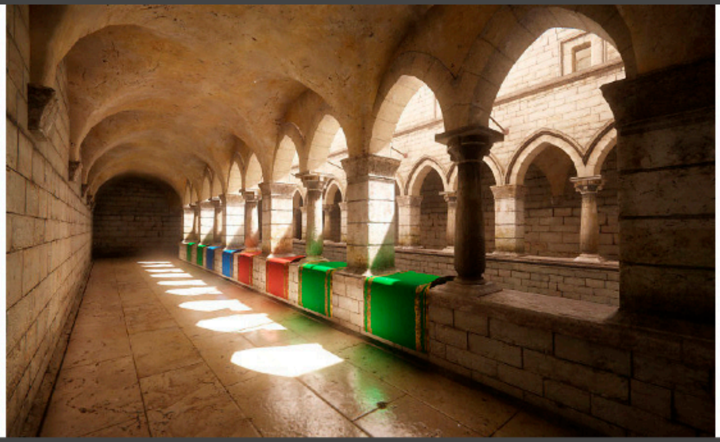
Real-Time High Quality Rendering

GAMES202, Lingqi Yan, UC Santa Barbara

Lecture 12:
Real-Time Ray Tracing 1

Announcements

- GAMES101 resubmission
  - Starting next Monday!
- GAMES202 homework 3
  - Will be released soon hopefully
  - Your understanding is greatly appreciated

Last Lectures

- Real-Time Physically-Based Materials
  - Microfacet BRDF, NDF, shadowing-masking
  - Kulla-Conty Approximation for multiple bounces
  - Disney principled BRDF
- Shading with microfacet BRDFs under polygonal lighting
  - Linearly Transformed Cosines (LTC)
- Non-photorealistic rendering (NPR)

Some Arrangements

- Volumetric / scattering materials will not be covered in this course
  - Too many dependencies (RTE, BSSRDF, single/multiple scattering, etc.)
  - Will be fully covered in offline rendering, together with RTR techniques (delta tracking, dual scattering, layered materials, etc.)

[Final Fantasy VII Remake]

[Black Myth: Wukong]

Some Arrangements

- Unreal Engine 5 early access is available now!
  - Again, both Nanite and Lumen are **TECHNICAL** breakthroughs
  - The underlying science is already understandable after learning this course
  - Will briefly analyze (or rather, guess) possible approaches in the last lecture

[UE5 Early Access Trailer (weakest boss ever)]

Today

- New Topic: Real-Time Ray Tracing (RTRT)
  - Basic idea
  - Motion vector
  - Temporal accumulation / filtering
  - Failure cases
- Filtering techniques and implementation (next lecture)
  - Joint bilateral filtering
  - Spatiotemporal Variance-Guided Filtering (SVGF)

RTRT is the Future

- In the real-time industry, people claim that

> "Ray tracing is the future
> **and ever will be**."
>
> — The real-time industry

RTRT is Happening

- In 2018, NVIDIA announced GeForce **RTX** series (Turing architecture)
  - Opening a $250 billion market

RTRT is Happening

- What does RTX do?

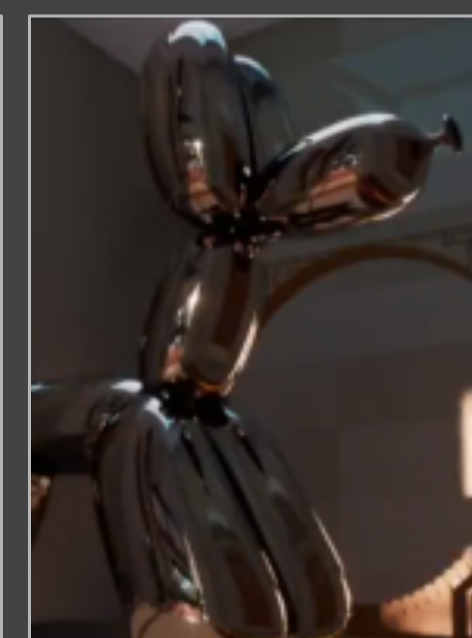
Impressive demos of RTRT

Star Wars Reflections

RTX Demo

Porsche 70 Trailer

SOL

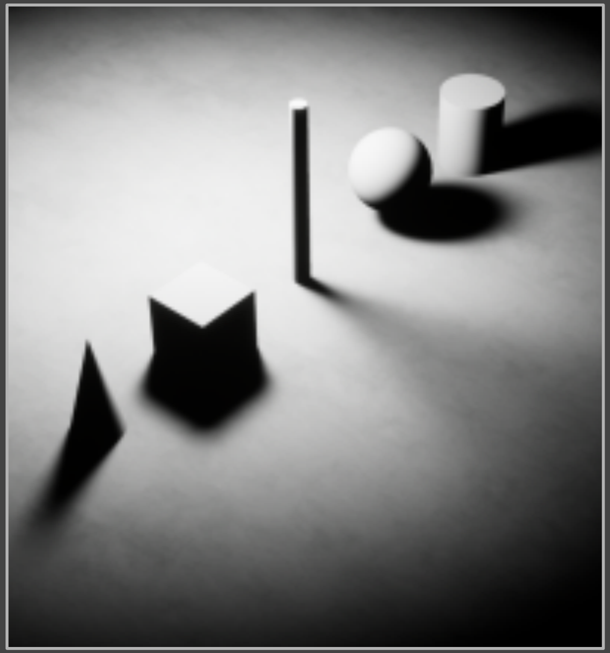


Rosewood Bangkok

RTRT is Happening

- What does RTX actually do?

Advanced **ray traced** effects



Shadows

Reflections & Specular

Ambient Occlusion

Global Illumination

RTRT is Happening

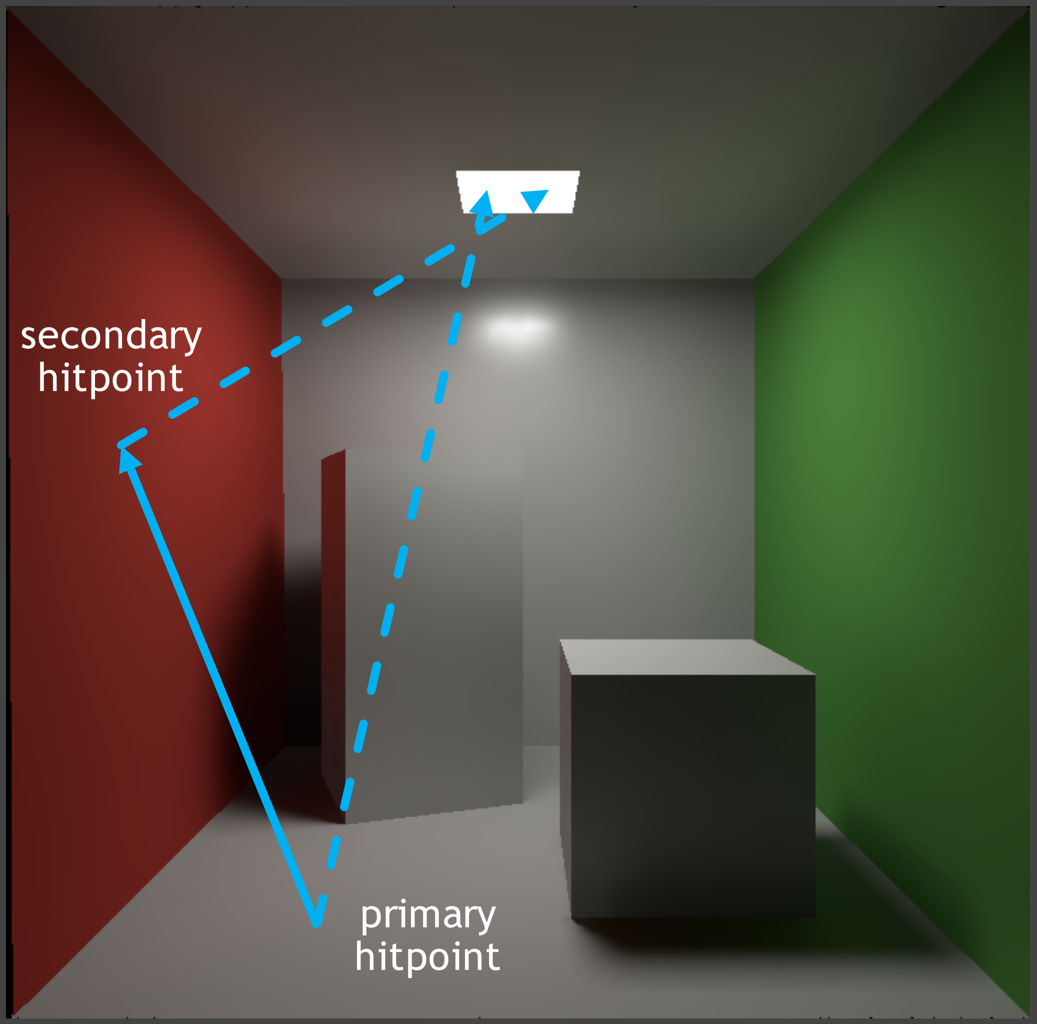
- What does RTX really do?

10 Giga rays per second == **1 sample per pixel**

(for real time applications)

RTRT is Happening

- What does RTX actually do?
- 1 SPP path tracing =
  - 1 rasterization (primary) +
  - 1 ray (primary visibility) +
  - 1 ray (secondary bounce) +
  - 1 ray (secondary vis.)



RTRT is Happening

- 1 SPP = Extremely noisy results
- Key technology
  - **Denoising**

Fun image on Twitter

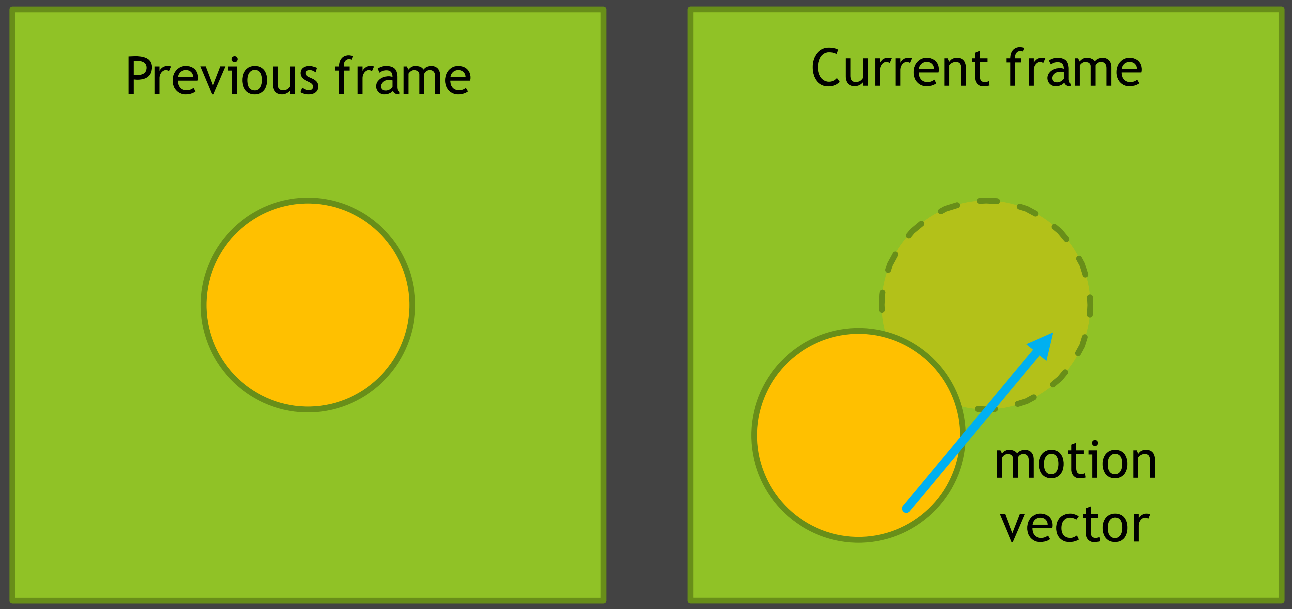
State of the Art* Denoising Solution

Before we proceed…

- Goals (**with 1 SPP**)
  - Quality (no overblur, no artifacts, keep all details…)
  - Speed (< 2 ms to denoise one frame)
- **Mission impossible**
  - Sheared filtering series (SF, AAF, FSF, MAAF, …)
  - Other offline filtering methods (IPP, BM3D, APR, …)
  - Deep learning series (CNN, Autoencoder, …)

Industrial Solution

- 3 most important ideas
  - Temporal!
  - **Temporal!!**
  - **Temporal!!!**

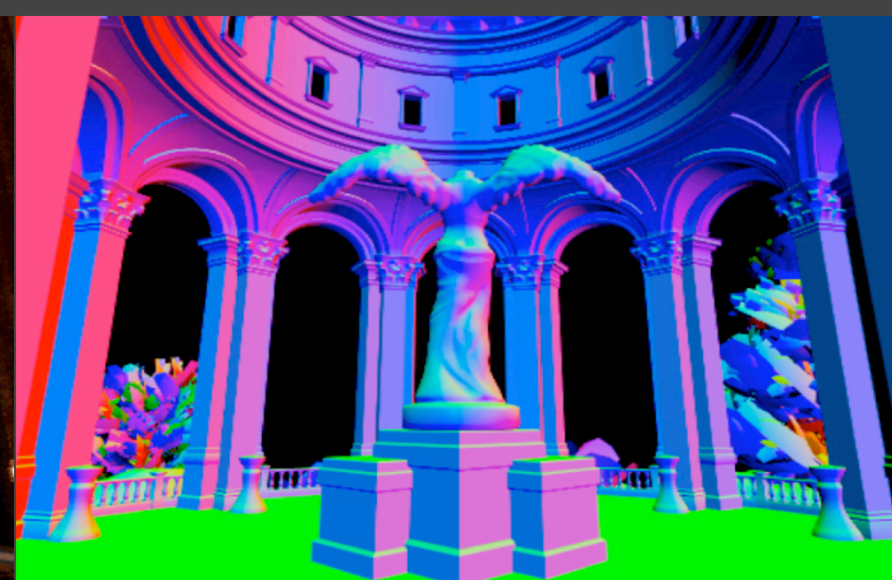


- Key idea
  - Suppose the previous frame is denoised and reuse it
  - Use **motion vectors** to find previous locations
  - Essentially increased SPP
  - Spatial?

The G-Buffers

- Geometry buffer
  - The auxiliary information acquired **FOR FREE*** during rendering
  - Usually, per pixel depth, normal, world coordinate, etc.
  - Therefore, only **screen space** info

Direct illumination

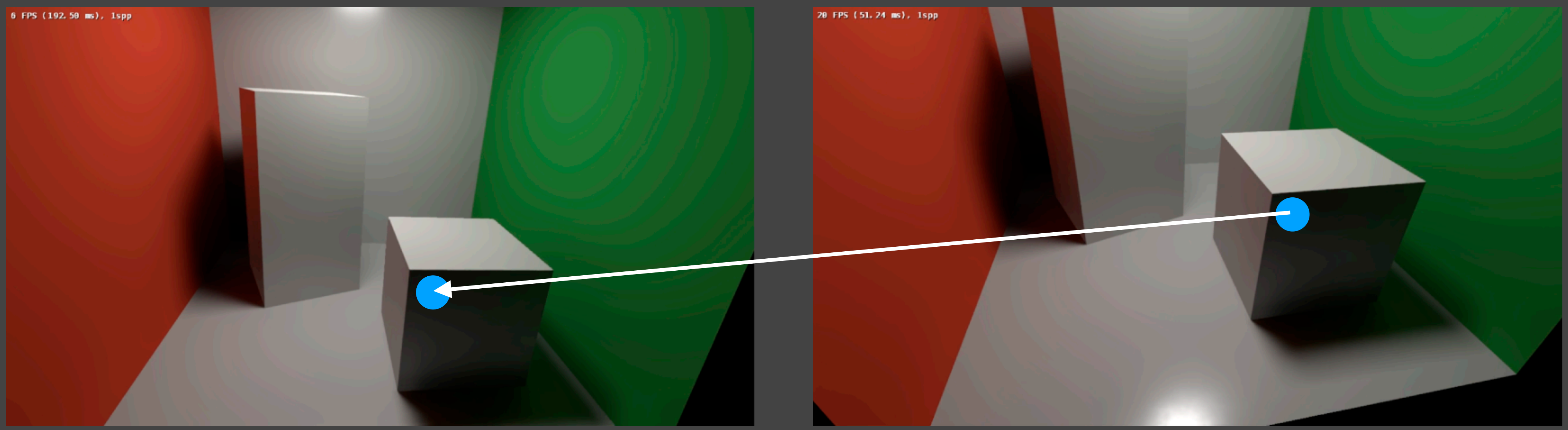


Normal

Albedo

Back Projection

- Pixel x in the current frame i
  - ~~Where was **it** in the last frame $i-1$?~~
  - What pixel in frame $i-1$ contains **the same place/point that you see though pixel x in frame i**?



frame i-1 frame i

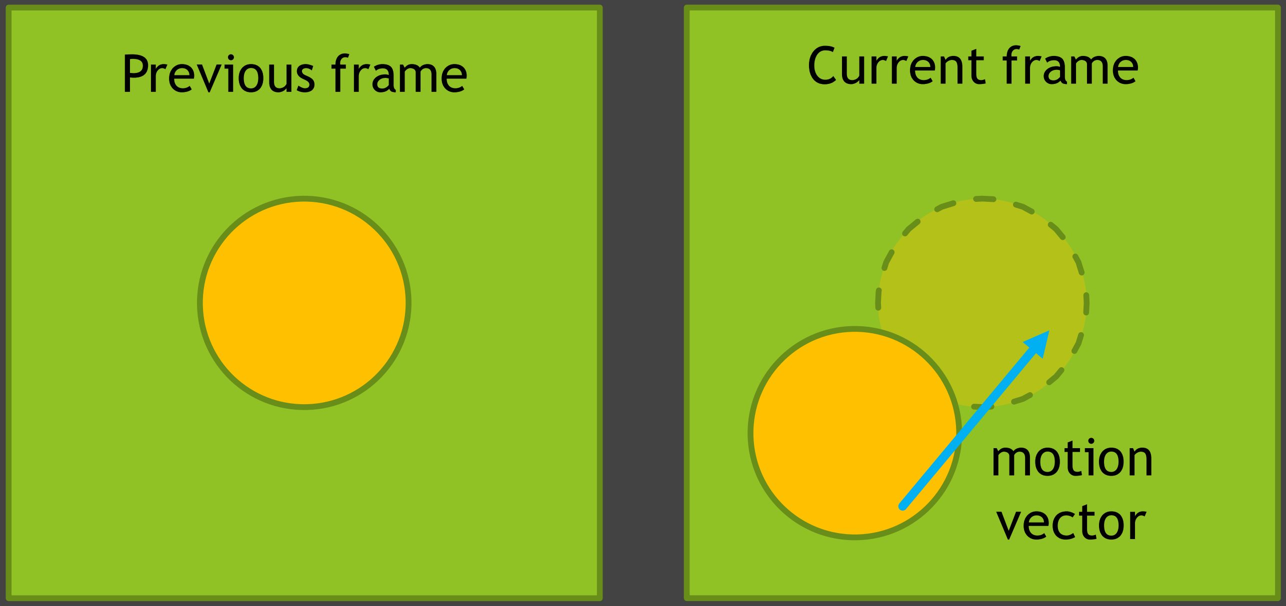
Back Projection

- Pixel x in the current frame i
  - Where was it in the last frame $i-1$?

- Back projection
  - If world coord s is available as a G-buffer, just take it
  - Otherwise, $s = M^{-1}V^{-1}P^{-1}E^{-1}x$ (still require z value)
  - Motion is known: $s' \xrightarrow{T} s$, thus $s' = T^{-1}s$
  - Project world coord in frame $i-1$ to its screen:
    $x' = P'V'M's'$

Temporal Accum./Denoising

- Let's denote:
  - ~ : unfiltered
  - - : filtered



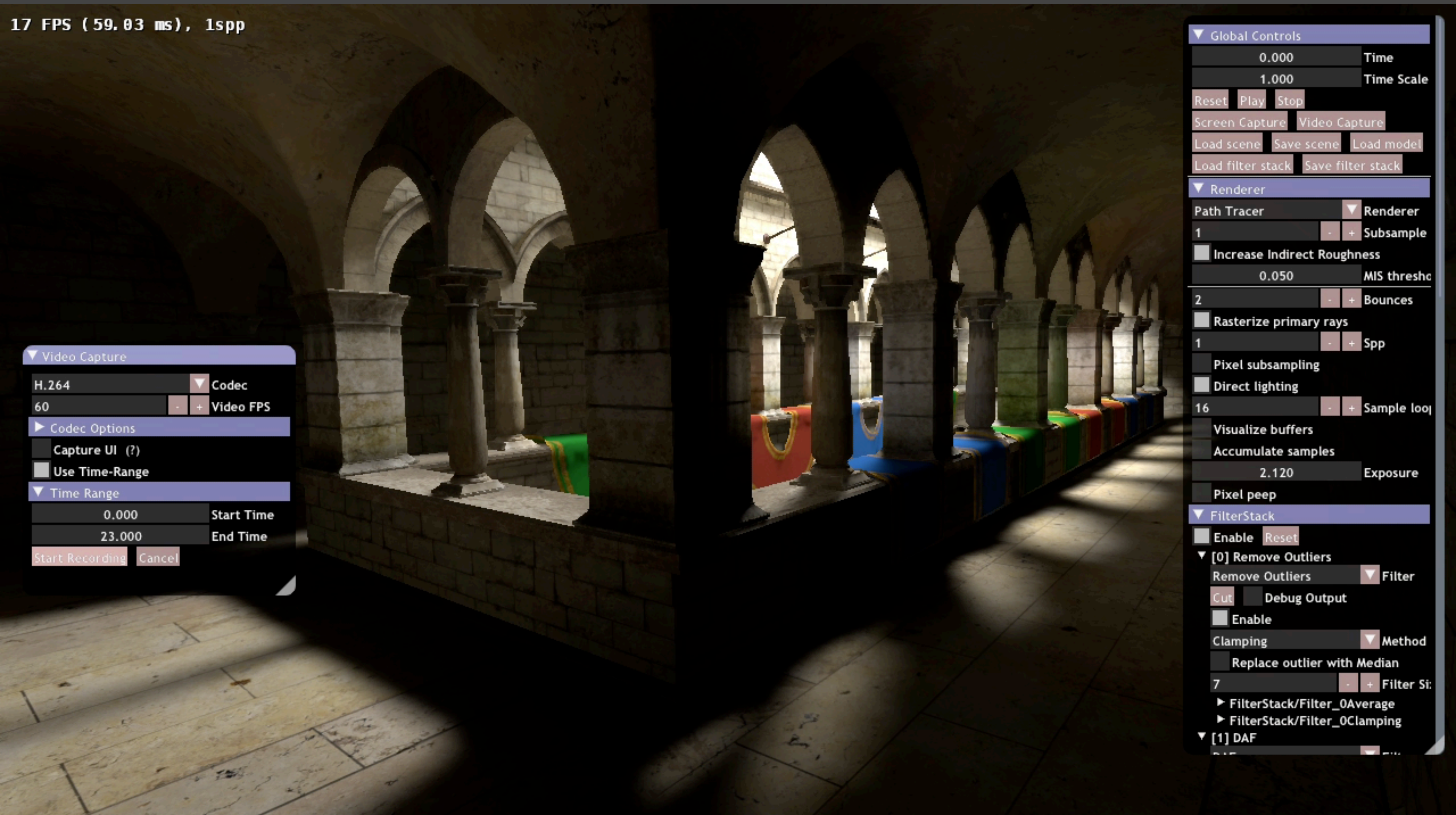
- This frame (i-th frame)

$$\bar{C}^{(i)} = Filter[\tilde{C}^{(i)}]$$

$$\bar{C}^{(i)} = \alpha \bar{C}^{(i)} + (1 - \alpha) C^{(i-1)}$$

80%-90% contributions
from last frame(s)!

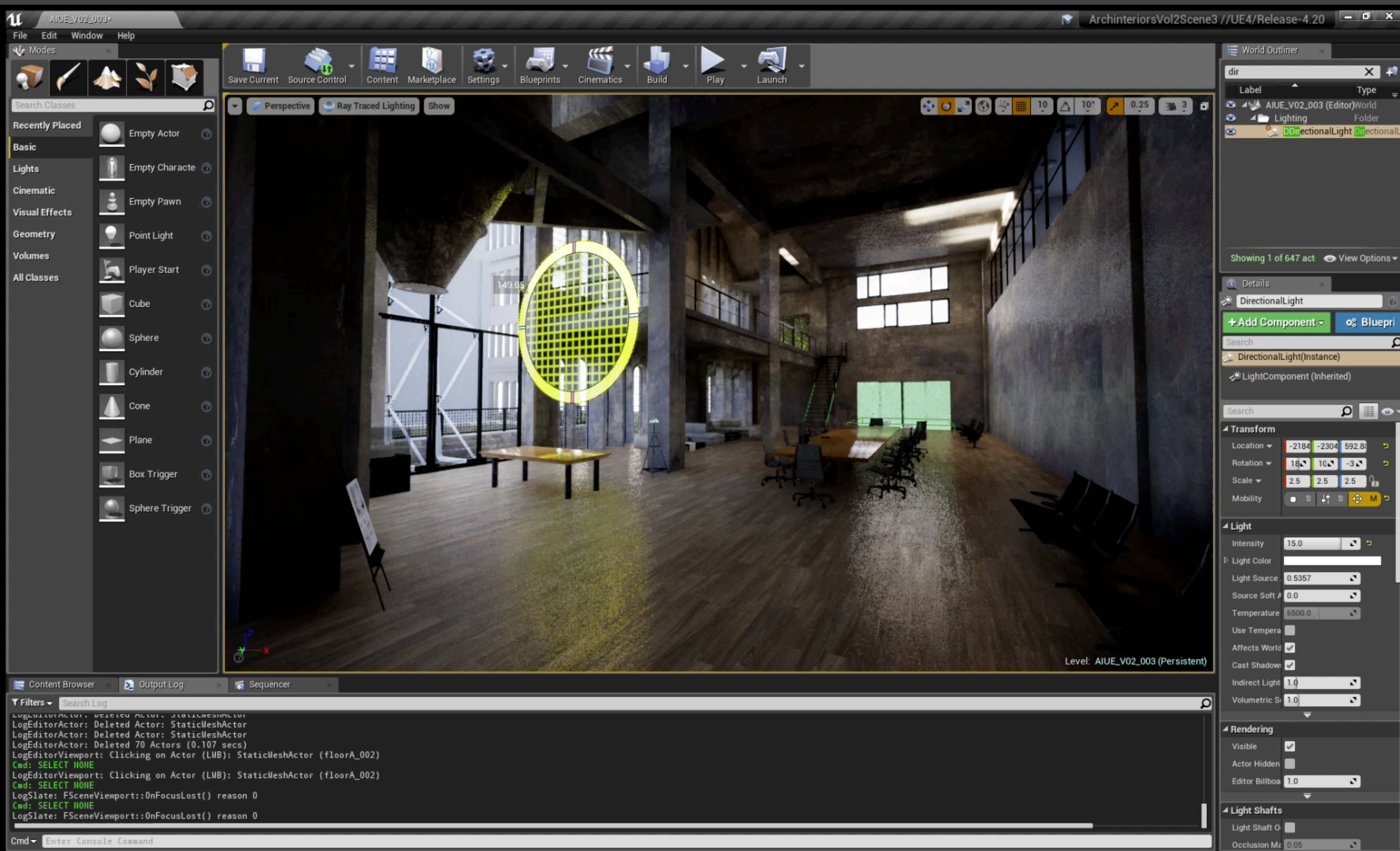
$$\alpha = 0.1 - 0.2$$


17 FPS (59.03 ms), 1spp
Video Capture
H.264
Codec
60
Video FPS
Codec Options
Capture UI (?)
Use Time-Range
Time Range
0.000
Start Time
23.000
End Time
Start Recording
Cancel
Global Controls
0.000
Time
1.000
Time Scale
Reset
Play
Stop
Screen Capture
Video Capture
Load scene
Save scene
Load model
Load filter stack
Save filter stack
Renderer
Path Tracer
Renderer
1
Subsample
Increase Indirect Roughness
0.050
MIS thresho
2
Bounces
Rasterize primary rays
1
Spp
Pixel subsampling
Direct lighting
16
Sample loo
Visualize buffers
Accumulate samples
2.120
Exposure
Pixel peep
FilterStack
Enable
Reset
[0] Remove Outliers
Remove Outliers
Filter
Cut
Debug Output
Enable
Clamping
Method
Replace outlier with Median
7
Filter Si
FilterStack/Filter_0Average
FilterStack/Filter_0Clamping
[1] DAF

1spp Ray Traced Global Illumination

1spp Ray Traced Global Illumination + Denoising

Ground Truth


AIUE_V02_003*
ArchinteriorsVol2Scene3 //UE4/Release-4.20
File Edit Window Help
Modes
Search Classes
Recently Placed
Basic
Lights
Cinematic
Visual Effects
Geometry
Volumes
All Classes
Empty Actor
Empty Characte
Empty Pawn
Point Light
Player Start
Cube
Sphere
Cylinder
Cone
Plane
Box Trigger
Sphere Trigger
Save Current
Source Control
Content
Marketplace
Settings
Blueprints
Cinematics
Build
Play
Launch
Perspective
Ray Traced Lighting
Show
Level: AIUE_V02_003 (Persistent)
World Outliner
dir
Label
Type
AIUE_V02_003 (Editor) World
Lighting Folder
DirectionalLight DirectionalL
Showing 1 of 647 act
View Options
Details
DirectionalLight
+Add Component
Bluepri
DirectionalLight(Instance)
LightComponent (Inherited)
Transform
Location -2184 -2304 592.8
Rotation
Scale 2.5 2.5 2.5
Mobility
Light
Intensity 15.0
Light Color
Light Source 0.5357
Source Soft 0.0
Temperature 6500.0
Use Tempera
Affects World
Cast Shadow
Indirect Light 1.0
Volumetric S 1.0
Rendering
Visible
Actor Hidden
Editor Billboa 1.0
Light Shafts
Light Shaft O
Occlusion Ma 0.05
Content Browser
Output Log
Sequencer
Filters
Search Log
LogEditorActor: Deleted Actor: StaticMeshActor
LogEditorActor: Deleted Actor: StaticMeshActor
LogEditorActor: Deleted 70 Actors (0.107 secs)
LogEditorViewport: Clicking on Actor (LMB): StaticMeshActor (floorA_002)
Cmd: SELECT NONE
LogEditorViewport: Clicking on Actor (LMB): StaticMeshActor (floorA_002)
Cmd: SELECT NONE
LogSlate: FSceneViewport::OnFocusLost() reason 0
Cmd: SELECT NONE
LogSlate: FSceneViewport::OnFocusLost() reason 0
Cmd
Enter Console Command

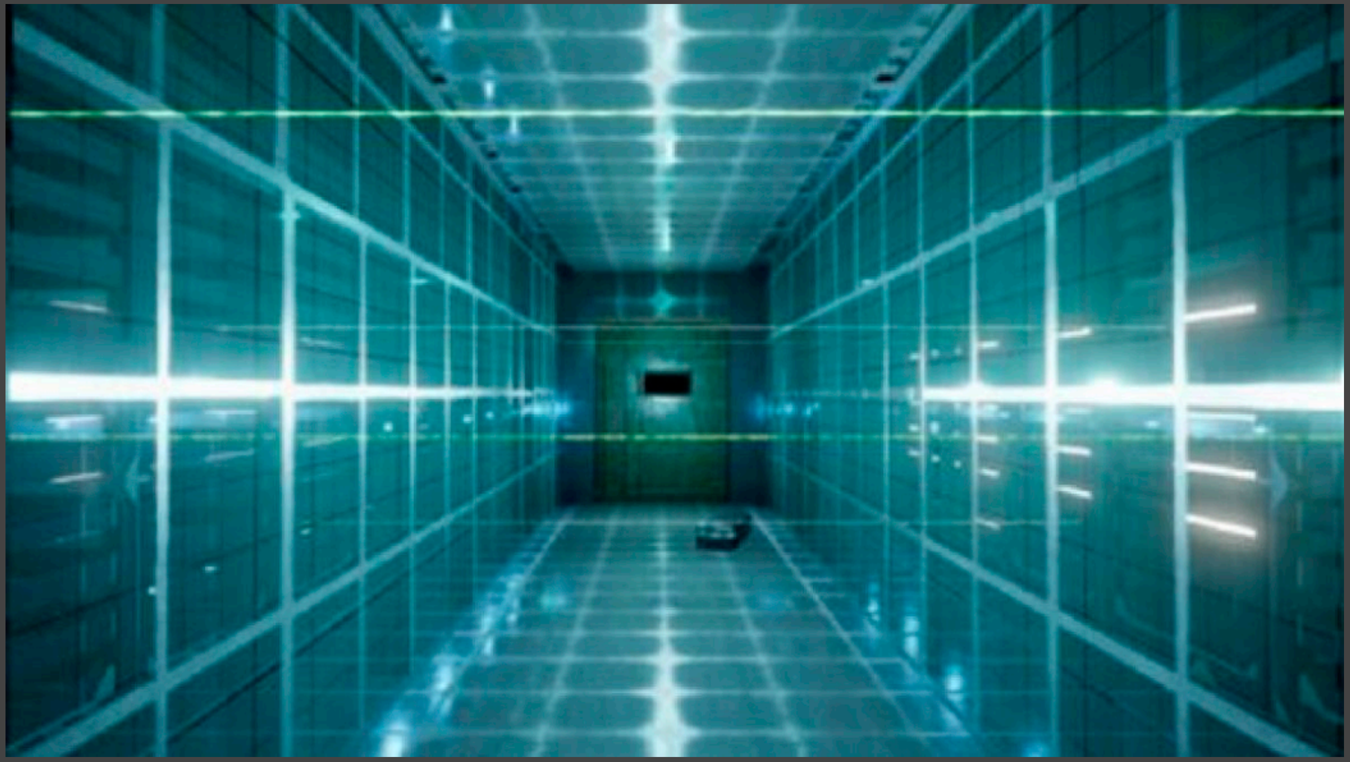
Temporal Failure

- Temporal info is not always available
  - Failure case 1: switching scenes
    (burn-in period)

[Monster Hunter Rise]

Temporal Failure

- Temporal info is not always available
  - Failure case 1: switching scenes
  - Failure case 2: walking backwards in a hallway **(screen space issue)**



[Resident Evil Movie]

Temporal Failure

- Temporal info is not always available
  - Failure case 1: switching scenes
  - Failure case 2: walking backwards in a hallway
  - Failure case 3: suddenly appearing background **(disocclusion)**

Ignoring Temporal Failure?

- We can still blindly use temporal information
  - Of course, this is incorrect
  - But what kind of artifact will it bring?
- **Lagging!**


10×
Traditional motion vectors (no clamp)
Shadow
Fence
Moving Objects
Shadow
Pink Room
Moving Light
Changing Light Sizes
Shadow
Apples
Curved Surfaces
Shadow
Fence
Multiple Lights
Glossy
Sun Temple
Moving Camera
Glossy
Restaurant
Moving Camera
Glossy
Restaurant
Curved Surfaces
Occlusion
PICA
Moving Objects
Occlusion
PICA 2
Moving Objects

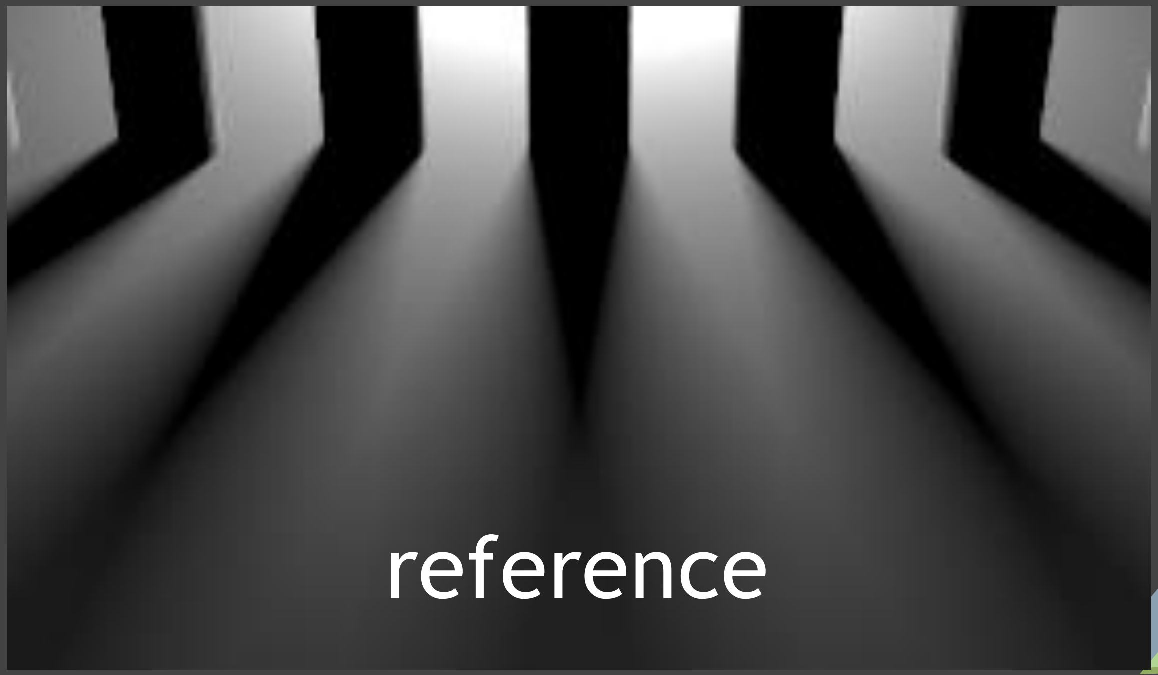
Adjustments to Temp. Failure

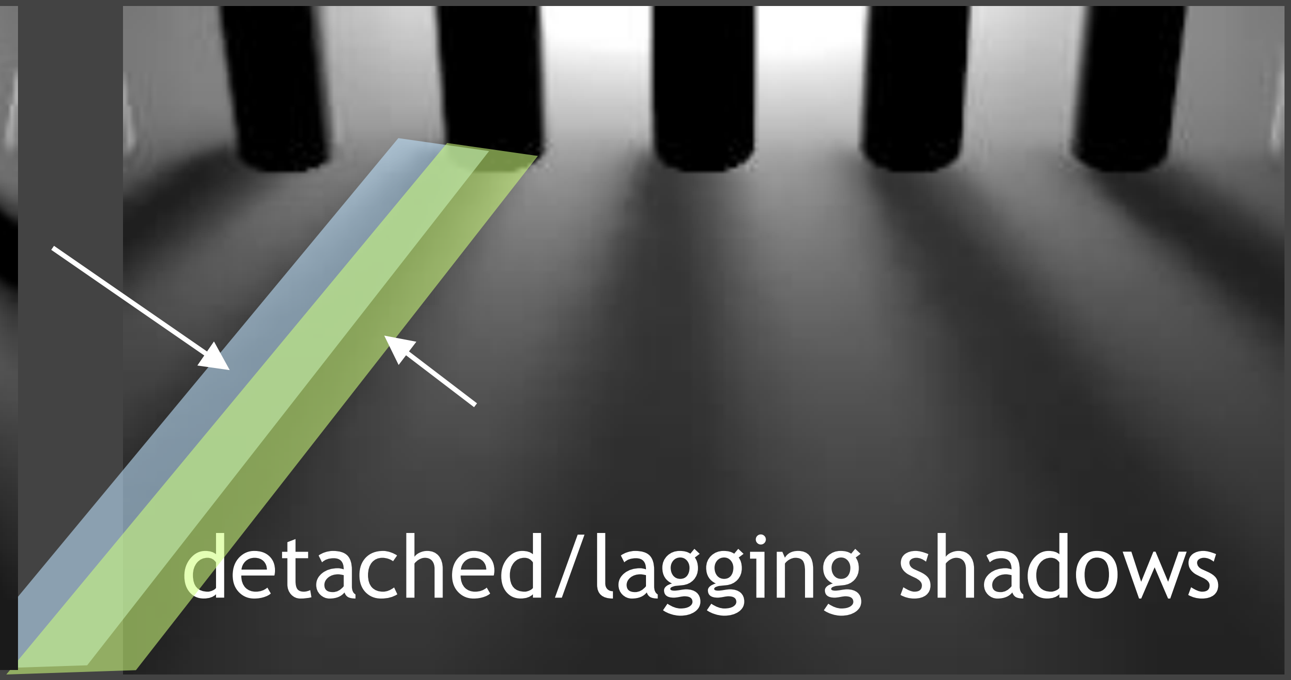
- Clamping $$\bar{C}^{(i)} = \alpha \bar{C}^{(i)} + (1 - \alpha) C^{(i-1)}$$
  - Clamp previous toward current
- Detection
  - Use e.g. object ID to detect temporal failure
  - Tune α, binary or continuously
  - Possibly strengthen / enlarge spatial filtering
- Problem: **re-introducing noise**!

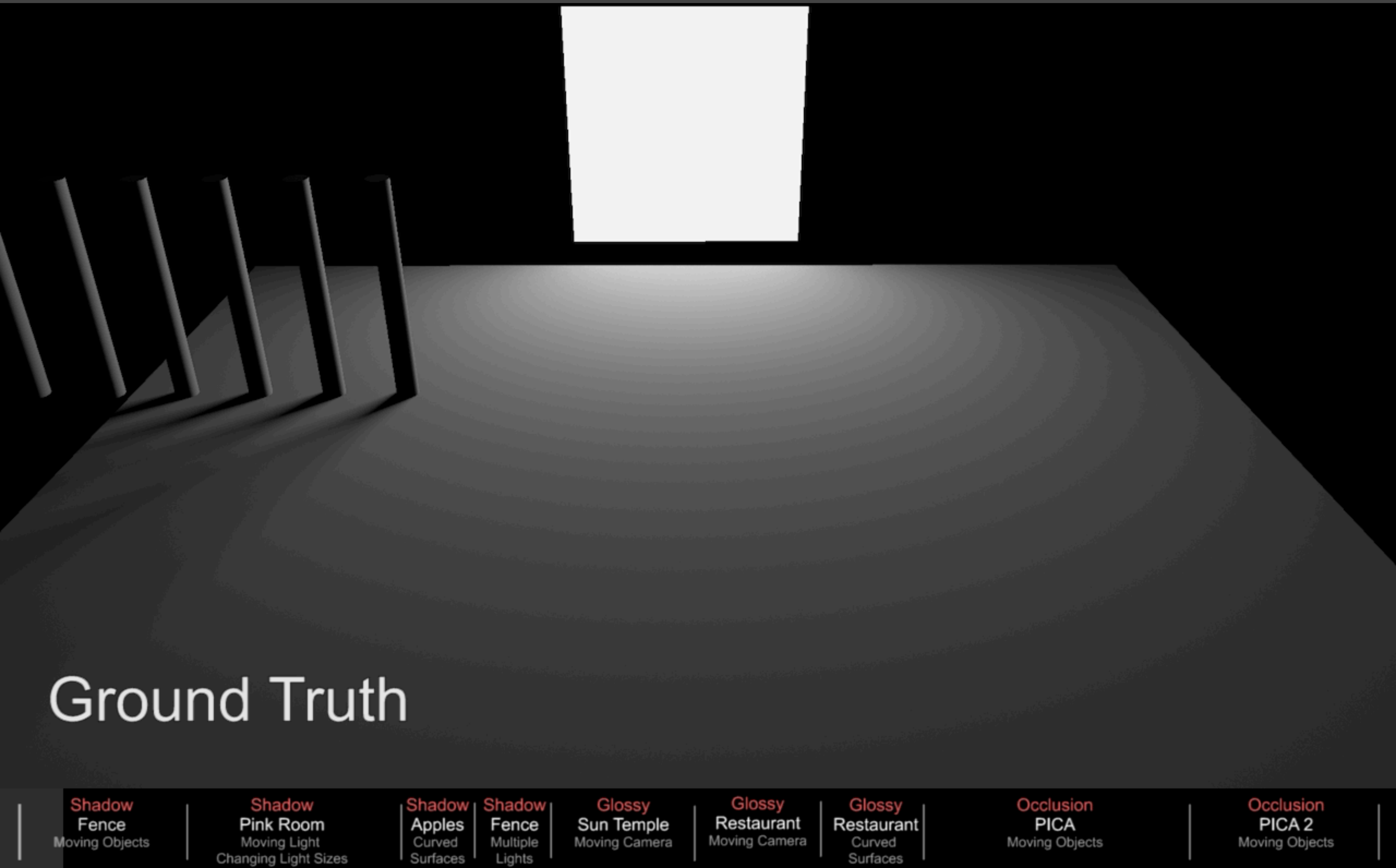

10×
Traditional motion vectors (clamp)
Shadow
Fence
Moving Objects
Shadow
Pink Room
Moving Light
Changing Light Sizes
Shadow
Apples
Curved
Surfaces
Shadow
Fence
Multiple
Lights
Glossy
Sun Temple
Moving Camera
Glossy
Restaurant
Moving Camera
Glossy
Restaurant
Curved
Surfaces
Occlusion
PICA
Moving Objects
Occlusion
PICA 2
Moving Objects

More Temporal Failure

- Temporal failure can also happen in shading
  - Consider the "fence" scene with a moving light behind
  - What's the motion vector of the **shadows**?

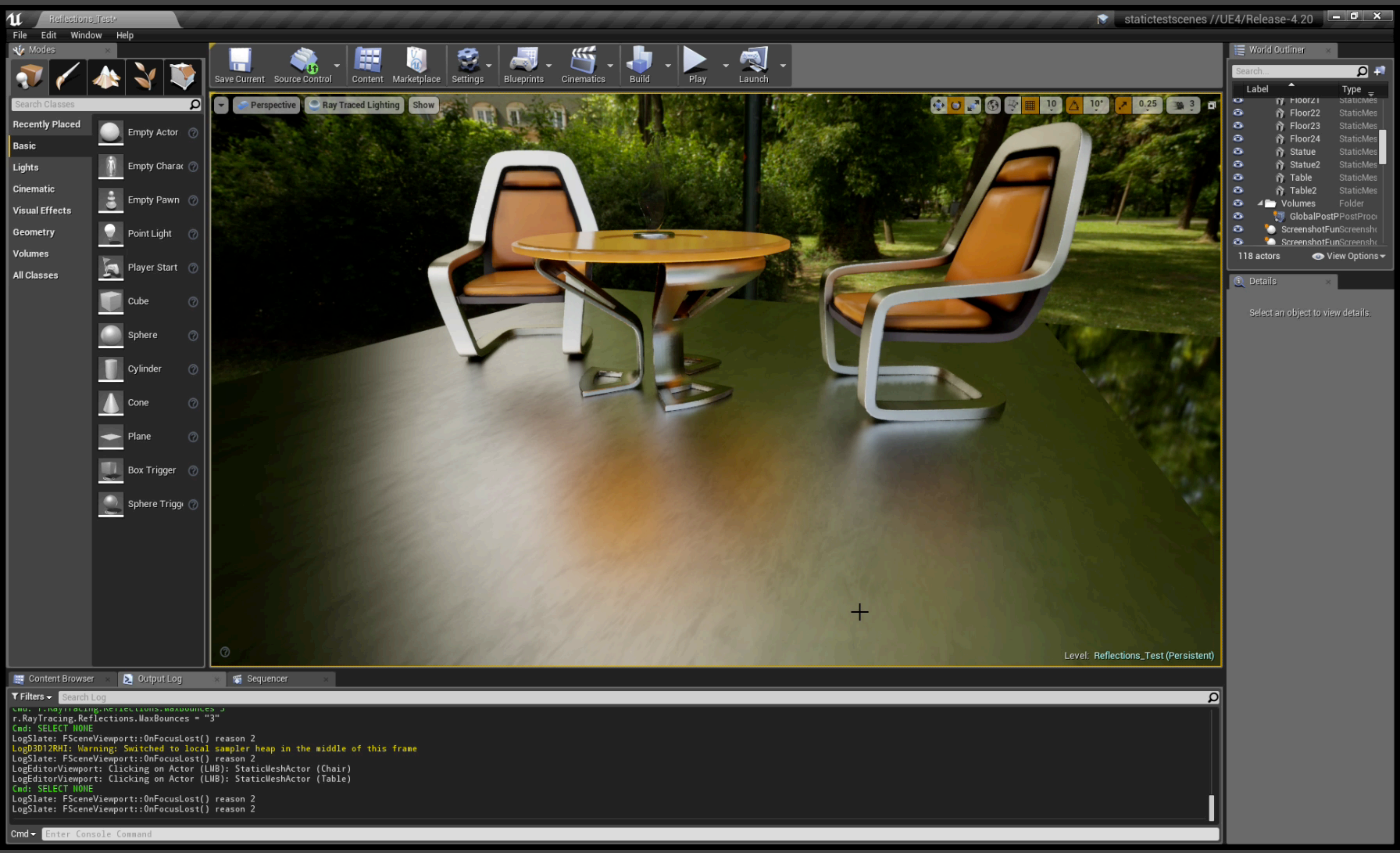





Ground Truth
Shadow
Fence
Moving Objects
Shadow
Pink Room
Moving Light
Changing Light Sizes
Shadow
Apples
Curved
Surfaces
Shadow
Fence
Multiple
Lights
Glossy
Sun Temple
Moving Camera
Glossy
Restaurant
Moving Camera
Glossy
Restaurant
Curved
Surfaces
Occlusion
PICA
Moving Objects
Occlusion
PICA 2
Moving Objects

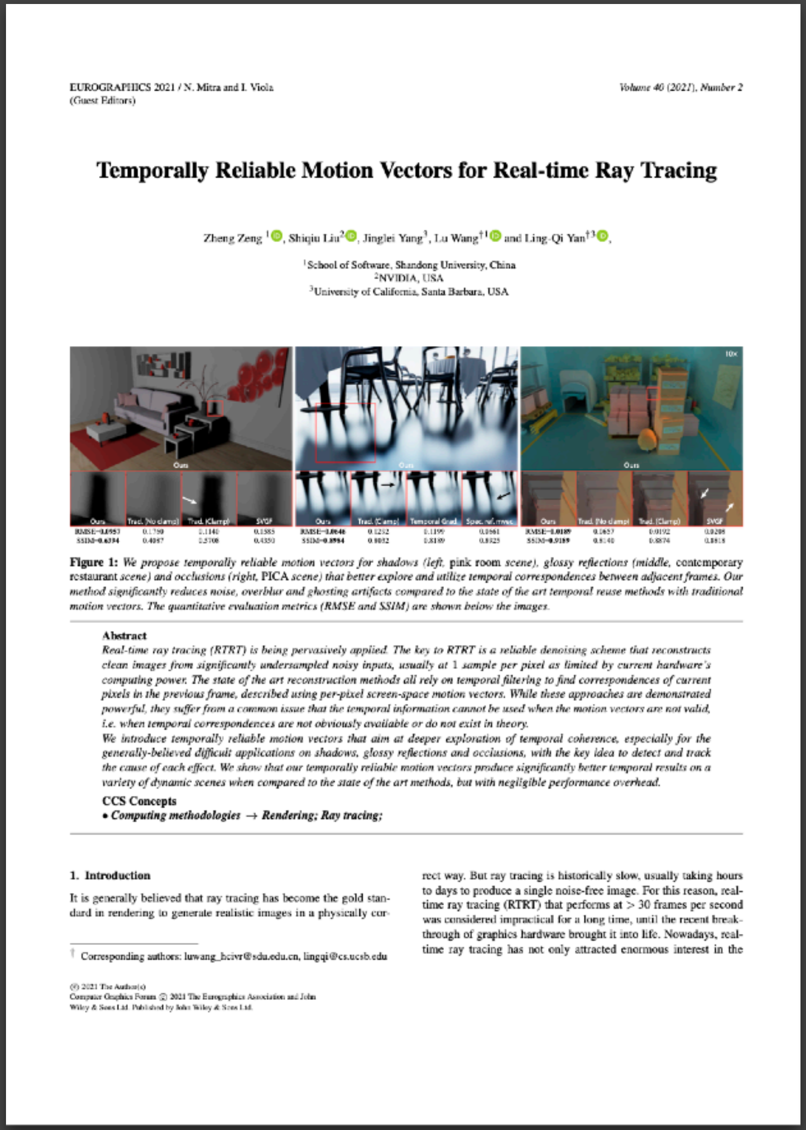
More Temporal Failure

- Temporal failure can also happen in shading
  - Consider the moving chairs
  - What's the motion vector of the **glossy reflected images**?


Reflections_Test*
staticttestscenes //UE4/Release-4.20
File Edit Window Help
Modes
Save Current
Source Control
Content
Marketplace
Settings
Blueprints
Cinematics
Build
Play
Launch
World Outliner
Search Classes
Recently Placed
Basic
Lights
Cinematic
Visual Effects
Geometry
Volumes
All Classes
Empty Actor
Empty Pawn
Point Light
Player Start
Cube
Sphere
Cylinder
Cone
Plane
Box Trigger
Perspective
Ray Traced Lighting
Show
Label
Type
Floor22
Floor23
Floor24
Statue
Statue2
Table
Table2
Volumes
Folder
118 actors
View Options
Details
Select an object to view details.
Level: Reflections_Test (Persistent)
Content Browser
Output Log
Sequencer
Filters
Search Log
r.RayTracing.Reflections.MaxBounces = "3"
Cmd: SELECT NONE
LogSlate: FSceneViewport::OnFocusLost() reason 2
LogD3D12RHI: Warning: Switched to local sampler heap in the middle of this frame
LogSlate: FSceneViewport::OnFocusLost() reason 2
LogEditorViewport: Clicking on Actor (LMB): StaticMeshActor (Chair)
LogEditorViewport: Clicking on Actor (LMB): StaticMeshActor (Table)
Cmd: SELECT NONE
LogSlate: FSceneViewport::OnFocusLost() reason 2
LogSlate: FSceneViewport::OnFocusLost() reason 2
Cmd
Enter Console Command

Some Side Notes

- The temporal accumulation is inspired by Temporal Anti-Aliasing (TAA)
  - They are very similar
  - Temporal reuse essentially increases the sampling rate
- Is there any research on further alleviating temporal failure?
  - Yes! Our Eurographics (EG) paper “Temporally Reliable Motion Vectors for Real-time Ray Tracing”



EUROGRAPHICS 2021 / N. Mitra and I. Viola
(Guest Editors)

Volume 40 (2021), Number 2

Temporally Reliable Motion Vectors for Real-time Ray Tracing

Zheng Zeng [1], Shiqiu Liu[2], Jinglei Yang[3], Lu Wang[†1] and Ling-Qi Yan[†3],

[1]School of Software, Shandong University, China
[2]NVIDIA, USA
[3]University of California, Santa Barbara, USA

Figure 1: *We propose temporally reliable motion vectors for shadows (left, pink room scene), glossy reflections (middle, contemporary restaurant scene) and occlusions (right, PICA scene) that better explore and utilize temporal correspondences between adjacent frames. Our method significantly reduces noise, overblur and ghosting artifacts compared to the state of the art temporal reuse methods with traditional motion vectors. The quantitative evaluation metrics (RMSE and SSIM) are shown below the images.*

Abstract
Real-time ray tracing (RTRT) is being pervasively applied. The key to RTRT is a reliable denoising scheme that reconstructs clean images from significantly undersampled noisy inputs, usually at 1 sample per pixel as limited by current hardware's computing power. The state of the art reconstruction methods all rely on temporal filtering to find correspondences of current pixels in the previous frame, described using per-pixel screen-space motion vectors. While these approaches are demonstrated powerful, they suffer from a common issue that the temporal information cannot be used when the motion vectors are not valid, i.e. when temporal correspondences are not obviously available or do not exist in theory.
We introduce temporally reliable motion vectors that aim at deeper exploration of temporal coherence, especially for the generally-believed difficult applications on shadows, glossy reflections and occlusions, with the key idea to detect and track the cause of each effect. We show that our temporally reliable motion vectors produce significantly better temporal results on a variety of dynamic scenes when compared to the state of the art methods, but with negligible performance overhead.

CCS Concepts
• ***Computing methodologies*** → ***Rendering; Ray tracing;***

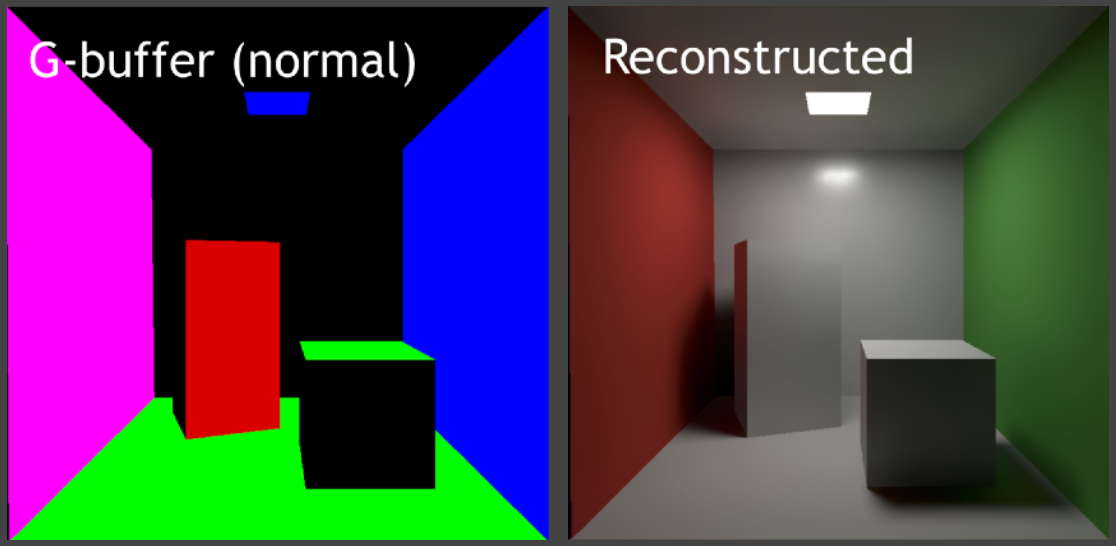
1. Introduction

It is generally believed that ray tracing has become the gold standard in rendering to generate realistic images in a physically correct way. But ray tracing is historically slow, usually taking hours to days to produce a single noise-free image. For this reason, real-time ray tracing (RTRT) that performs at > 30 frames per second was considered impractical for a long time, until the recent breakthrough of graphics hardware brought it into life. Nowadays, real-time ray tracing has not only attracted enormous interest in the

[†] Corresponding authors: luwang_hcivr@sdu.edu.cn, lingqi@cs.ucsb.edu

(c) 2021 The Author(s)
Computer Graphics Forum © 2021 The Eurographics Association and John Wiley & Sons Ltd. Published by John Wiley & Sons Ltd.

Spatial Denoising (Next Lec.)



- This frame (i-th frame)

$$\bar{C}^{(i)} = Filter[\tilde{C}^{(i)}]$$

- How to filter the current frame?
  - Bilateral filter? (https://en.wikipedia.org/wiki/Bilateral_filter)
  - Cross / joint bilateral filter (and their variants)
    - Taking more info into account
    - G-buffers: normal / depth / object ID, etc.

Next Lecture

- Real-Time Ray Tracing 2
  (filtering techniques and implementation)

[Spatiotemporal Variance-Guided Filtering]

Thank you!