radiosity



Totally **128 bits per pixel** are used by **Radiosity Processing**. **2-Layer Input Radiance** are calculated with the direct light from **Deep G-Buffer** and the indirect light of previous bounce from **Reprojected Radiance**. **Output Radiance** is the place to store the radiosity result from the **2-Layer Input Radiance**. **Reprojected Radiance** is used in temporal filtering, with the reprojected **Output Radiance** at previous frame in it. **Input Radiance 2** is also used to be a temporary buffer for bilateral and temporal filtering.

2. Render Pipeline



2.1 Update Deep G-Buffer Pass

Before making a draw call, copy the existing **Depth 1** to **Last Depth**. Then we make a draw call, judging whether a surface should be discarded or not by comparing the current depth value reprojected to previous view and the value in **Last Depth**. **Depth 1&2** can be directly attached to **GL_DEPTH_ATTACHMENT**.

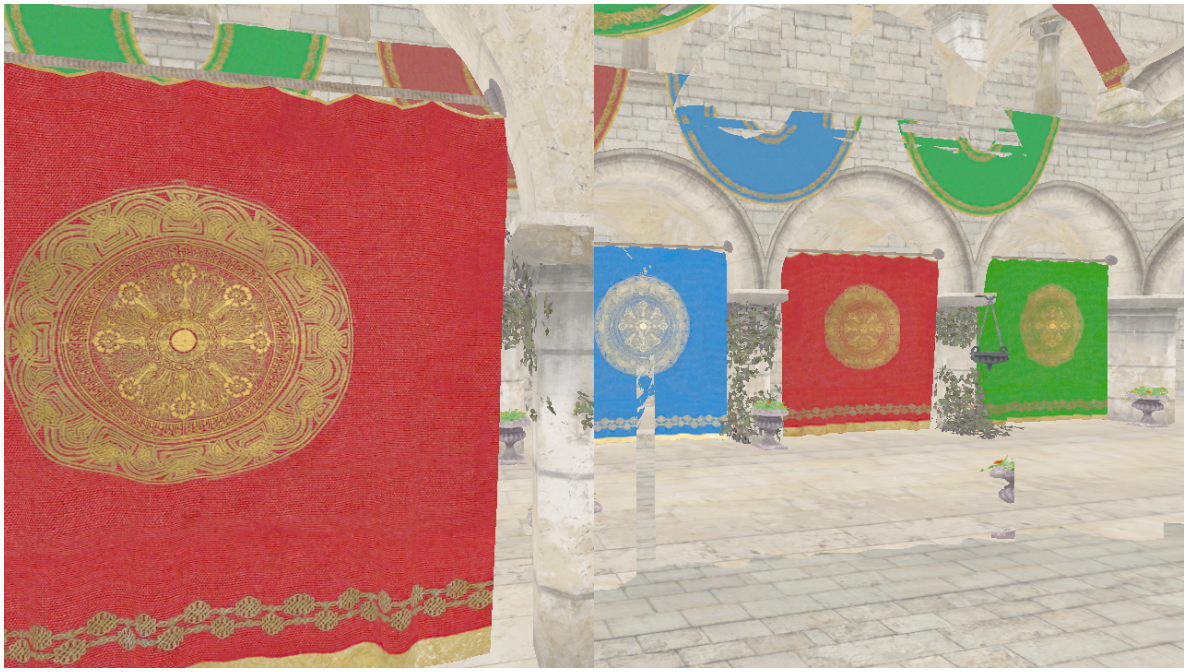


Figure 1: The left part is **Albedo 1**, and the right part is **Albedo 2**

2.2 Reproject Radiance Pass

Use **Output Radiance** as input (the final result of previous frame), reproject it to current frame's perspective, The reprojected result is placed in **Reproject Radiance**. **Last Depth** is used to judge occluded surface.

2.3 Prepare Input Radiance Pass

Calculate direct light for both layers (I only implemented direct sun light using a shadow map), the result is stored in **Input Radiance 1&2**. For the first layer, the value in **Reprojected Radiance** is added. In my implementation, I accumulated them with $E_{input} = E_{direct} + \frac{E_{reprojected} \cdot Albedo_1}{\pi}$

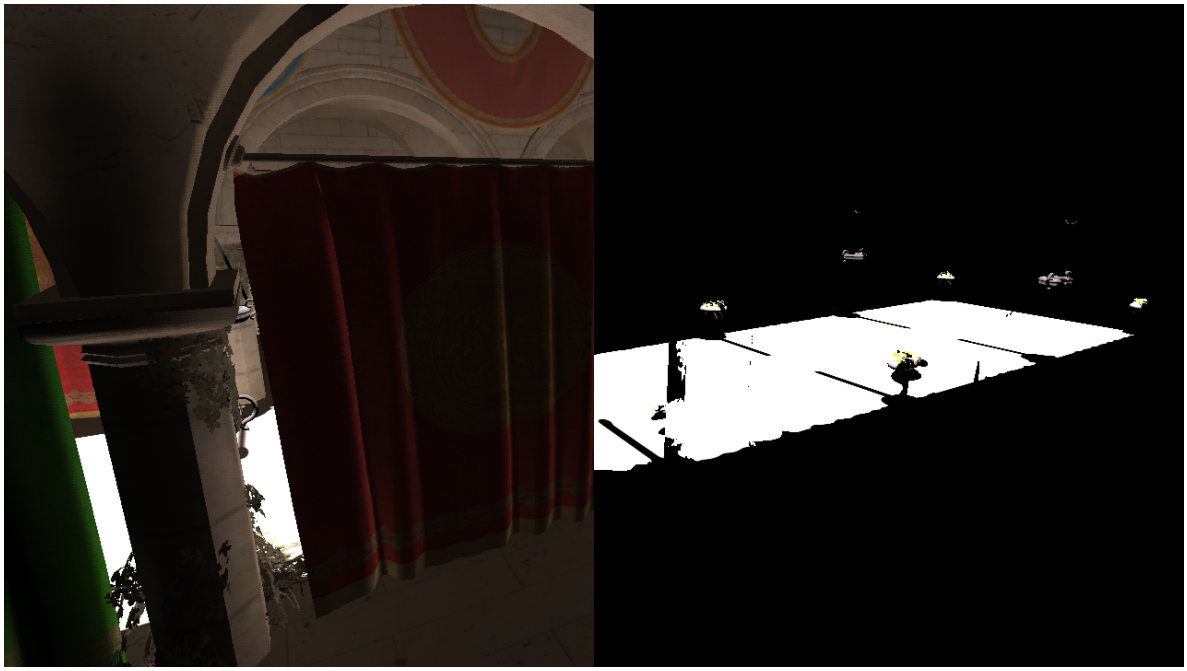


Figure 2: The left part is **Input Radiance 1**, which is accumulate with the value of last bounce. The right part is **Input Radiance 2**, which only has direct light.

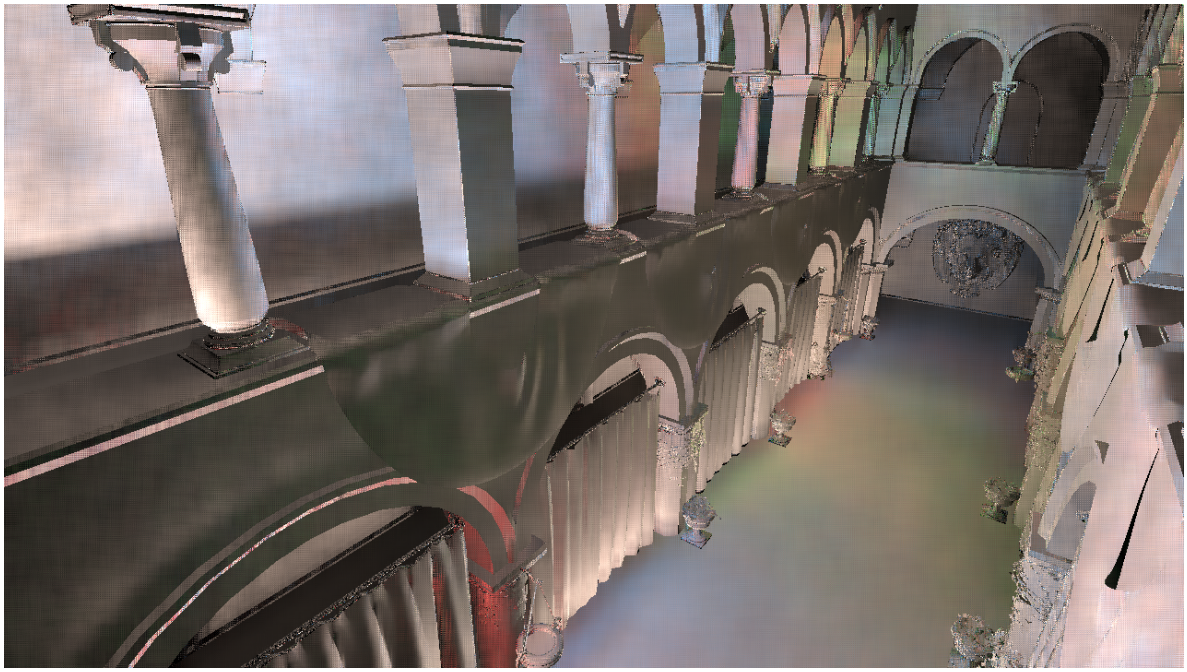
2.4 Sample Radiosity Pass

Sample radiosity with **Input Radiance 1&2**, store the result to **Output Radiance**. The details are described in the origin paper.

Figure 3: Unfiltered noisy result.

2.5 Bilateral Blur Pass

The blur is applied to **Output Radiance**. It has 2 passes, I first blur it horizontally and then vertically. Considering **Input Radiance 2** has no use after **Sample Radiosity Pass**, I use it for storing the temporary result (only horizontally blurred), the final result will be stored back to **Ouput Radiance**. **Depth 1** and **Normal 1** are used to calculated bilateral weight for a nice edge preservation.



2.6 Temporal Blend Pass

Just blend **Output Radiance** and **Reproject Radiance** with $E_{blended} = \alpha \cdot E_{reprojected} + (1 - \alpha) \cdot E_{output}$. I again store the blended result in **Input Radiance 2** and then copy it back to **Ouput Radiance**.

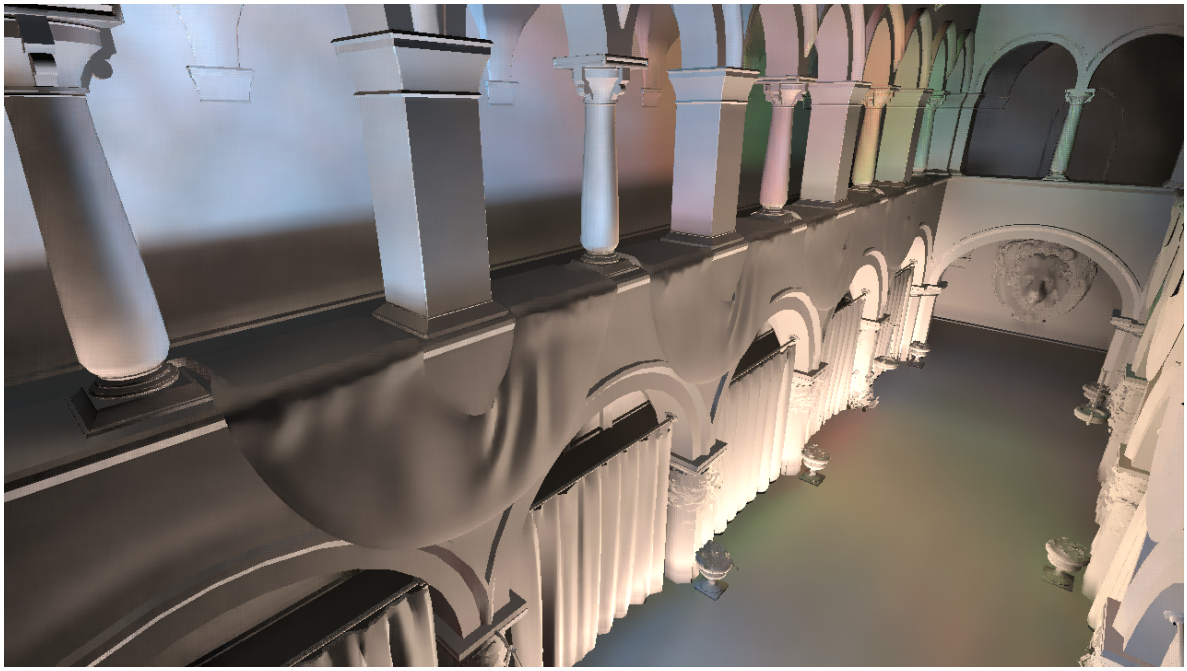


Figure 5: The result is much smother after this pass.

2.7 Final Pass

At last, I combine **Albedo 1**, **Input Radiance 1**, **Output Radiance** and generate the final image with $E_{final} = E_{input_1} + E_{output} \cdot Albedo_1$