

ReSTIR

In D3D12

Jilin Liu
Keyi Yu
Li Zheng

Milestone 2

1. Finished Biased ReSTIR
 - a. Falcor Base Code
 - b. Setup ReSTIR pipeline
 - c. Finished 5 steps in ReSTIR
 - i. Generate initial candidates *raytraced shader*
 - ii. Evaluate visibility for initial candidates *raytraced shader*
 - iii. Temporal reuse *pixel shader*
 - iv. Spatial reuse *pixel shader*
 - v. Compute pixel color *pixel shader*
2. Applied Intel Open Image Denoiser

RIS compared with trivial sampling

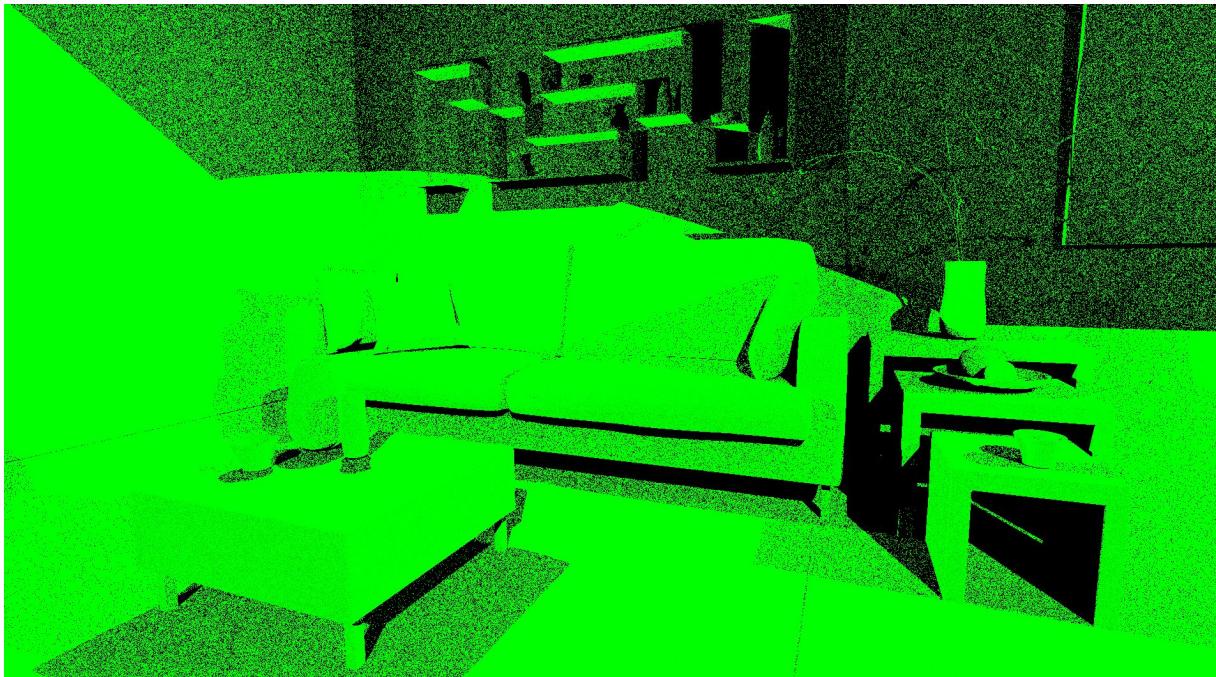


Randomly select a light sample



Resampled Importance Sampling($M = 32$)

ReSTIR: Evaluate visibility for initial candidates



Black: Shadowed Green: Visible

ReSTIR: Spatial Reuse

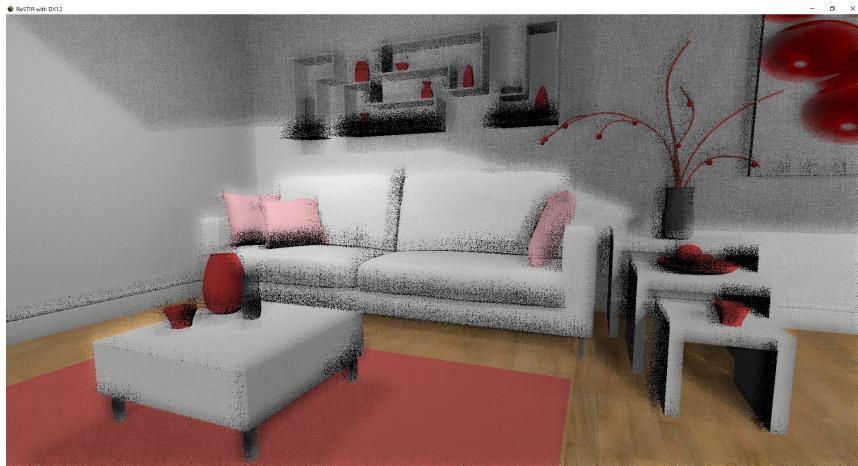


↔ RIS Only



Spatial Reuse On

ReSTIR: Spatial Reuse -- Reject Neighbors



Without neighbor rejection



With neighbor rejection

Reject the neighbor if either normal difference or scene depth exceed some threshold (10% of current pixels depth and 25° , respectively).

ReSTIR: Adding Temporal Reuse



Random Light Sample

Our ReSTIR

<https://youtu.be/yCGolgbfjk0>

Open Image Denoiser



Denoised

Challenges

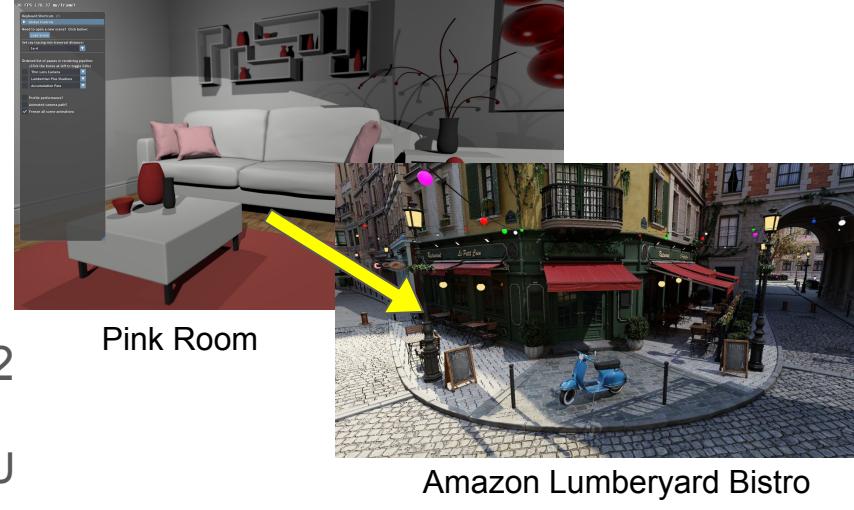
1. Getting used to DXR and HLSL.
2. Loading and sampling area lights.
3. Integrating Denoiser into our passes.

Next...

- Add complex scene with more lights
- Optimize the implementation in D3D12

Raytracing : ~25ms on CPU, ~7ms on GPU

Pixel Shaders: ~8ms in total



Pink Room

Amazon Lumberyard Bistro

Pink Room
i5-3600 @ 3.59GHz
16GB, RTX 2060 6GB

- Integrate denoiser into our pipeline

Questions?