DirectX Raytracing: ReSTIR

Megan Reddy (meganr28@gmail.com)

Github: https://github.com/meganr28/DXR-Pathtracer

Penn Engineering

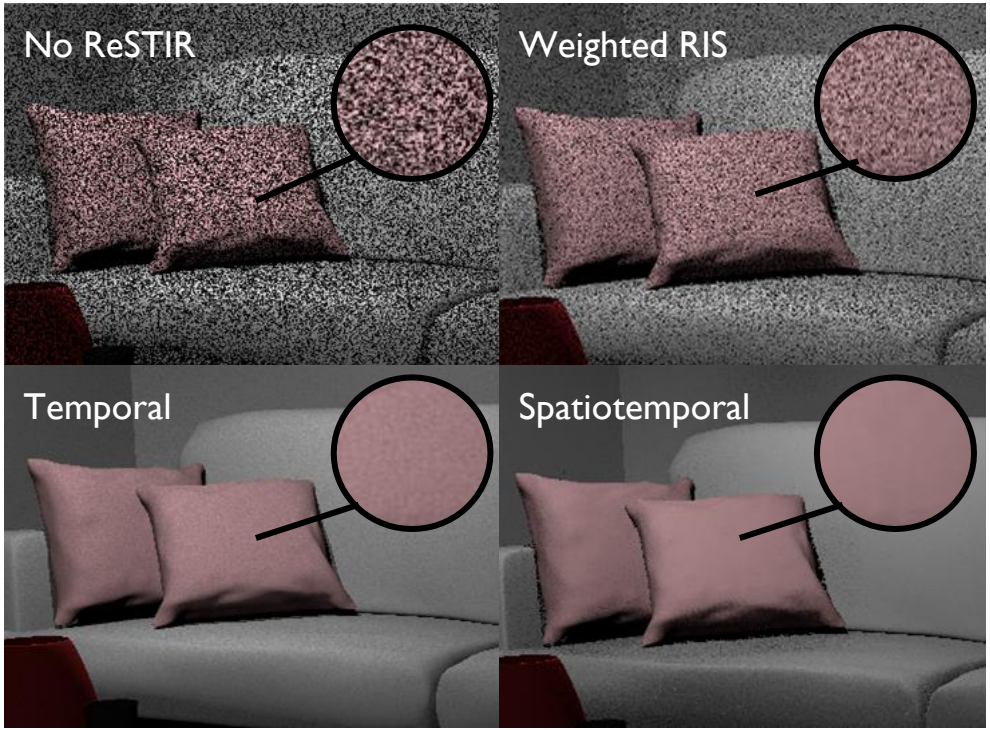
Project Overview

- Implementation of **ReSTIR** (Reservoir-based Spatiotemporal Importance Resampling) from SIGGRAPH 2020
- Investigated **ReGIR** (Reservoir Grid-Based Importance Resampling) from Ray Tracing Gems II
- **DirectX Raytracing**
- NVIDIA **Falcor** Framework

ReSTIR with A-Trous Denoising

Features

- DXR Pathtracer
  - Direct illumination, global illumination, Lambertian materials
- ReGIR (setup but not used)
- ReSTIR
  - Raytraced G-Buffer
  - Weighted RIS
  - Visibility reuse
  - Temporal reuse
  - Spatial reuse
  - A-Trous denoising (optional)



DXR Path Tracer

- Raytraced G-Buffer pass
- Antialiasing
- Depth-of-field
- Lambertian shading
- Environment mapping
- Direct and indirect illumination
- Tone mapping

Direct illumination

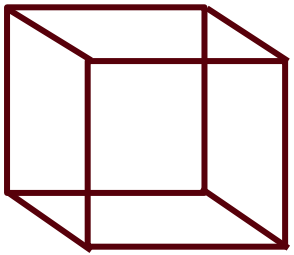
Indirect illumination

ReGIR

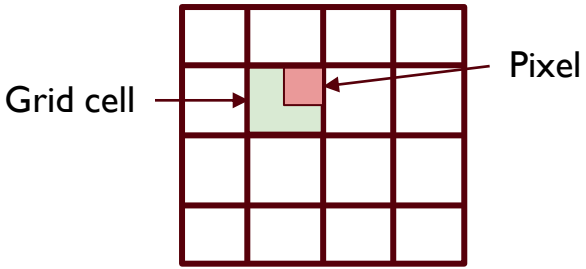
BuildCellReservoirsPass.cpp - create uniform world-space voxel grid (each voxel holds n reservoirs)

SampleLightGridPass.cpp - for each pixel, find nearest voxel and select final light sample

ShadeWithReservoirsPass.cpp - using final light sample, perform shading



Voxel (64-256 reservoirs)



ReSTIR - Weighted RIS

- Sample M = 32 initial candidates per pixel

No RIS - one sample per pixel

RIS - one sample per pixel

ReSTIR - Temporal Reuse

- **Backpropagate** to find position of pixel in previous frame
- Store **previous** frame's reservoirs in a texture
- Clamp previous reservoir M to at most **20x** current reservoir M

Weighted RIS

Temporal Reuse

ReSTIR - Spatial Reuse (Neighbor Count)

- Pick k neighbors within 30-pixel radius
- Skip neighbors that differ greatly in geometry/material to decrease bias

k = 5 (in paper)

k = 20

k = 50

ReSTIR - Spatial Reuse (Number of Passes)

- Can perform multiple spatial reuse passes
- Runtime is O(k + M) per-pixel and O(nk + M) for n iterations

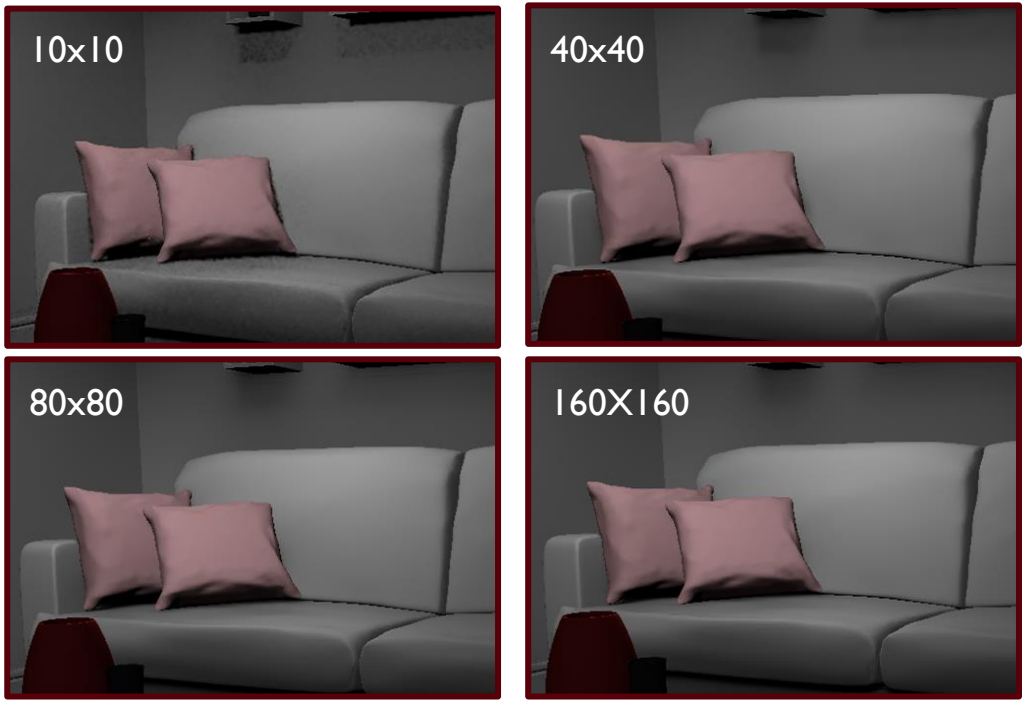
n = 1

n = 2 (in paper)

k = 5, r = 30

À-Trous Denoising

- Edge-Avoiding À-Trous Wavelet Transform for Fast Global Illumination Filtering (Dammertz et al. 2010)

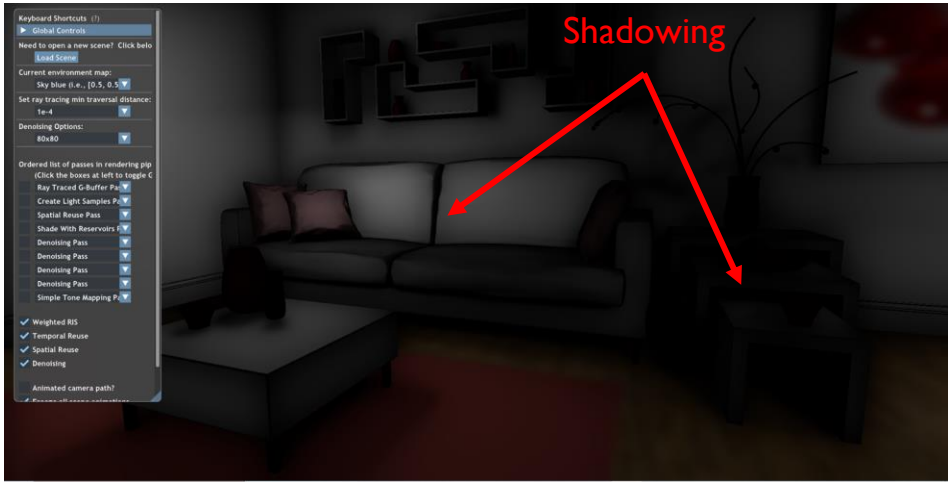


color_phi = 0.5
normal_phi = 0.0625
position_phi = 0.05

Biased vs. Unbiased ReSTIR (WIP)

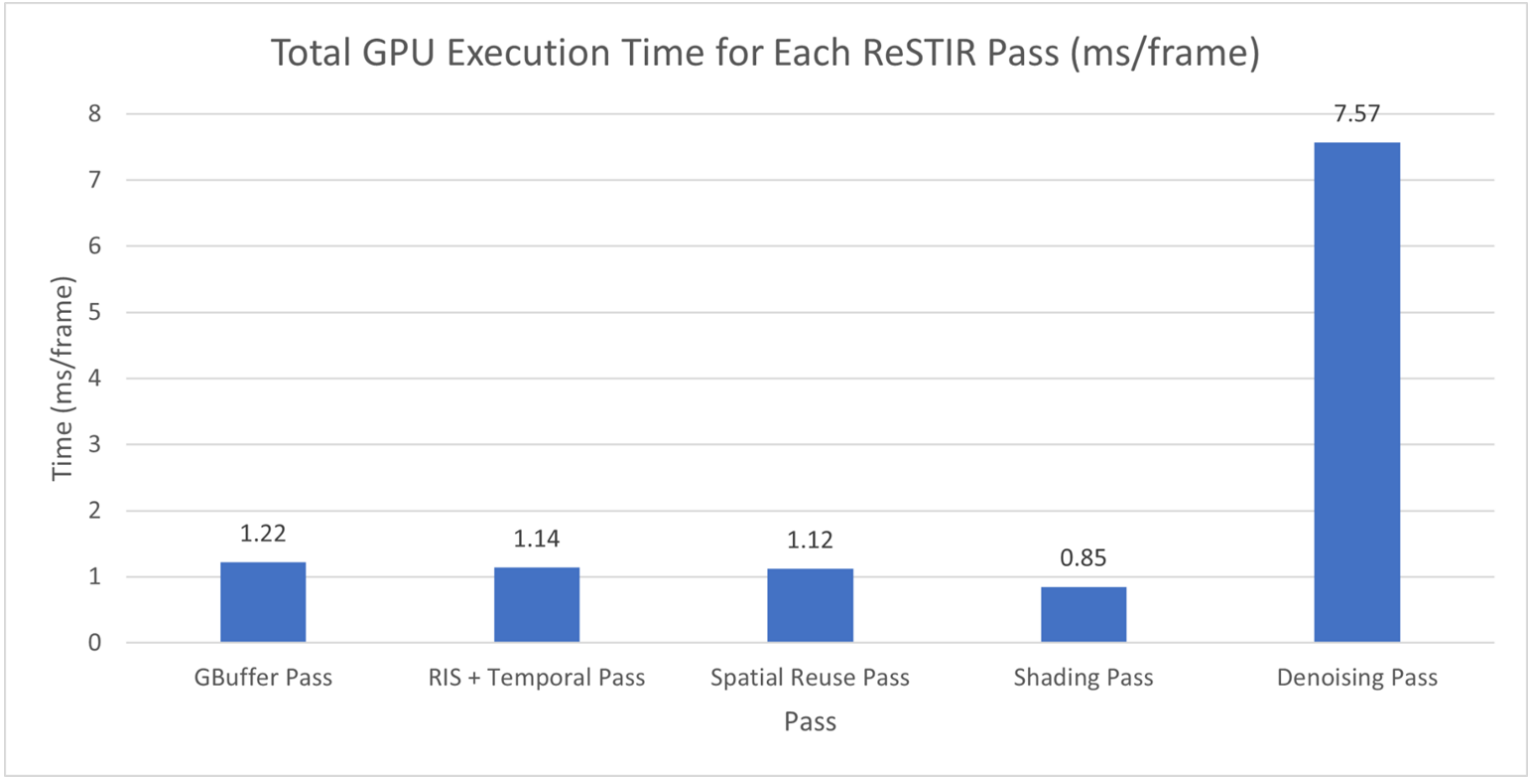
- Divide by number of candidates with non-zero PDFs (instead of by number of candidates M)

Biased



Unbiased

Performance (time spent in each pass)



Limitations and Future Work

- **Limitations**
  - Falcor framework (limited to one scene, limited GUI control)
  - Only supports point lights (no area or mesh lights)
- **Future work**
  - Support N > 1 samples
  - Dynamic lighting
  - Extend to world-space (e.g. ReGIR)
  - Global illumination (ReSTIR GI)
  - Better temporal coherence
    - Temporally Reliable Motion Vectors for Real-time Ray Tracing (Zeng et al. 2021)
    - Better temporal coherence for shadows, glossy reflections, and occlusions

References

- Spatiotemporal reservoir resampling for real-time ray tracing with dynamic direct lighting (SIGGRAPH 2020)
- Spatiotemporal Reservoir Resampling (ReSTIR) - Theory and Basic Implementation - Shubham Sachdeva
- Edge-Avoiding À-Trous Wavelet Transform for Fast Global Illumination Filtering (HPG 2010)
- A Gentle Introduction To DirectX Raytracing - SIGGRAPH 2018 Course
- Rendering Many Lights with Grid-Based Reservoirs - Ray Tracing Gems II Chapter 23