Real-Time High Quality Rendering

GAMES202, Lingqi Yan, UC Santa Barbara

Lecture 14:
A Glimpse of Industrial Solutions




Announcements

- GAMES101 resubmission has started
  - http://smartchair.org/GAMES101-Spring2021
- GAMES202 homework 4 & 5 will be released soon
- Course certification with my signature
  - Will be sent out in electronic version after all the resubmissions
  - Sign up for “Certification Request” (like a homework)

- Today: the last lecture of GAMES202!

Last Lectures

- Real-Time Ray Tracing (RTRT)
  - Basic idea
  - Temporal
    - Motion vector
    - Temporal accumulation / filtering
    - Temporal failures
  - Spatial
    - Implementing a spatial filter
    - Joint bilateral filtering
    - Outlier removal

Today

- Finishing up: specific filtering solutions for RTRT
  - Spatiotemporal Variance-Guided Filtering (SVGF)
  - Recurrent AutoEncoder (RAE)
- Practical Industrial solutions
  - Anti-aliasing
  - Super sampling and DLSS
  - Cascaded / multi-resolution solutions
  - /tiled/deferred shading，particles，engines

Specific Filtering Approaches for RTRT

SVGF — Basic Idea

- Spatiotemporal Variance-Guided Filtering [Schied et al.]
  - Very similar to the basic spatio-temporal denoising scheme
  - But with some additional variance analysis and tricks

[Spatiotemporal Variance-Guided Filtering]

SVGF — Joint Bilateral Filtering

- 3 factors to guide filtering
  - **Depth**

$$w_z = \exp\left(-\frac{|z(p) - z(q)|}{\sigma_z |\nabla z(p) \cdot (p - q)| + \epsilon}\right)$$

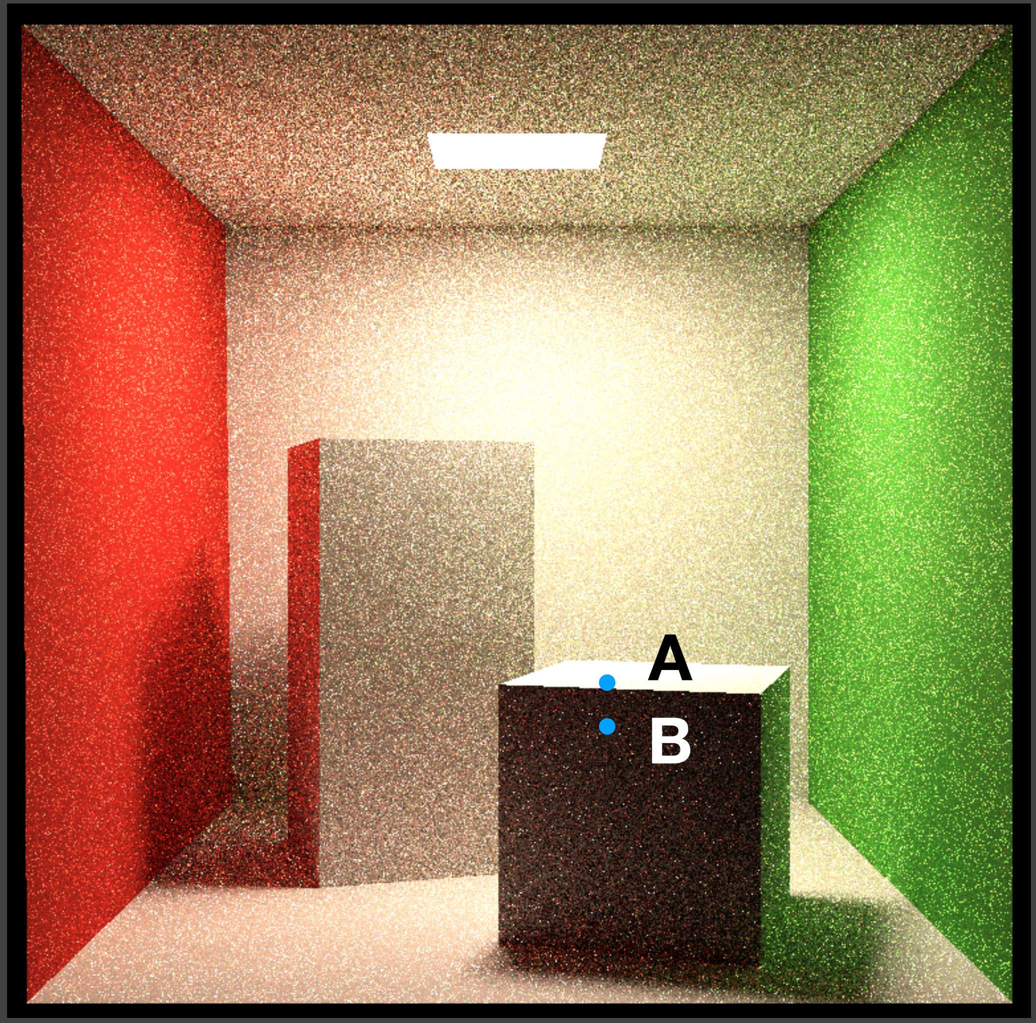
  - Understanding:
    - A and B are on the same plane, of similar color, so they should contribute to each other
    - But the depth between A and B are very different!
    - Therefore, it is preferred to use the depth difference **w.r.t. the tangent plane**

SVGF — Joint Bilateral Filtering

- 3 factors to guide filtering
  - **Normal**

    $$w_n = \max(0, n(p) \cdot n(q))^{\sigma_n}$$

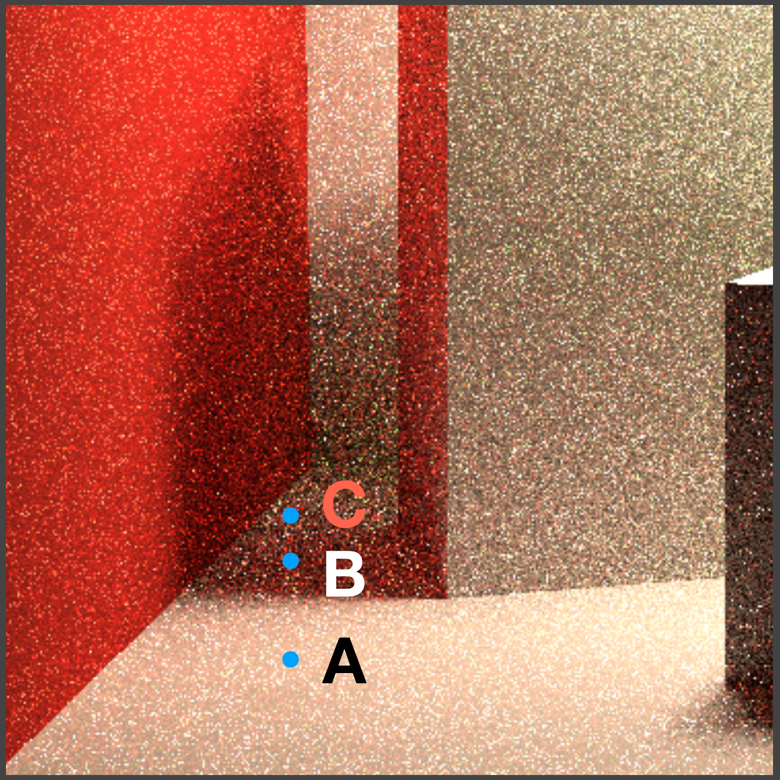
  - Recall, does not have to be a Gaussian
  - Note: in case normal maps exist, use macro normals



SVGF — Joint Bilateral Filtering

- 3 factors to guide filtering
  - **Luminance** (grayscale color value)

$$w_l = \exp\left(-\frac{|l_i(p) - l_i(q)|}{\sigma_l \sqrt{g_{3\times3}(\mathrm{Var}(l_i(p)))} + \epsilon}\right)$$



- Variance
  - Calculate spatially in 7x7
  - Also averaged over time using motion vectors
  - Take another 3x3 spatial filter before use

SVGF — Results

SVGF — Results

SVGF — Failure Cases

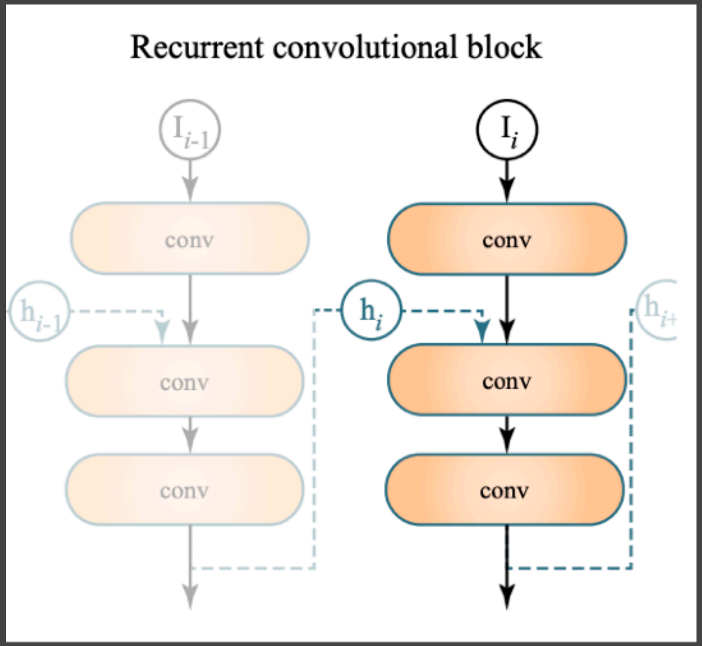
RAE — Basic Idea

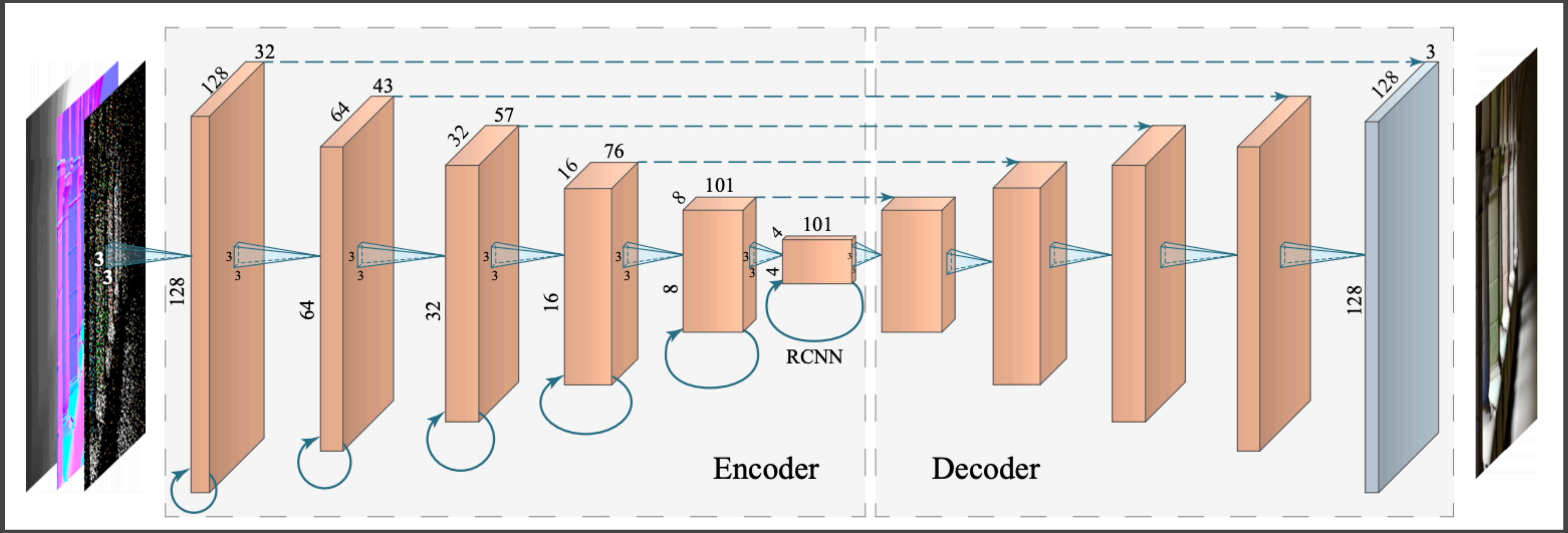
- Interactive Reconstruction of Monte Carlo Image Sequences using a Recurrent denoising AutoEncoder [Chaitanya et al.]
  - A post-processing network that does denoising (noisy -> clean)
  - With the help of G-buffers
  - The network automatically performs temporal accumulation

- Key architecture design
  - AutoEncoder (or U-Net) structure
  - Recurrent convolutional block

RAE — Architecture

- AutoEncoder
  - Skip / residual connections for faster and better training
- Recurrent block
  - Accumulates (and gradually forgets) information from previous frames





RAE — Results


Learning-based filter
Recurrent autoencoder
1 sample/pixel input

RAE — Results

Comparison

	Quality	Artifact	Performance	Explanability	Where did the paper go
SVGF	**Clean**	Ghosting	**Fast**	**Yes**	HPG
RAE (when first invented)	Overblur	Ghosting	Slow	No	**SIGGRAPH**

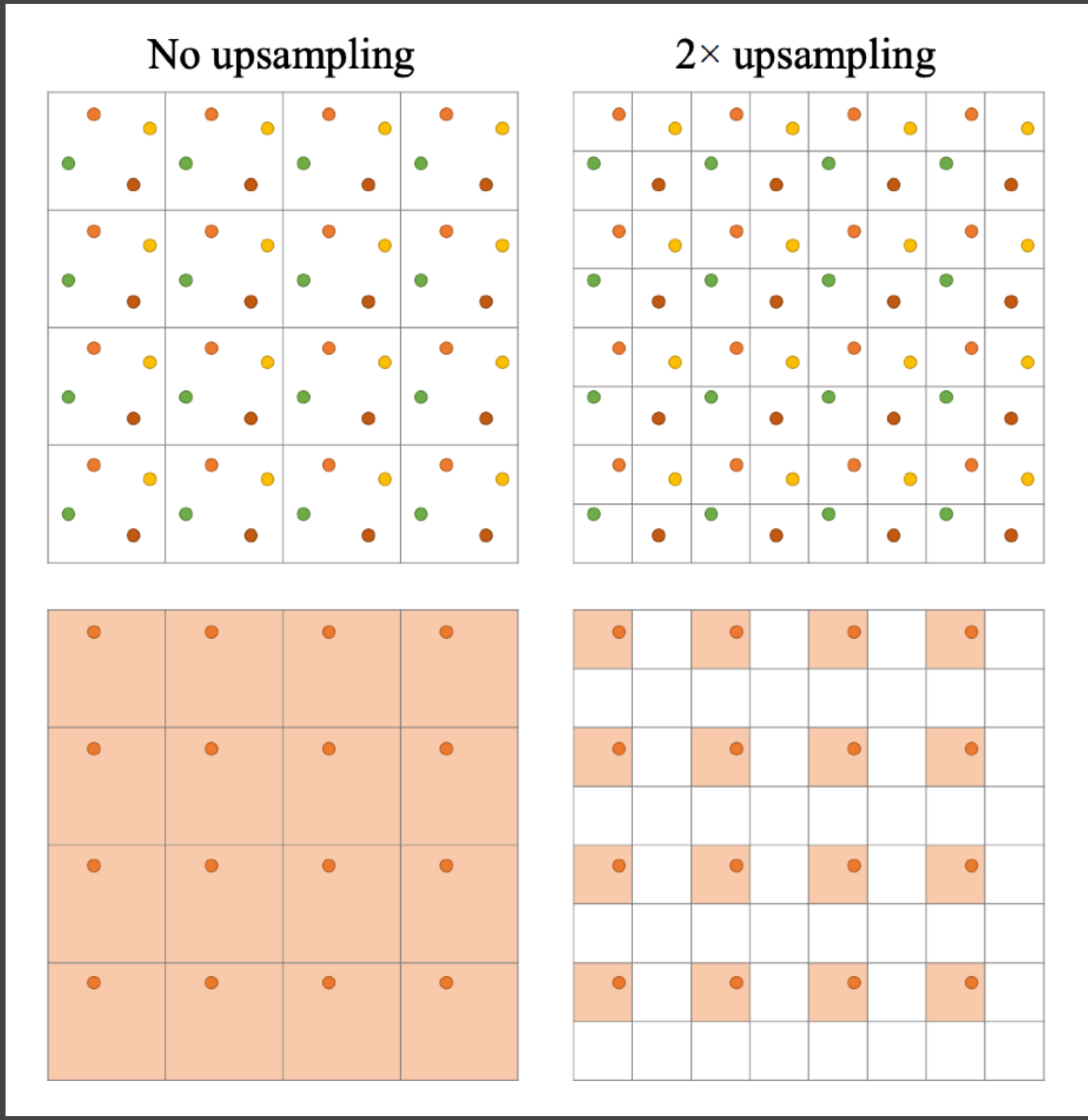
Questions?

Practical Industrial solutions

(Still, from the scientific perspective)

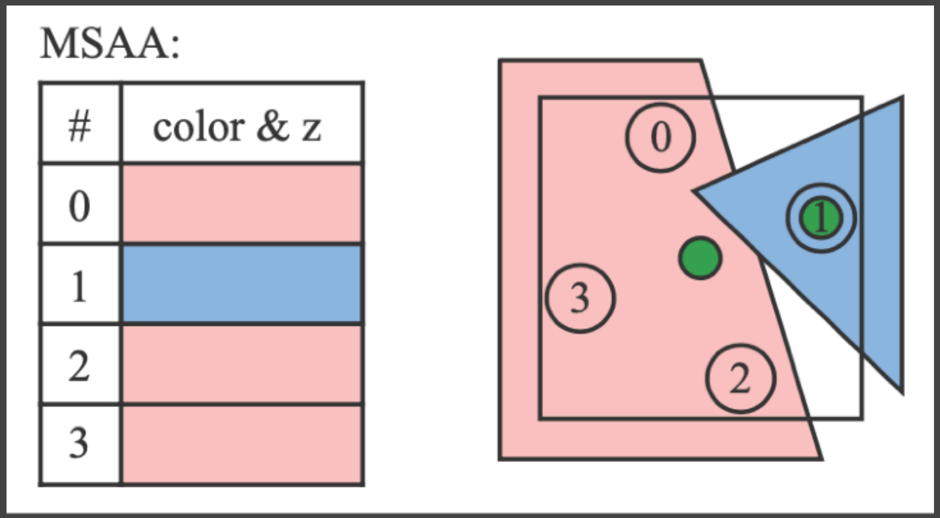
Temporal Anti-Aliasing (TAA)

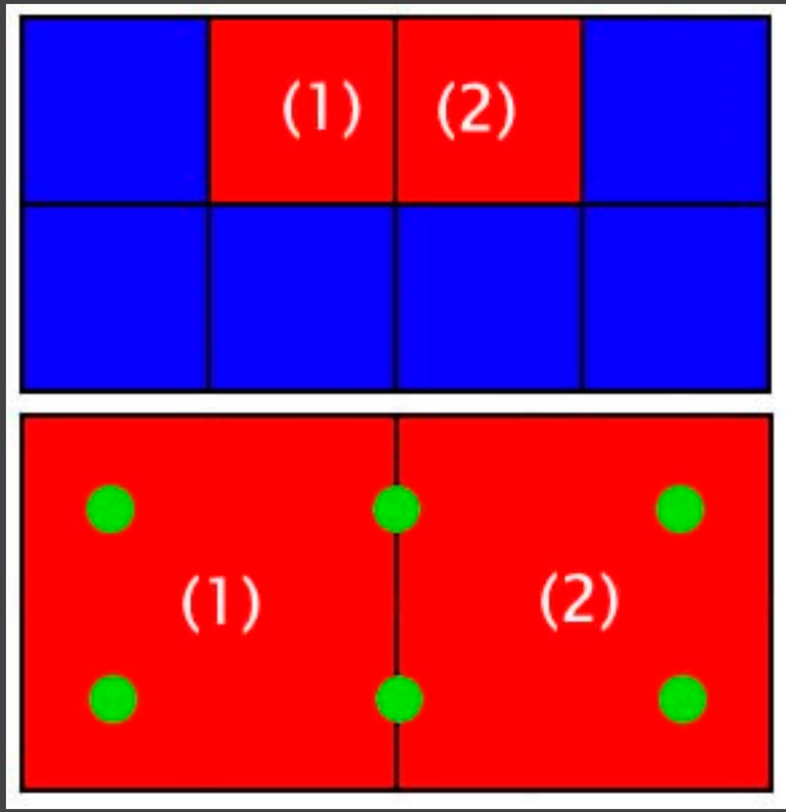
- Recall: why aliasing?
  - Not enough samples per pixel during rasterization
  - Therefore, the ultimate solution is to use more samples
- Temporal Anti-Aliasing
  - Distributing / reuse samples across frames (time)
  - Almost exactly the same as in RTRT



Notes on Anti-Aliasing

- Additional note 1
  - MSAA (Multisample) vs SSAA (Supersampling)
- SSAA is straightforward
  - Rendering at a larger resolution, then downsample
  - The ultimate solution, but costly
- MSAA: an improvement on performance
  - The same primitive is shaded only once
  - Reuse samples across pixels

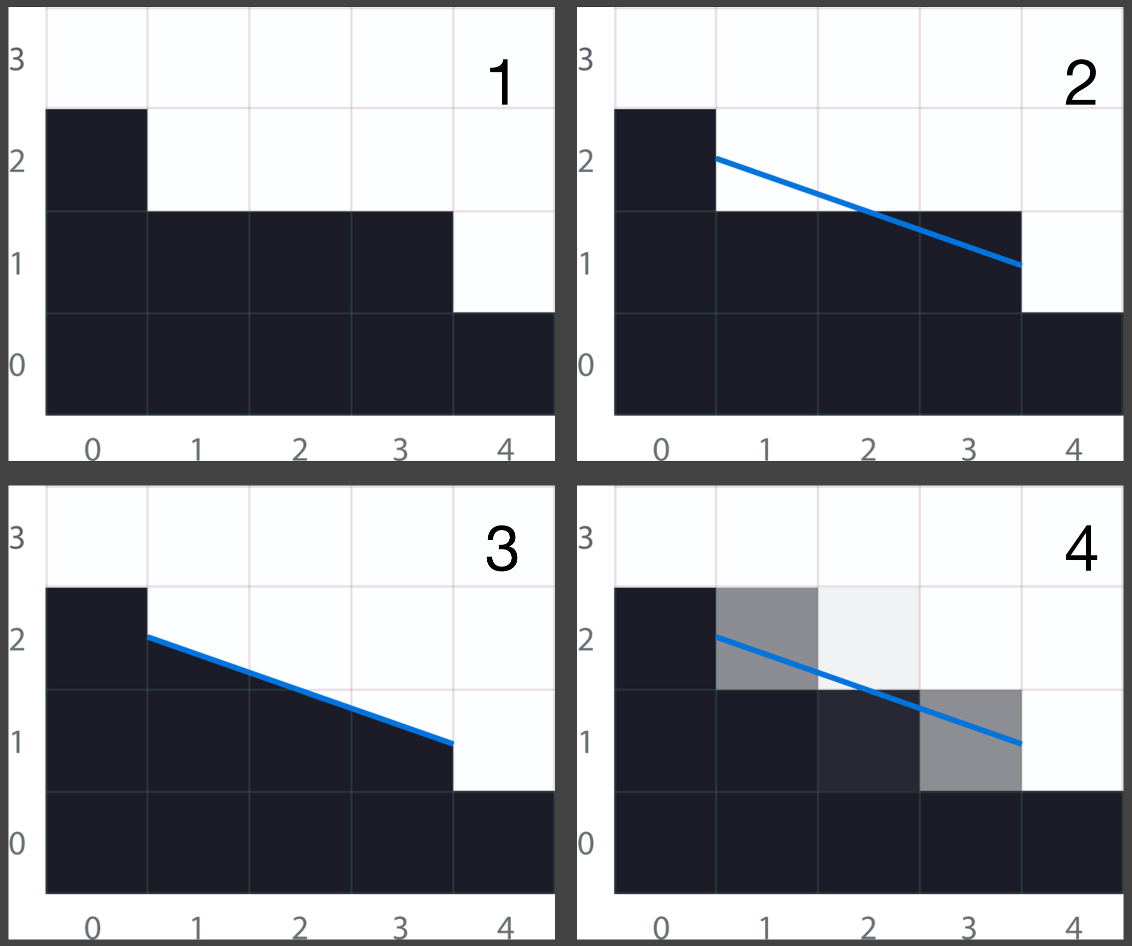




https://www.sapphirenation.net/anti-aliasing-comparison-performance-quality

Notes on Anti-Aliasing

- Additional note 2
  - State of the art image based anti-aliasing solution
  - SMAA (Enhanced subpixel morphological AA)
  - History: FXAA -> MLAA (Morphological AA) -> SMAA
- Additional note 3
  - G-buffers should never be anti-aliased!



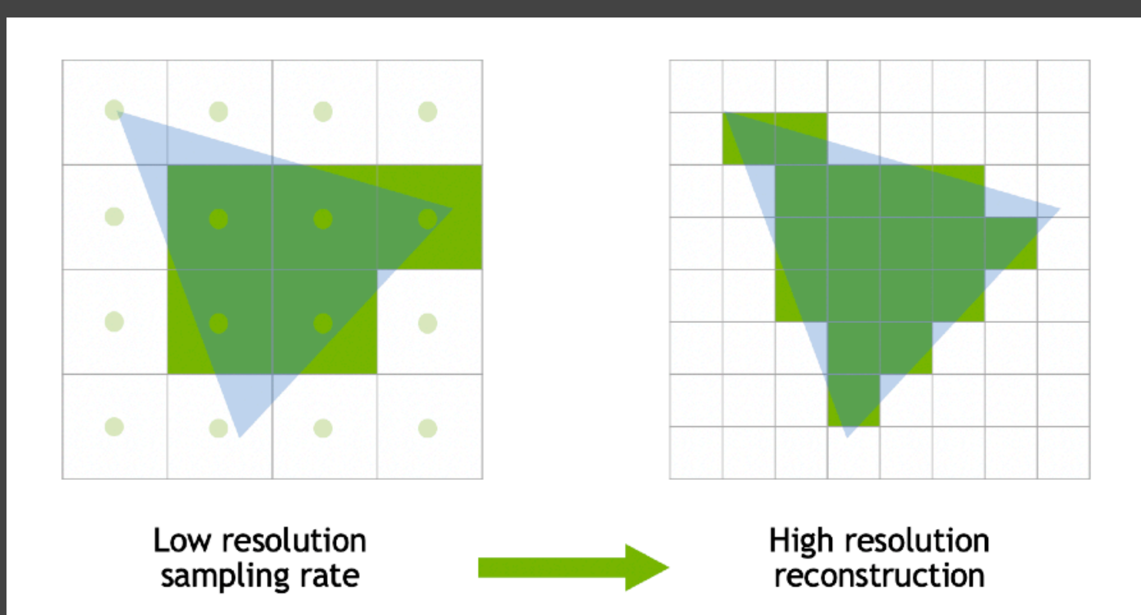
http://www.iryoku.com/smaa/

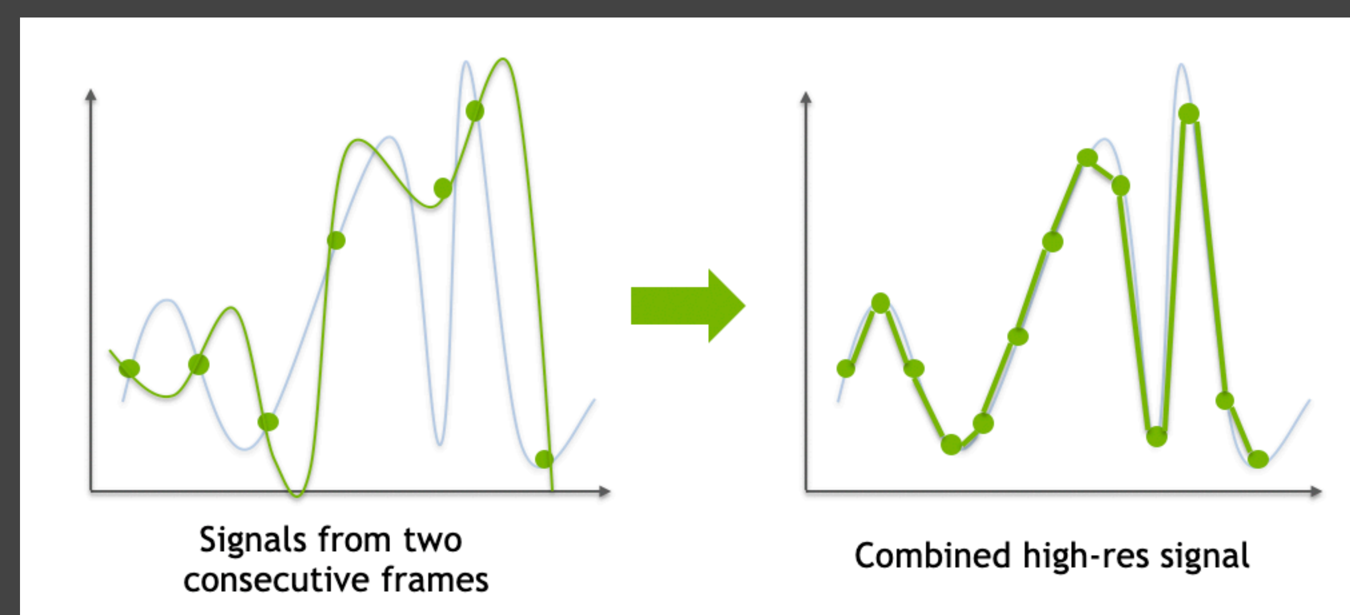
Temporal Super Resolution

- Super resolution (or super sampling)
  - Literal understanding: increasing resolution
  - Source 1 (DLSS 1.0): out of nowhere / completely guessed
  - Source 2 (DLSS 2.0): from temporal information
- Key idea of Deep Learning Super Sampling (DLSS) 2.0
  - Yet another TAA-like application
  - Temporally reuse samples to increase resolution

DLSS 2.0

- Main problem
  - Upon temporal failure, clamping is no longer an option
  - Because we need a clear value for each smaller pixel
  - Therefore, key is how to use temporal info smarter than clamping





DLSS 2.0

DLSS 2.0

DLSS 2.0

DLSS 2.0

- An importance practical issue
  - If DLSS itself runs at 30ms per frame, it's dead already
  - Network inference performance optimization (classified)
- Counterpart of DLSS
  - By AMD: FidelityFX Super Resolution
  - By Facebook: Neural Supersampling for Real-time Rendering [Xiao et al.]
- Any future work?
  - Also classified
  - But wish me good luck in SIGGRAPH Asia 2021