

20220802 Project Meeting

Juhyeon Kim

2022.08.02.

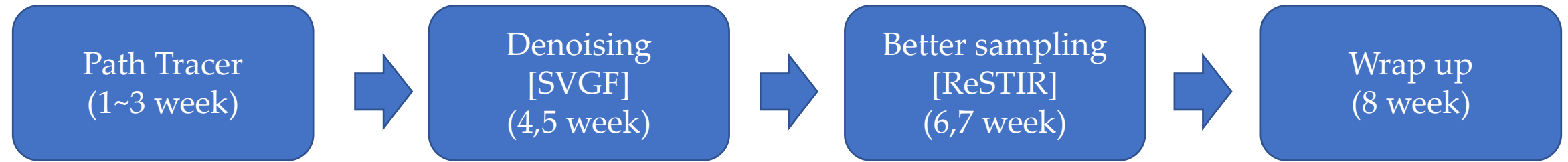
PEARLABYSS

Contents

- (5 Week) Work Progress
 - SVGF
 - Refactoring
 - GUI
 - ReSTIR
- TODO

Project Overview

- Project Goal : Implement a 1-spp real-time path tracer with denoising & better sampling technique.

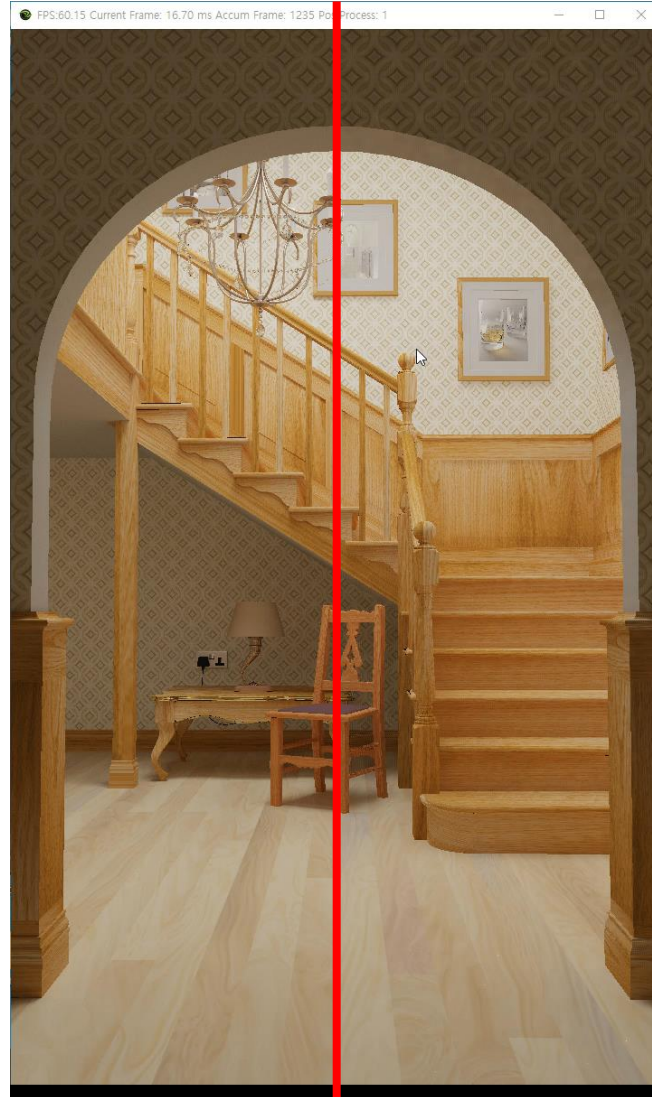


- Implement a **real-time path tracer using DX12**.
- Study basics of DX12 and physically based rendering.
- Implement **denoising** technique for a path-traced image.
- Choose to implement SVGF (2017).
- Implement **sampling quality enhancement** technique.
- Currently working on ReSTIR (2020)
- Try ReSTIR GI (2021) or ReSTIR PT (2022).
- Write a report.
- Wrap up the project.

Recall - (4 Week) Work Progress

- Implemented SVGF

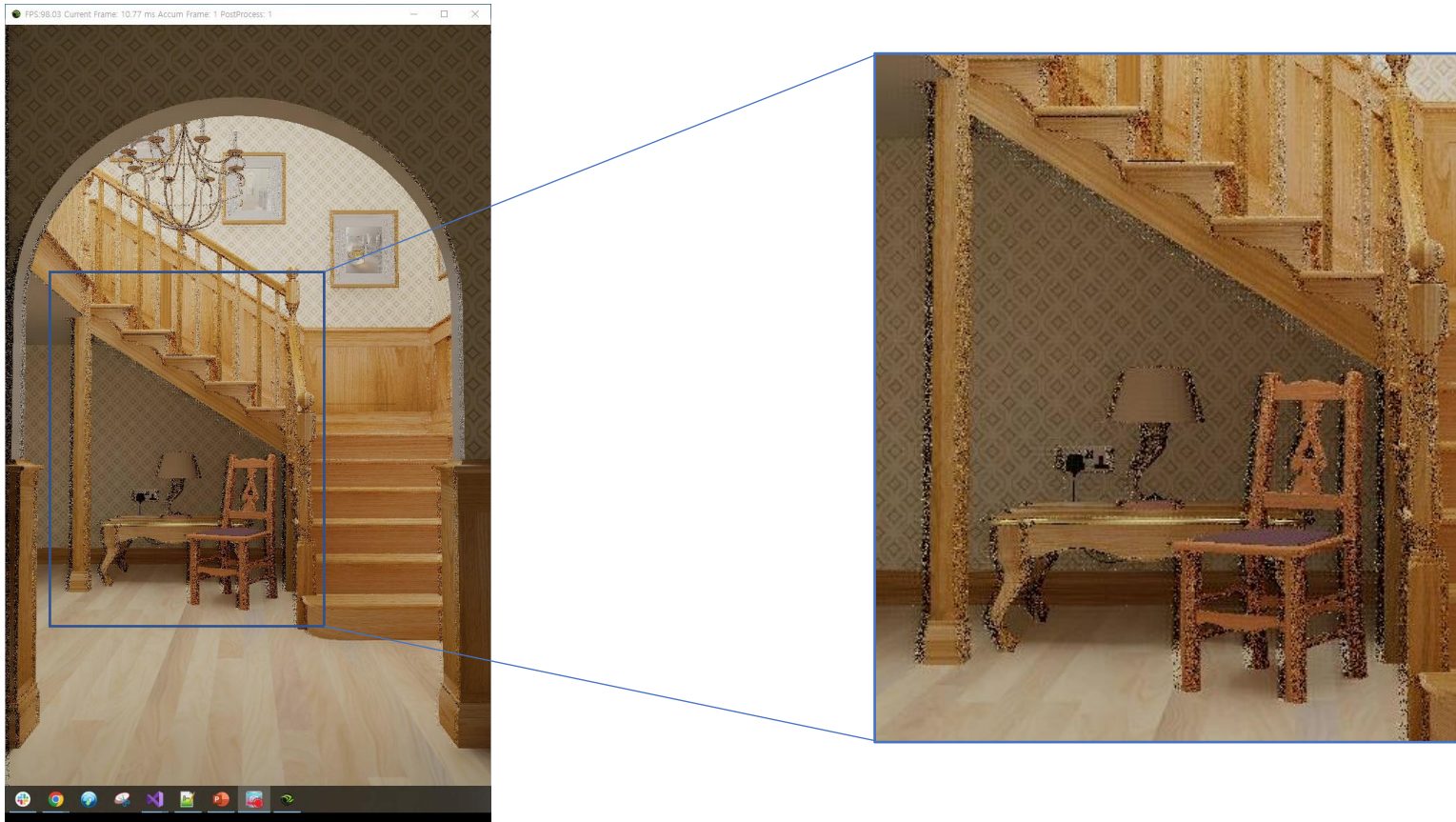
1 spp path traced



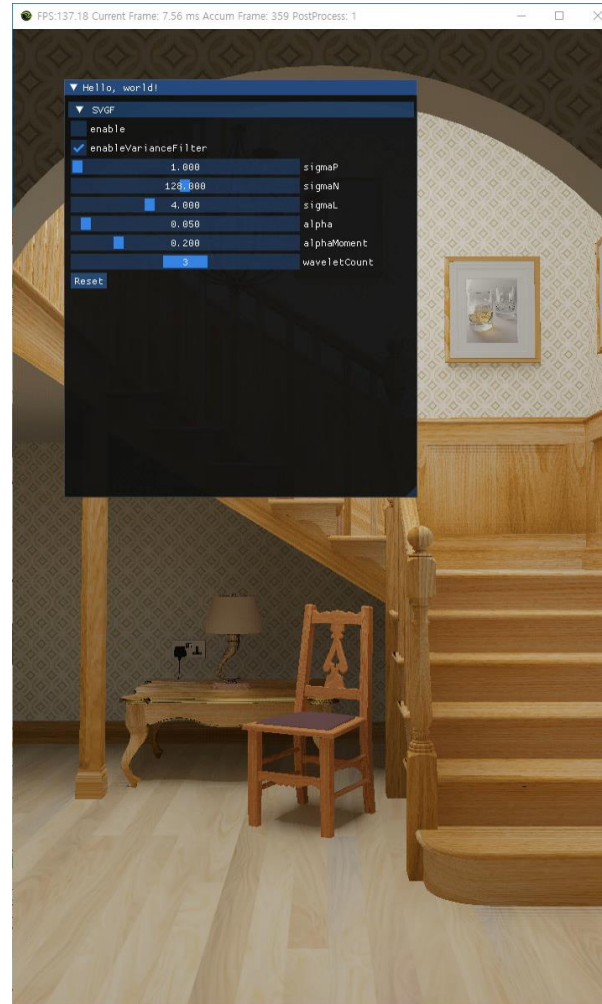
After applying SVGF

SVGF - Variance Filtering

- Just after disocclusion, temporal variance doesn't work.
- Solution → estimate variance spatially if history length < 4 .



- Solution → estimate variance spatially if history length < 4 .



- Implement ReSTIR (Benedikt, 2020) that improves light sampling for direct illumination.
- Implement RIS & WRS (Algo 3. in the paper)
- Implement temporal reuse (Algo 5. 14 line in the paper)

Algorithm 3: Streaming RIS using weighted reservoir sampling.

```

1 foreach pixel  $q \in \text{Image}$  do
2    $\text{Image}[q] \leftarrow \text{shadePixel}(\text{RIS}(q), q)$ 
3 function  $\text{RIS}(q)$ 
4   Reservoir  $r$ 
5   for  $i \leftarrow 1$  to  $M$  do
6     generate  $x_i \sim p$ 
7      $r.\text{update}(x_i, \hat{p}_q(x_i)/p(x_i))$ 
8    $r.W = \frac{1}{\hat{p}_q(r.y)} \left( \frac{1}{r.M} r.w_{\text{sum}} \right)$  // Equation (6)
9   return  $r$ 
10 function  $\text{shadePixel}(\text{Reservoir } r, q)$ 
11   return  $f_q(r.y) \cdot r.W$ 

```

- RIS : Resampled Importance Sampling
- WRS : Weighted Reservoir Sampling

Algorithm 5: Our algorithm for RIS with spatiotemporal reuse.

Input : Image sized buffer containing the previous frame's reservoirs

Output : The current frame's reservoirs

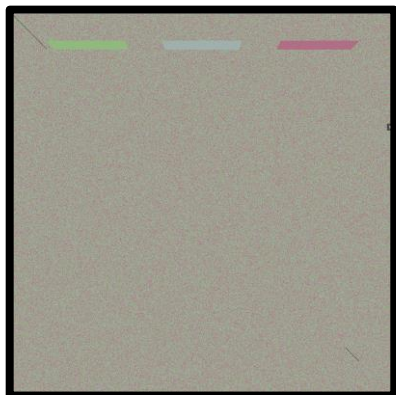
```

1 function  $\text{reservoirReuse}(\text{prevFrameReservoirs})$ 
2   reservoirs  $\leftarrow$  new Array[ImageSize]
3   // Generate initial candidates
4   foreach pixel  $q \in \text{Image}$  do
5     reservoirs[ $q$ ]  $\leftarrow \text{RIS}(q)$  // Alg. 3
6   // Evaluate visibility for initial candidates
7   foreach pixel  $q \in \text{Image}$  do
8     if shadowed(reservoirs[ $q$ ]. $y$ ) then
9       reservoirs[ $q$ ]. $W \leftarrow 0$ 
10  // Temporal reuse
11  foreach pixel  $q \in \text{Image}$  do
12     $q' \leftarrow \text{pickTemporalNeighbor}(q)$ 
13    reservoirs[ $q$ ]  $\leftarrow \text{combineReservoirs}(q, \text{reservoirs}[q],$ 
14                                     prevFrameReservoirs[ $q'$ ]) // Alg. 4

```

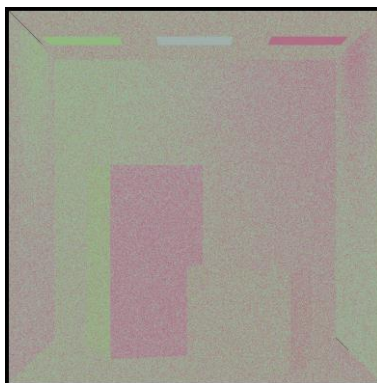
No RIS (uniform)

$$x_i \sim u$$



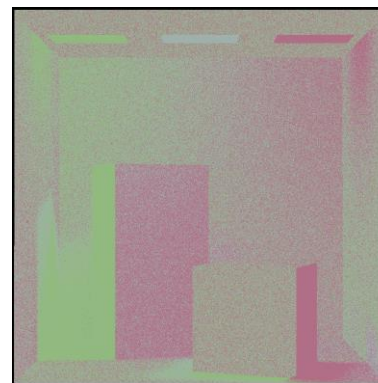
RIS w/o temporal reuse

$$x_i \sim \rho L_e G$$



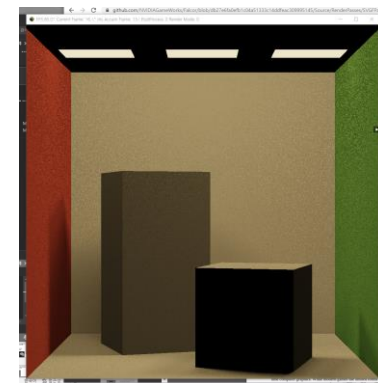
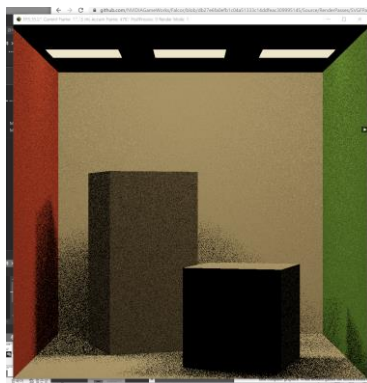
RIS w/ temporal reuse

$$x_i \sim \rho L_e G V$$



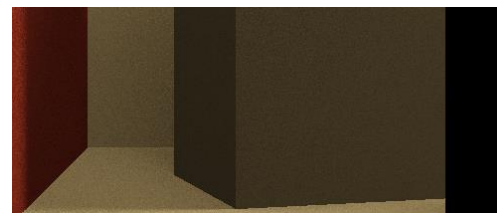
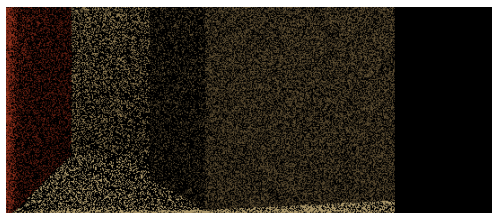
u : uniform pdf
 ρ : BRDF
 L_e : emission
 G : Geometric term
 V : Visibility

Light Weight



1-spp
Rendered Image

(no accumulation)



TODO

- Full ReSTIR implementation. (spatial reuse, unbiased version)
- Incorporation into the existing path tracer pipeline.
 - Considering NEE & MIS & ReSTIR is complex!!