AGGREGATE G-BUFFER ANTI-ALIASING

Cyril Crassin[1], Morgan McGuire[1,2], Kayvon Fatahalian[3], Aaron Lefohn[1]

[1]NVIDIA [2]Williams College [3]Carnegie Mellon University

i3D 2015

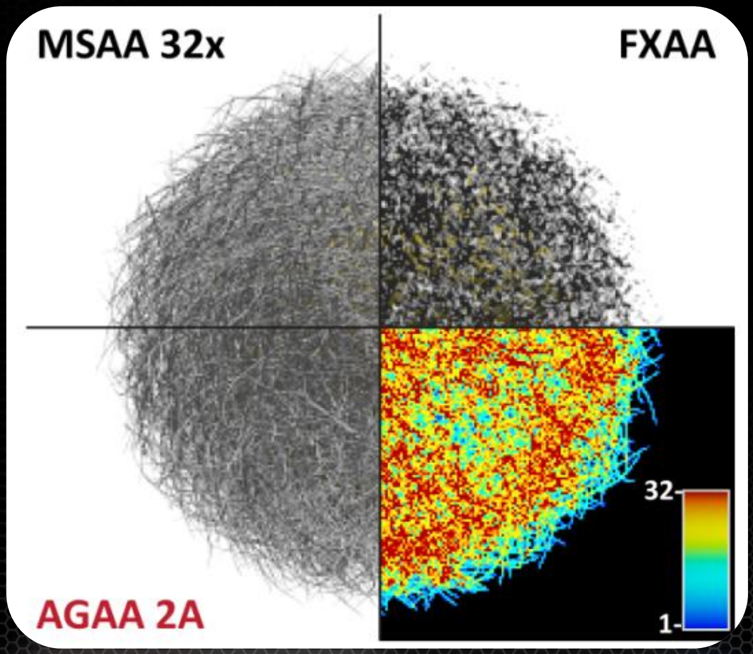
NVIDIA

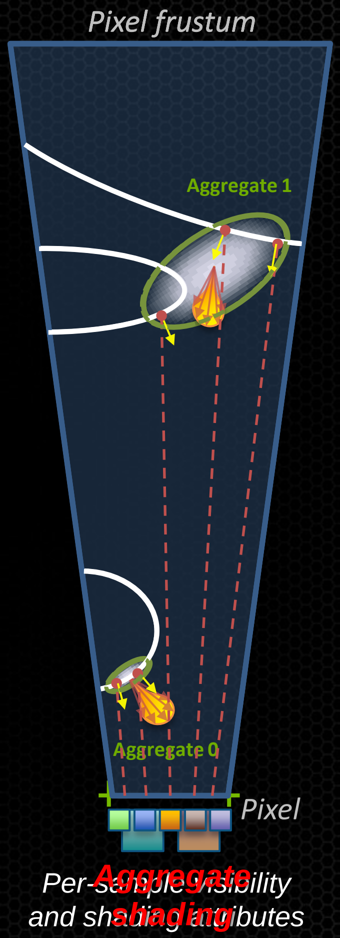
Motivation

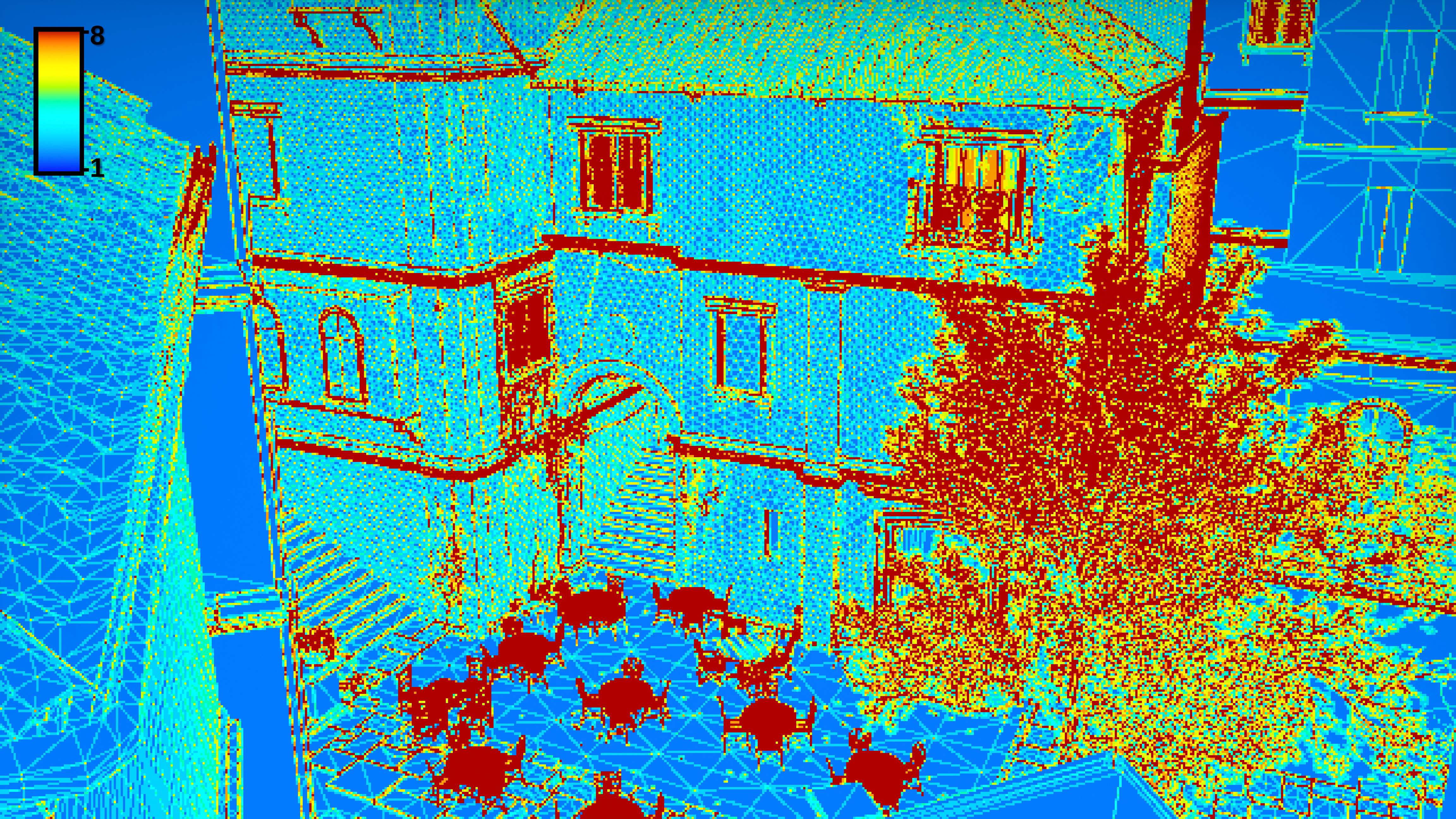
The Mummy – [© Universal Pictures / Digital Domain / Rhythm&Hues]

Overview

- **High frequency shading is too costly**
- Idea: Strong decoupling of shading rate
  - Shading statistical geometry distributions






8
1


Single-sampled
MSAA 8x
AGAA (2A)

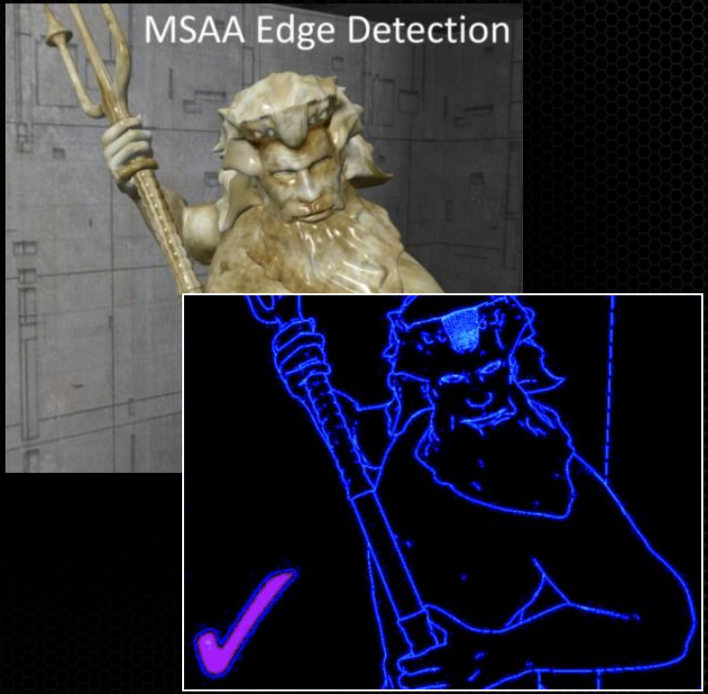

Single-sampled
MSAA 8x
AGAA (2A)


Single-sampled
MSAA 8x
AGAA (2A)

Related work 1/2

Deferred shading

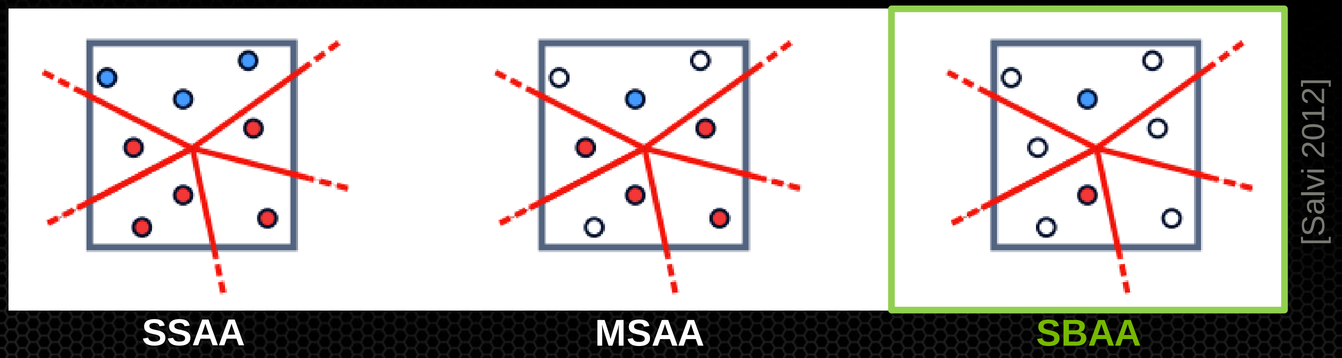
- **Simple/Complex** [Lauritzen 2010]
  - Segmentation based on geometric complexity
  - Shading amortization:
    - Across same planar surface
    - ...Or almost planar -> Quality reduction
  - Simple scenes + large memory requirement

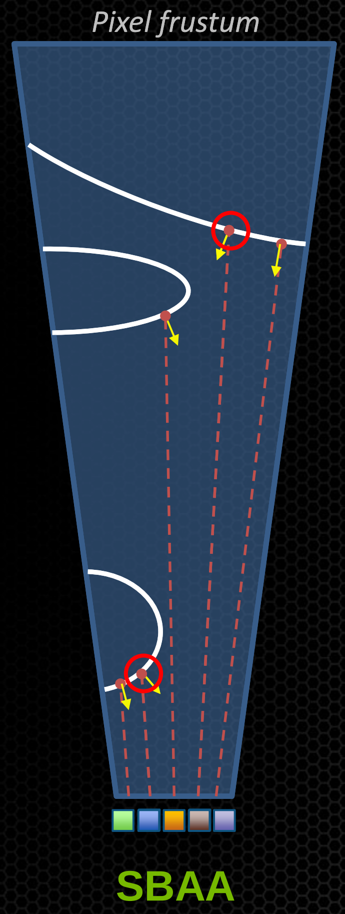


Credit: Crytek [Sousa 2013]

Related work 2/2

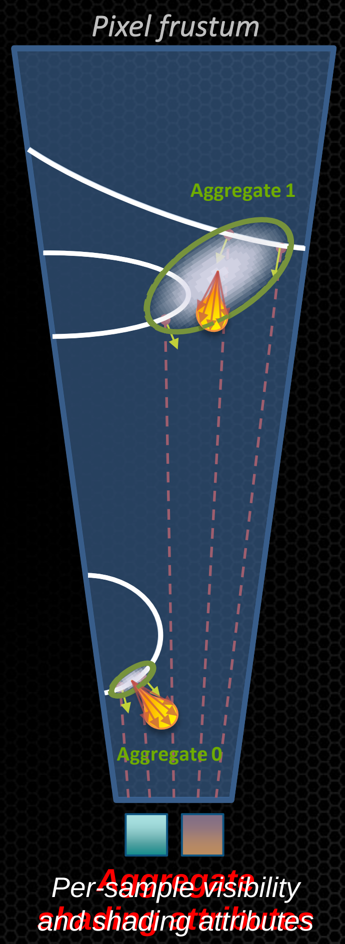
- Surface Based Anti-Aliasing (**SBAA**) [Salvi 2012]
  - High frequency visibility
    - Simplified geometry pre-pass
  - Only store and shade the *n* most contributing surfaces
  - Non-shaded surface information discarded
    -> Aliasing in complex regions



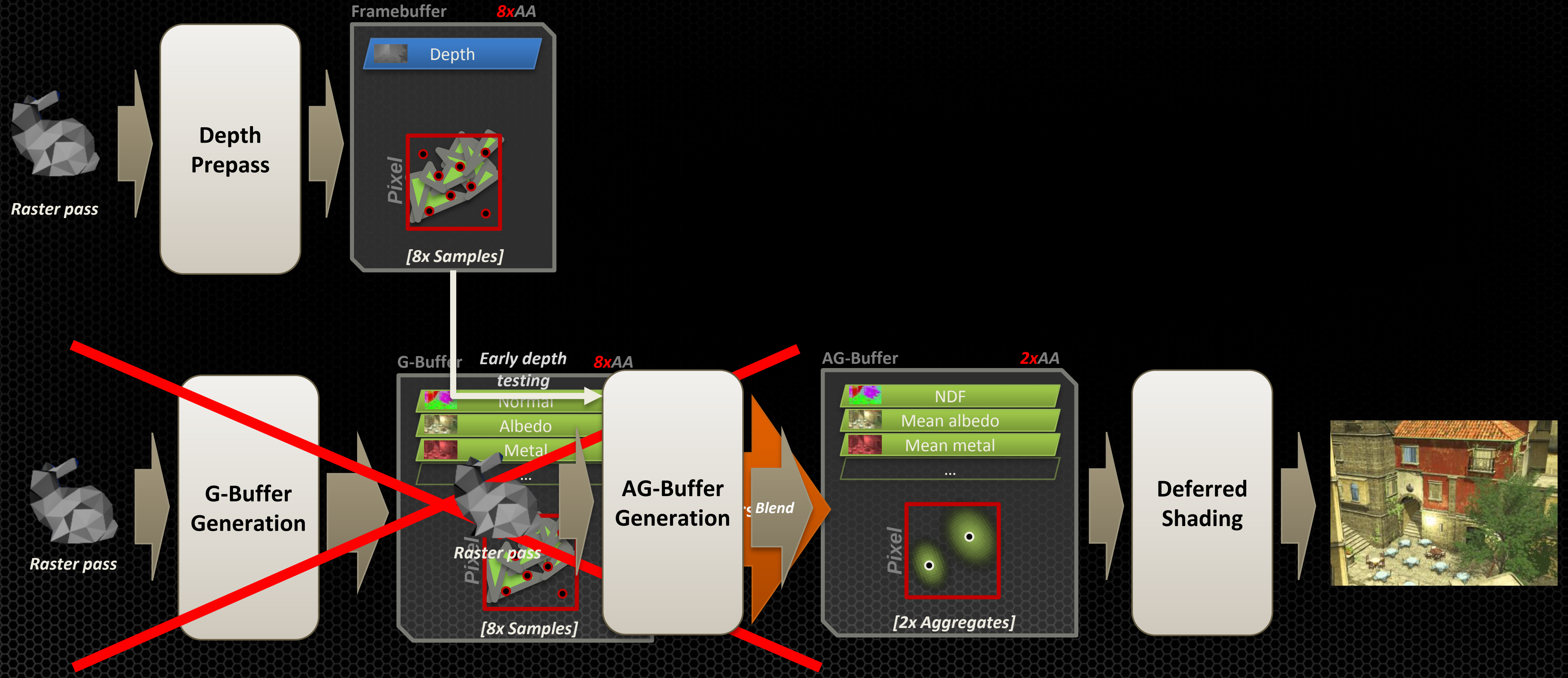


AGAA overview

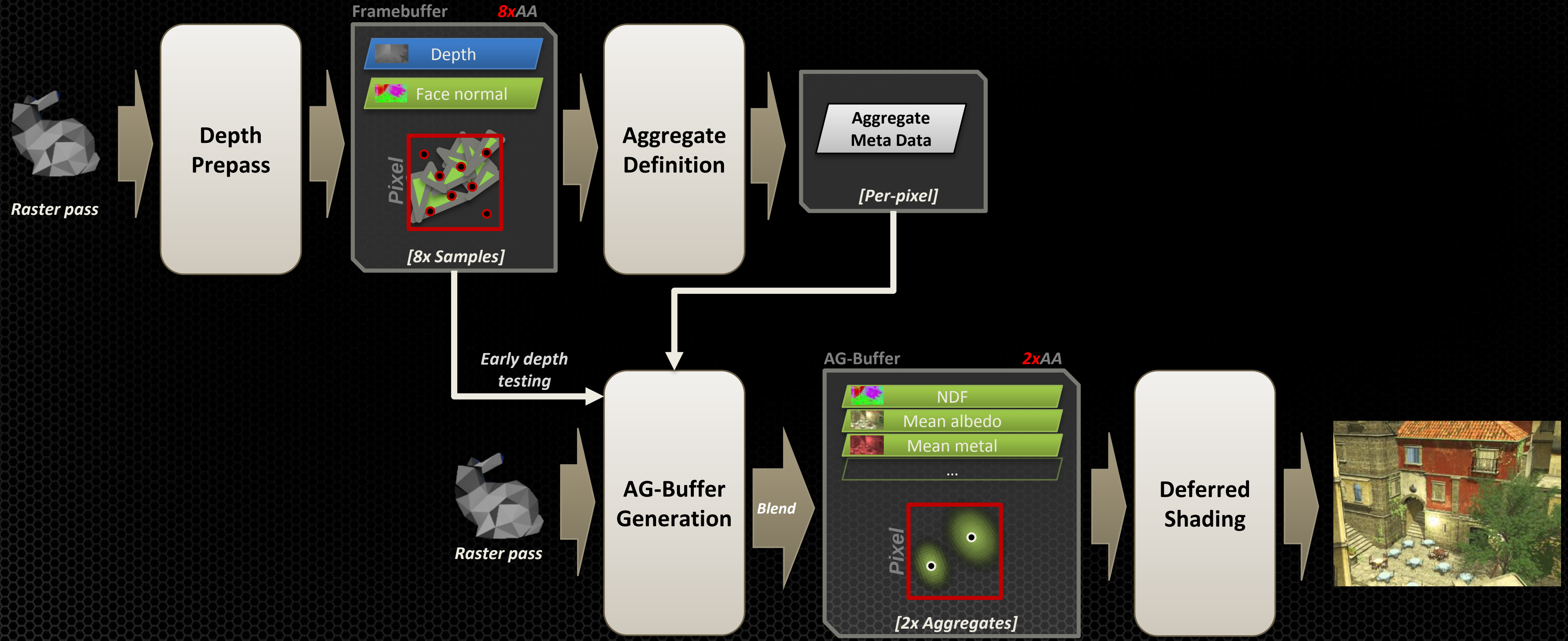
- Integrate ALL geometry samples
  - All surfaces + <u>curvature</u>
- Accumulate and filter geometry samples in pixel-space before shading
  - Per-sample visibility (Z-buffer)
  - Similar to texture/voxel-space pre-filtering
    - But *pixel-space* + *on-the-fly* aggregation
  - *Aggregate G-Buffer* statics:
    - Normal Distribution Function *(NDF)*
      + sub-pixel position distribution.
    - Average Albedo, Metal coef., emissive, etc.



Rendering with Aggregates

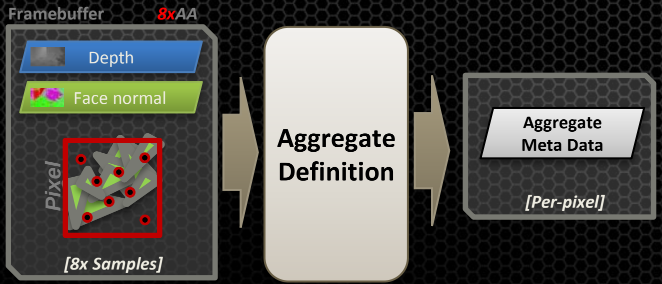


Rendering with Aggregates



NVIDIA

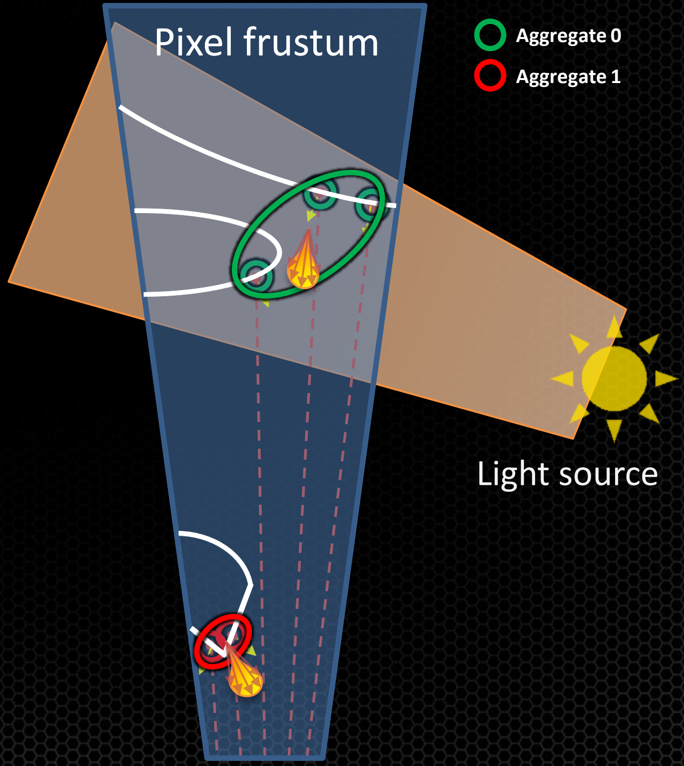
Aggregate definition



- Clustering visibility samples:
  - Cross-primitives + Disjoint surfaces

- **Goal**: Minimize shading errors due to correlated attributes [Bruneton and Neyret 2012]
  - Distance function:

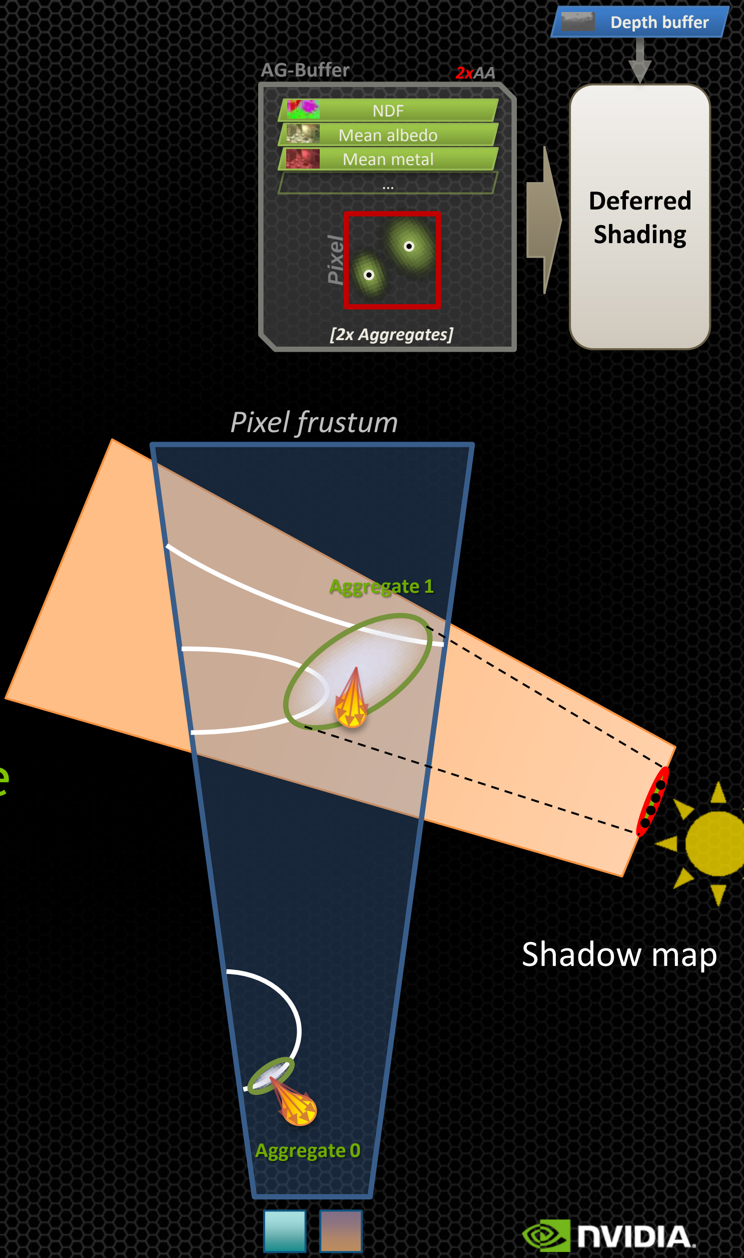
  Orientation + distance-based clustering
  - Control over characteristic length of the scene



NVIDIA

Deferred Shading

- Similar to shading from filtered texture maps
  - *AGAA is independent from the shading model*
- We used the Blinn-Phong BRDF model
  - Filtering Specular component using Toksvig [Toksvig 2005]
  - +Analytic approx. from Toksvig for Lambertian diffuse [Baker and Hill 2012]
- Shadowing must be filtered

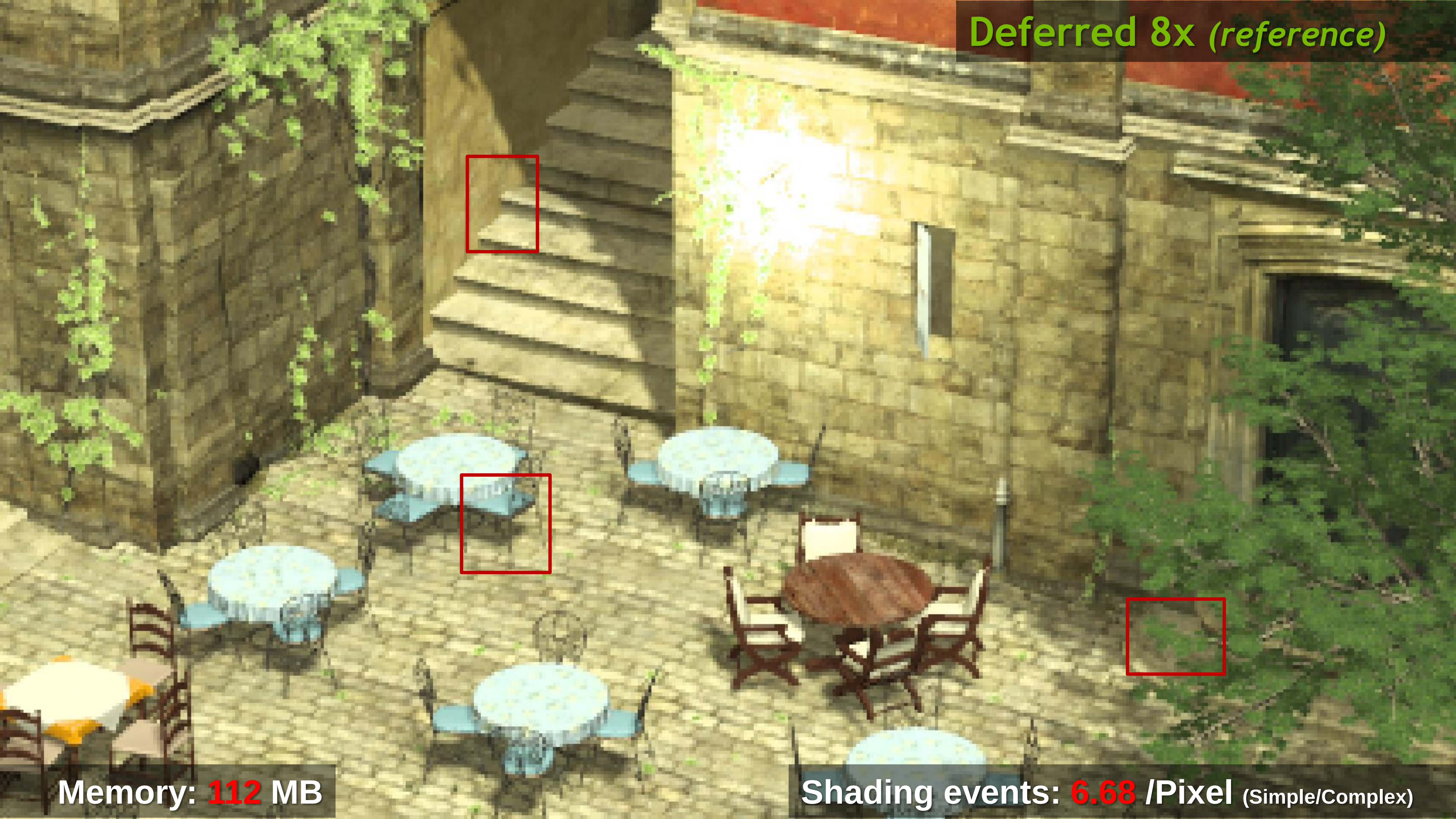


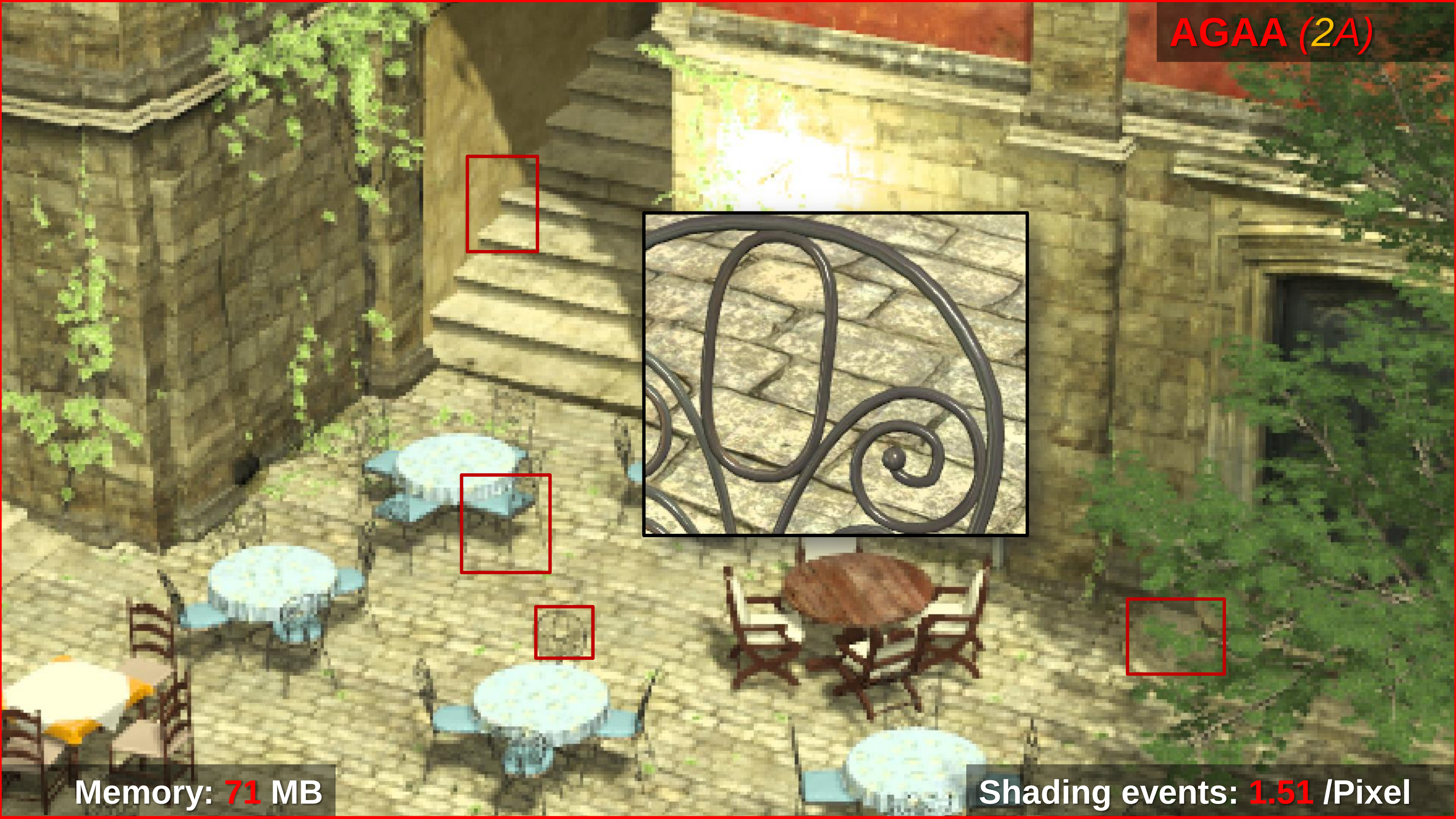

Deferred 8x (reference)


FXAA


AGAA (1A)
Memory: 41 MB
Shading events: 1 /Pixel

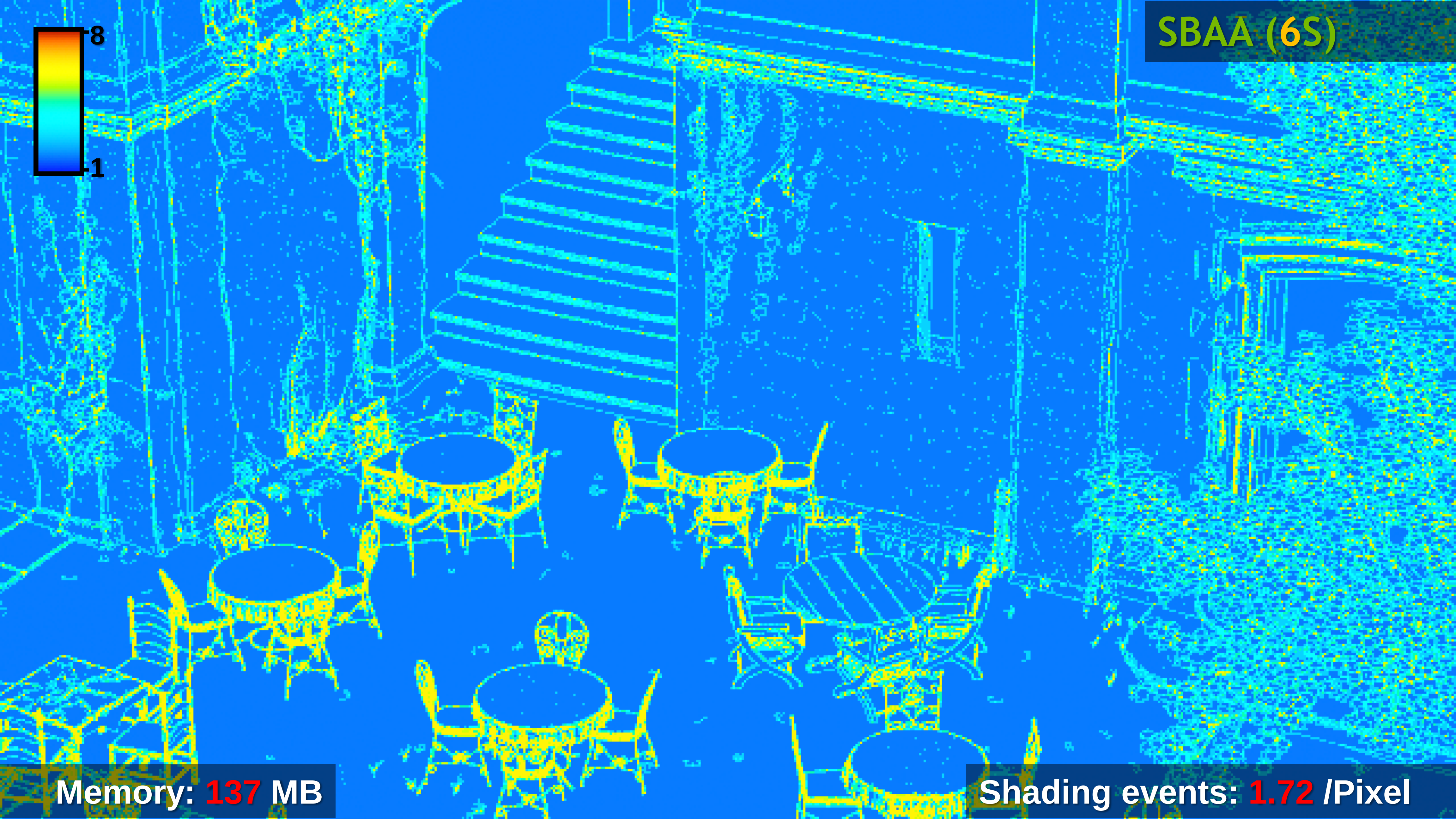

AGAA (2A)
Memory: 71 MB
Shading events: 1.51 /Pixel


Deferred 8x (reference)
Memory: 112 MB
Shading events: 6.68 /Pixel (Simple/Complex)

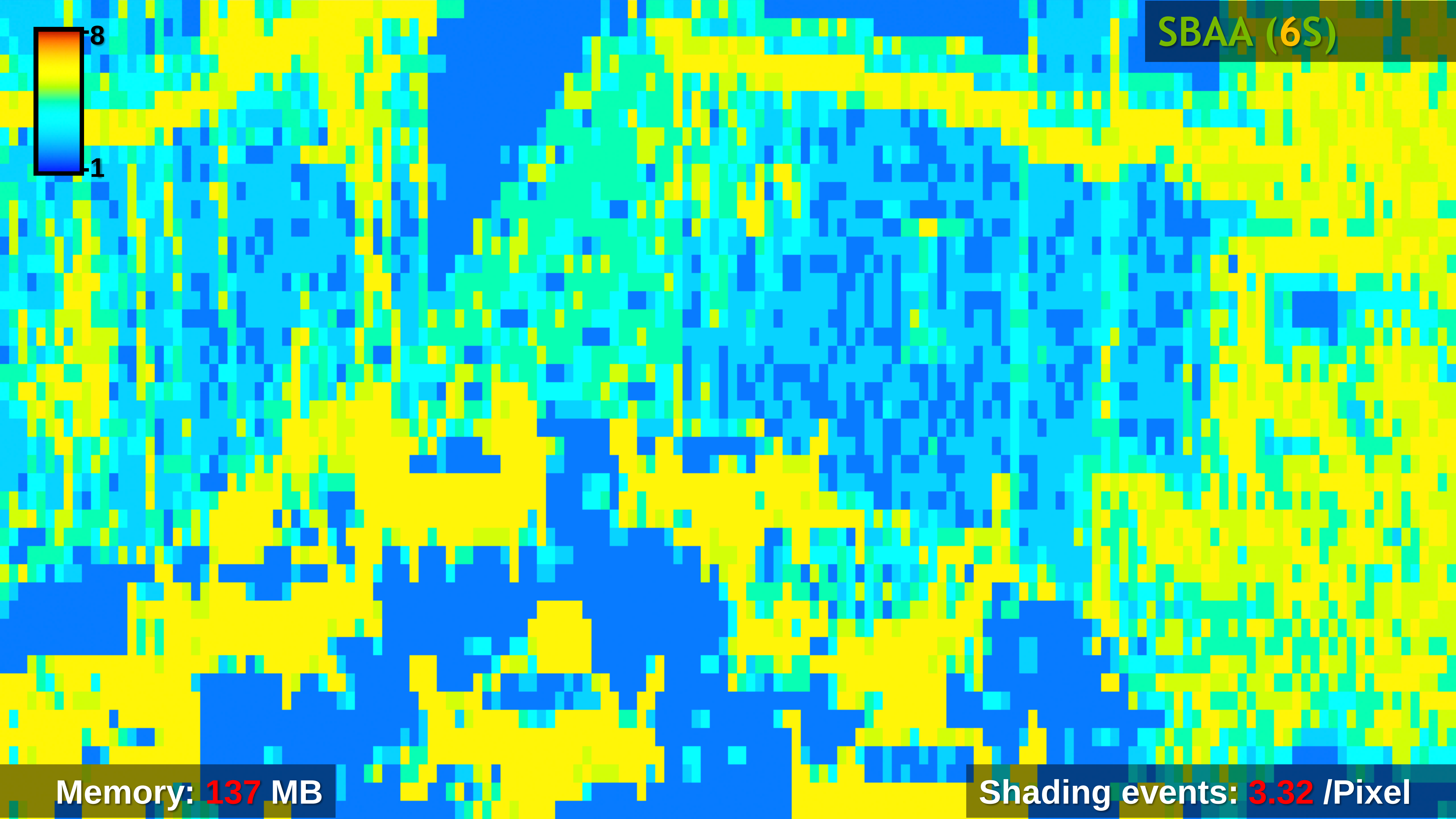

AGAA (2A)
Memory: 71 MB
Shading events: 1.51 /Pixel

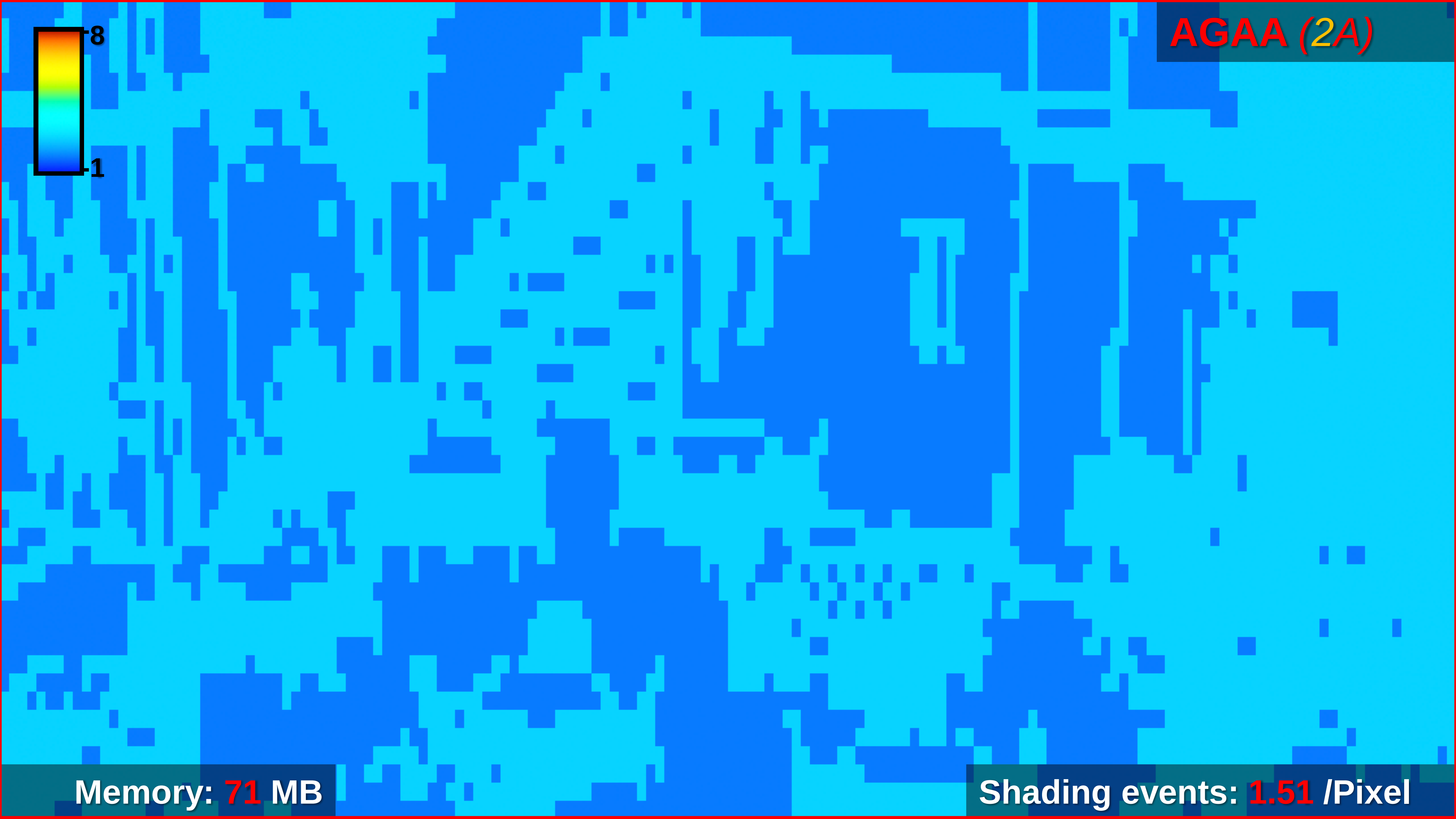

SBAA (2S)
Memory: 88 MB
Shading events: 1.72 /Pixel


SBAA (6S)
Memory: 137 MB
Shading events: 3.32 /Pixel



8
1
SBAA (6S)
Memory: 137 MB
Shading events: 1.72 /Pixel


8
1
SBAA (6S)
Memory: 137 MB
Shading events: 3.32 /Pixel


8
1
AGAA (2A)
Memory: 71 MB
Shading events: 1.51 /Pixel

Results : Performance

Deferred shading @8x MSAA 720p - Comparison with Simple/Complex [Lauritzen 2010] - NVIDIA GTX980 (Maxwell GM204)

54% Faster rendering than
Simple/Complex
(2.84x Faster shading)

Old City

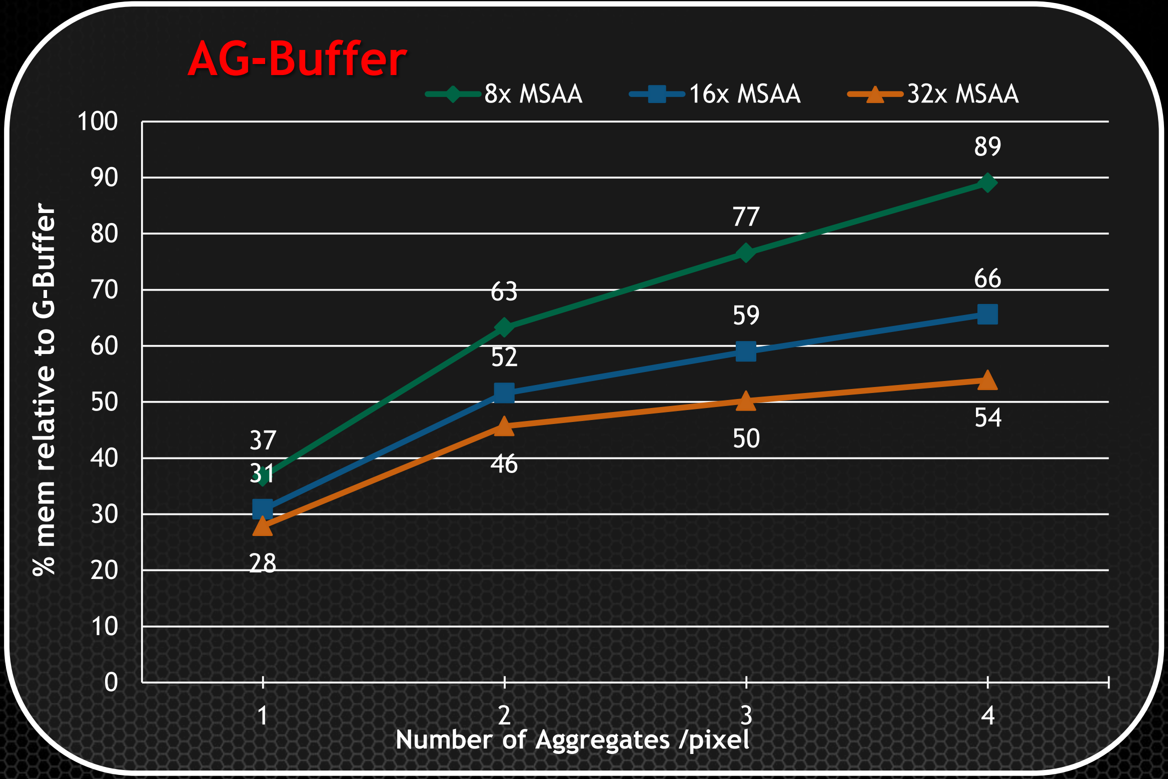
Time (ms)	AGAA					Deferred Simple/Complex [Ref]			
Scene	1. Prepass	2. Aggregate Def.	3. AG-Buffer Generation	4. Shading	Total	G-Buffer Gen.	Simple/ Complex	Shading	Total
OldCity	2.61	0.6	3.8	3.6	**10.61**	5.73	0.41	10.24	16.38
UE3 FoliageMap	2	0.6	2.47	3.67	**8.74**	4.35	0.41	10.45	15.21

Results: Memory savings

- % Memory relative to super-sampled G-Buffer
  - 37% saving @8x - 2 Aggregates
  - *20% less than SBAA @2 Surfaces - ~40% less @Iso quality*

AGAA G-Buffer layout:
[16B/Aggregate + 6B/Sample]

G-Buffer layout: [16B/sample]



NVIDIA

Constraints and Limitations

- Unified material with unique shading model
  - No material switch (Skin, water, hair, cloth…)
  - All shading inputs must be filterable !
- NDF precision:
  - A few very differently oriented surfaces
  - But uni-modal Gaussian distrib.
    - Specular sparkling
- Correlation issues:
  - Lit green foliage + Shadowed red wall

1 Aggregate/ pixel

Conclusion

- Path forward for very high sampling rates in real-time
  - Scalable: Bounded storage and shading rate

Properties:

- Cross-primitives + Cross-surfaces amortization
- All geometric details integrated

Remaining work on pre-filtering schemes for advanced unified shading/material models

THANK YOU !

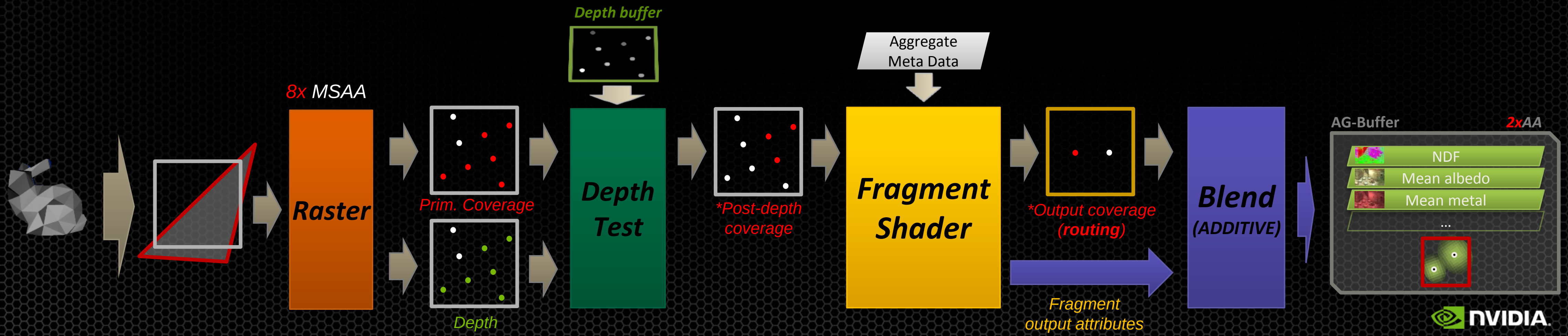
Questions ?

BACKUPS

*[Maxwell or DX12 hardware specific features]

Step 3: AG-Buffer generation

- Rasterize @ sample frequency (Eg. 8x MSAA)
  - Inside a set of color buffers @ Aggregate frequency (Eg. 2x) *[TIR M->N]
  - Per-sample early depth testing
- Fragment shader:
  - Generates attributes for accum., weighted by visible coverage *[Post-depth coverage]
  - Route output attributes into one of the aggregates *[Out coverage override]
- Additive blending:
  - Additive blending inside AG-Buffer (weighted sum)



NVIDIA