NVIDIA

DLSS 2.0 - IMAGE RECONSTRUCTION FOR REAL-TIME RENDERING WITH DEEP LEARNING

Edward Liu, Senior Research Scientist, March 23rd 2020

OUR TEAM

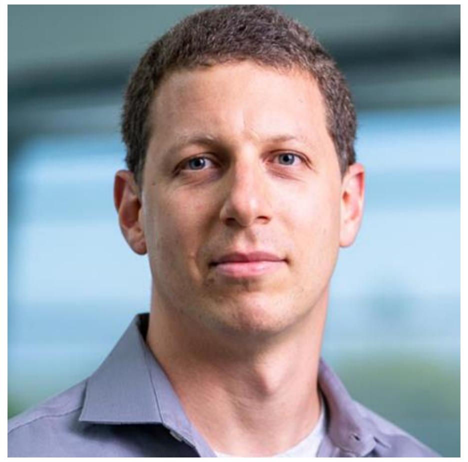

Edward(Shiqiu) Liu

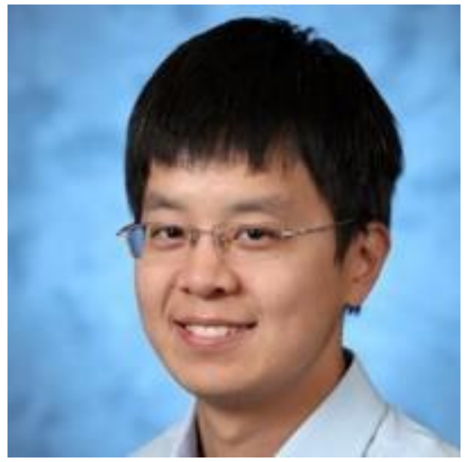

Robert Pottorff

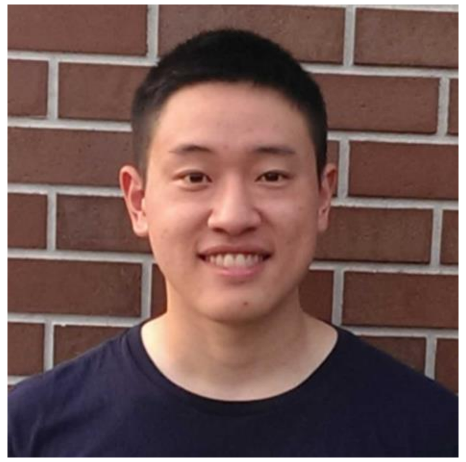

Guilin Liu


Karan Sapra


Jon Barker


David Tarjan


Lei Yang


Kevin Shih


Marco Salvi


Andrew Tao


Bryan Catanzaro

AGENDA

DLSS 2.0 Introduction

Overview of high-level features of DLSS2.0

Challenges in Image Super-Resolution for Gaming

Explains why super-res for gaming is difficult and how DL is providing value

Integrating DLSS 2.0

Tips for integration DLSS2.0 in a modern game engine

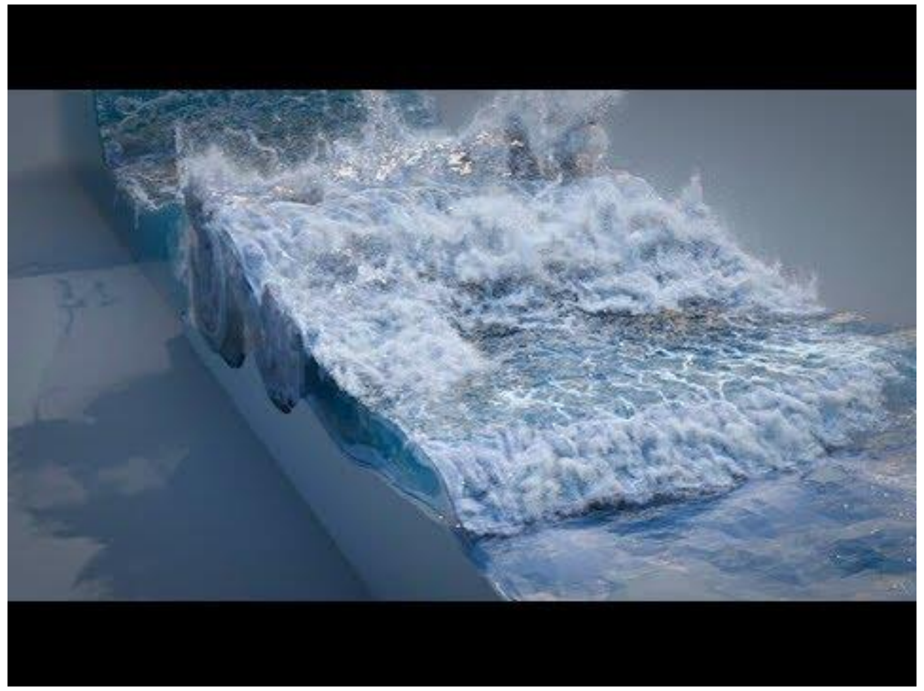
NEXT GEN GAMES NEED SUPER RESOLUTION

Ray tracing, physics, AR/VR, and higher resolution displays drive up GPU computing needs exponentially

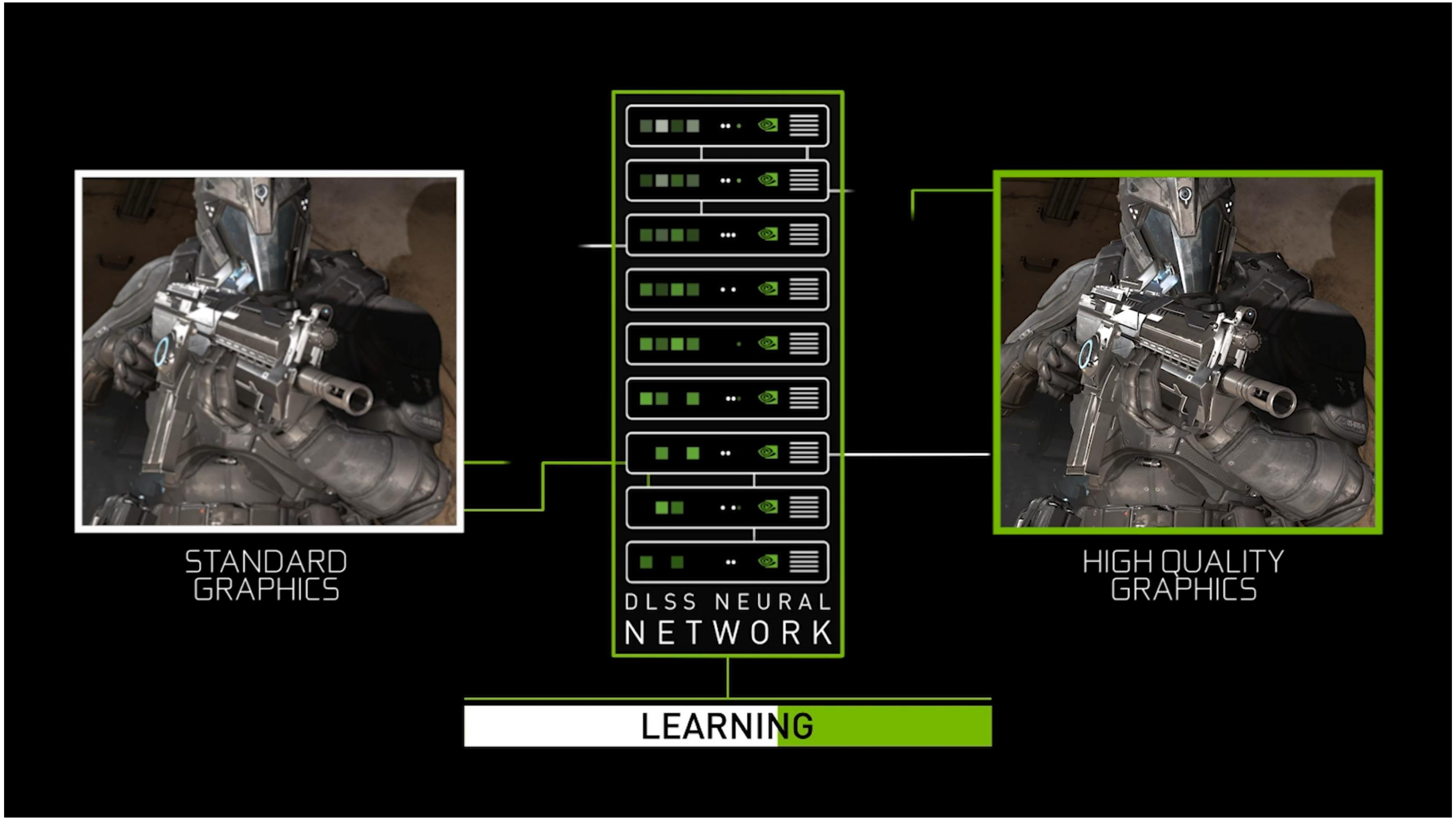
Ray tracing alone can demand many times the computing power of traditional rendering techniques

Super resolution technique become necessary

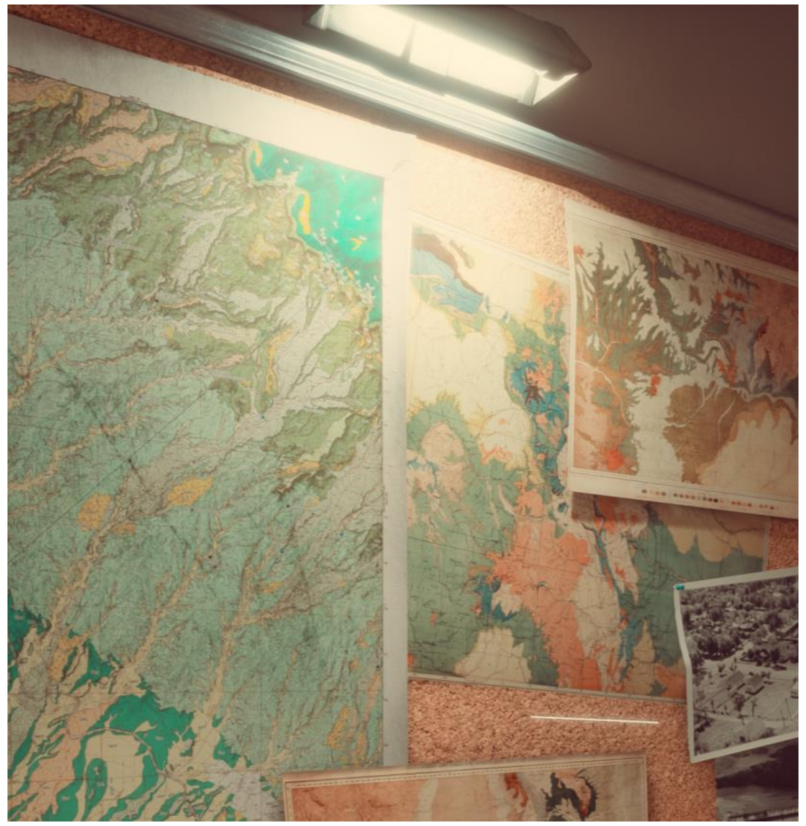
RTX GPUs have tensor cores to accelerate deep learning workloads



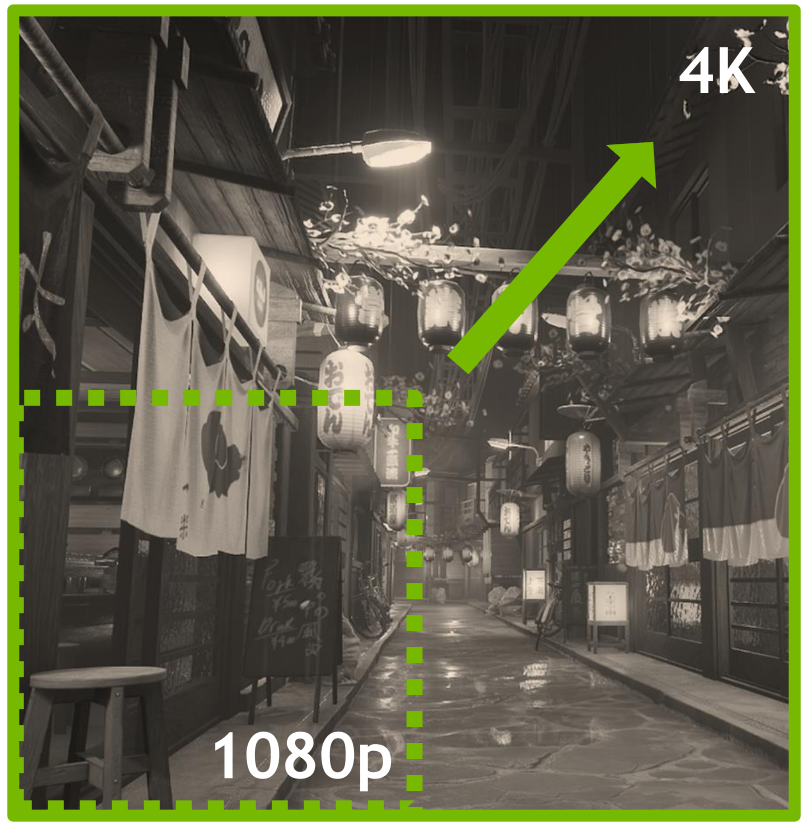
DLSS OVERVIEW



INTRODUCING DLSS 2.0

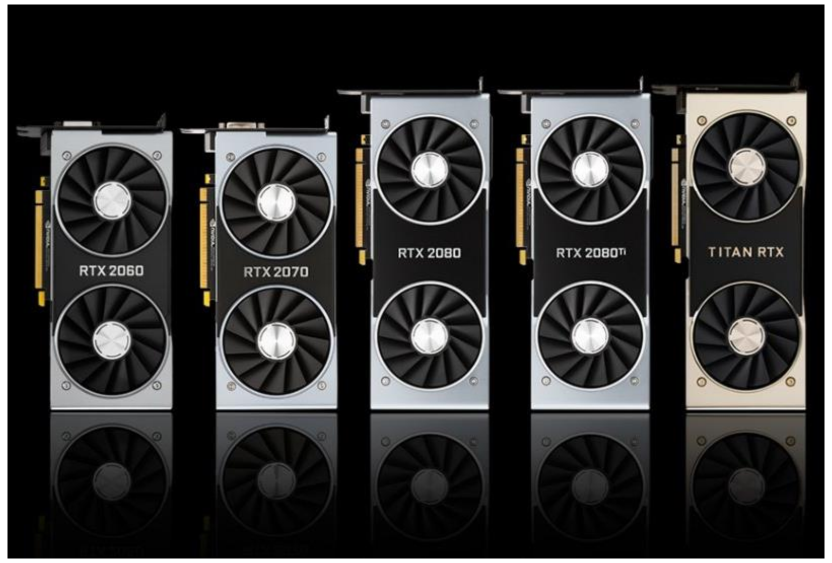


Great Image Quality
Details rival native resolution



4x Upscaling Ratio
540p to 1080p, 1080p to 4k

Generalized Model
One model to rule them all!



1.5ms at 4K on 2080Ti
Works on all RTX GPUs at all resolution


720p

DLSS 1.0 720p to 1080p

DLSS 2.0 720p to 1080p


1080p


540p

DLSS 2.0 540p to 1080p


540p
DLSS2.0
720p
DLSS1.0
1080p TAA
540p TAA

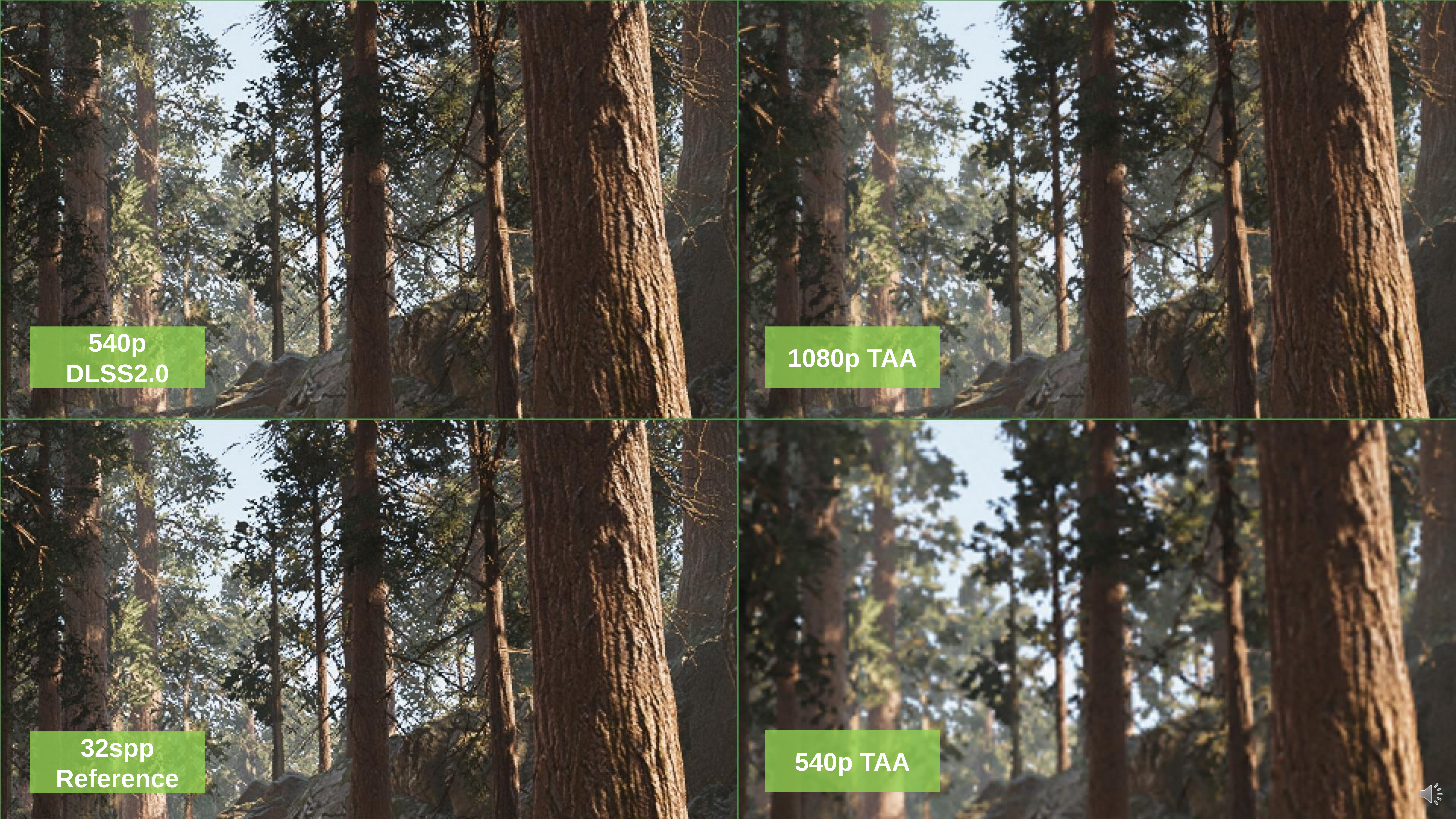

540p
DLSS2.0
720p
DLSS1.0
1080p TAA
540p TAA


540p - 89fps


1080p - 48fps


540p to 1080p w/ DLSS2.0 - 86fps


32spp Reference 1080p


540p
DLSS2.0
1080p TAA
32spp
Reference
540p TAA

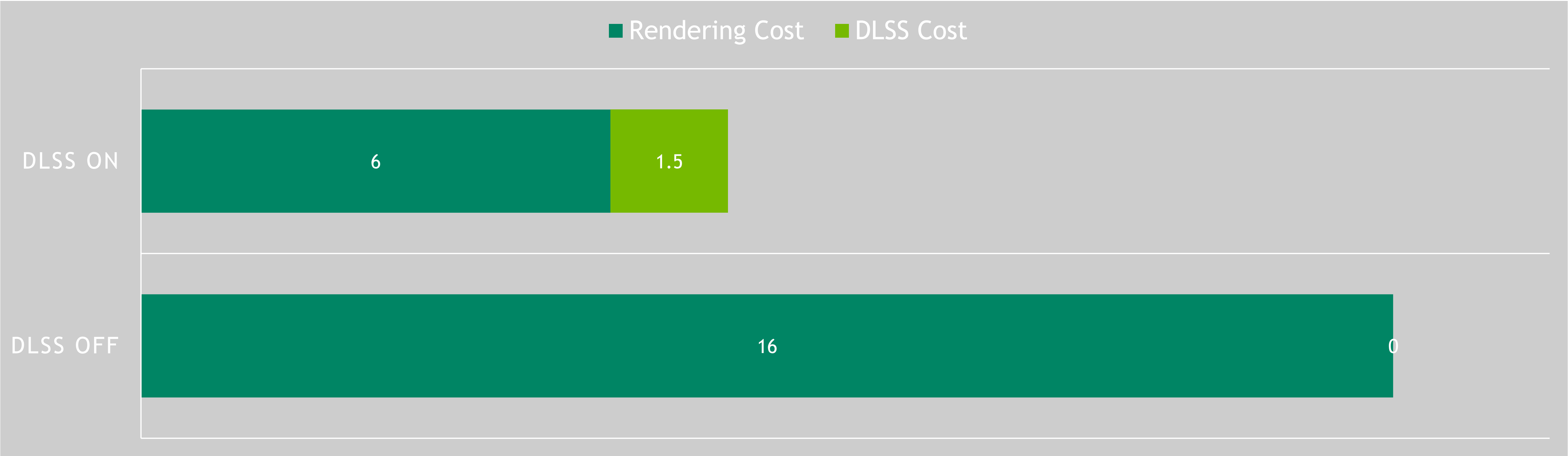
DLSS 2.0 PERFORMANCE MODES

Three different quality modes:

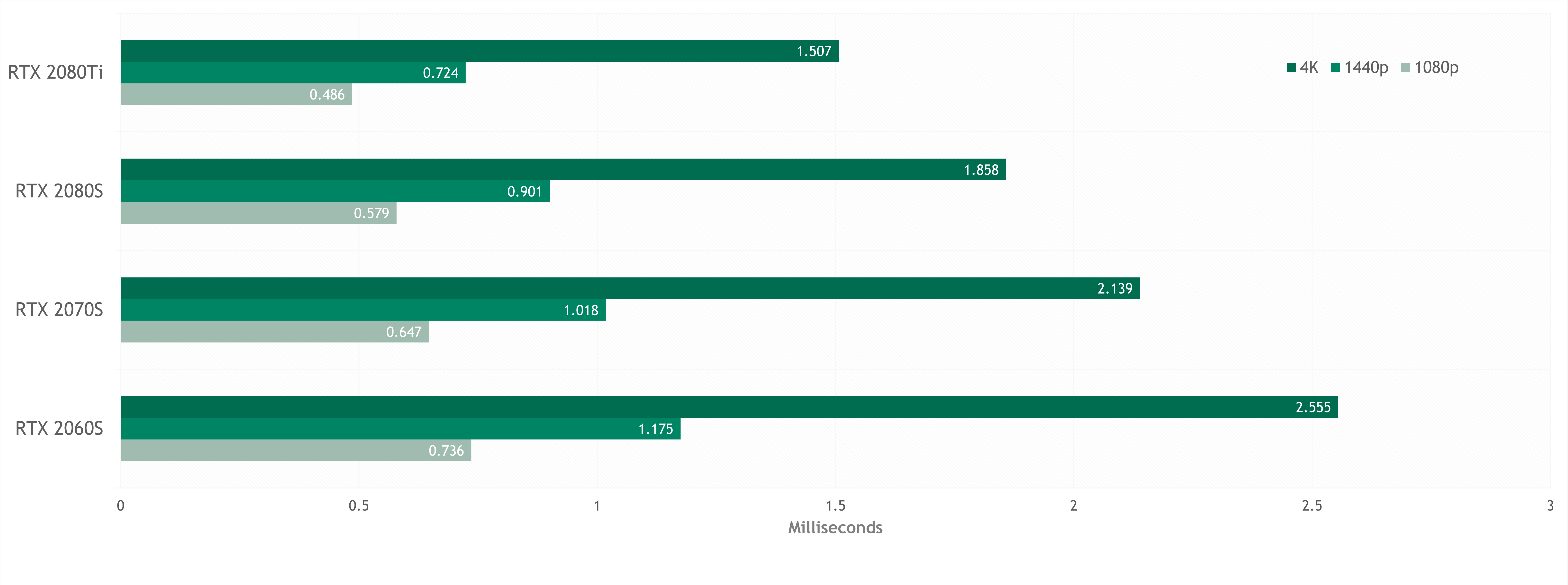
Performance, Balanced, Quality

4x, 3x, 2x pixels upsampling

DLSS 2.0 - ACCELERATED RENDERING



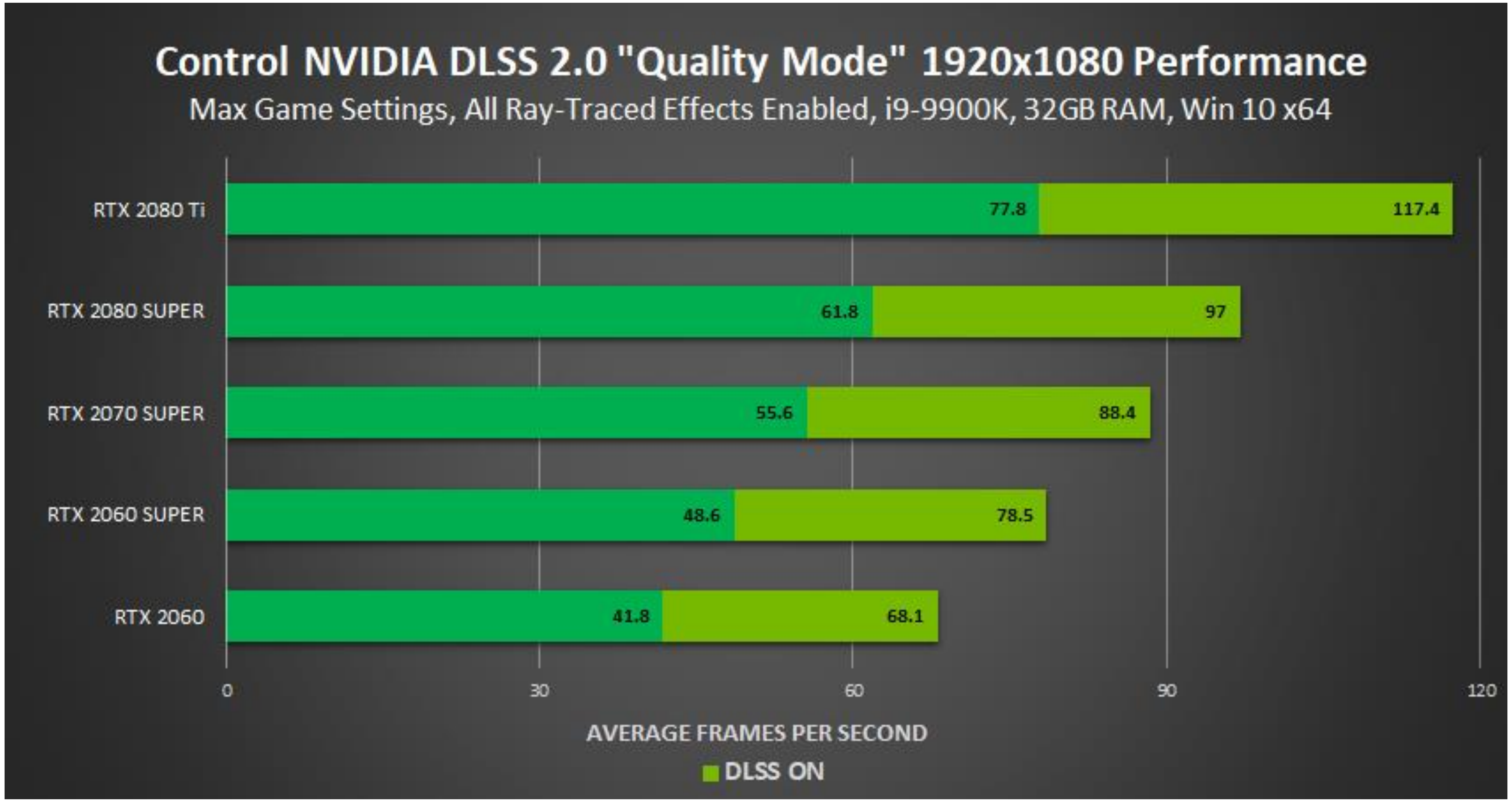
DLSS 2.0 COST



RTX 2080Ti
RTX 2080S
RTX 2070S
RTX 2060S
1.507
0.724
0.486
1.858
0.901
0.579
2.139
1.018
0.647
2.555
1.175
0.736
4K
1440p
1080p
0
0.5
1
1.5
2
2.5
3
Milliseconds

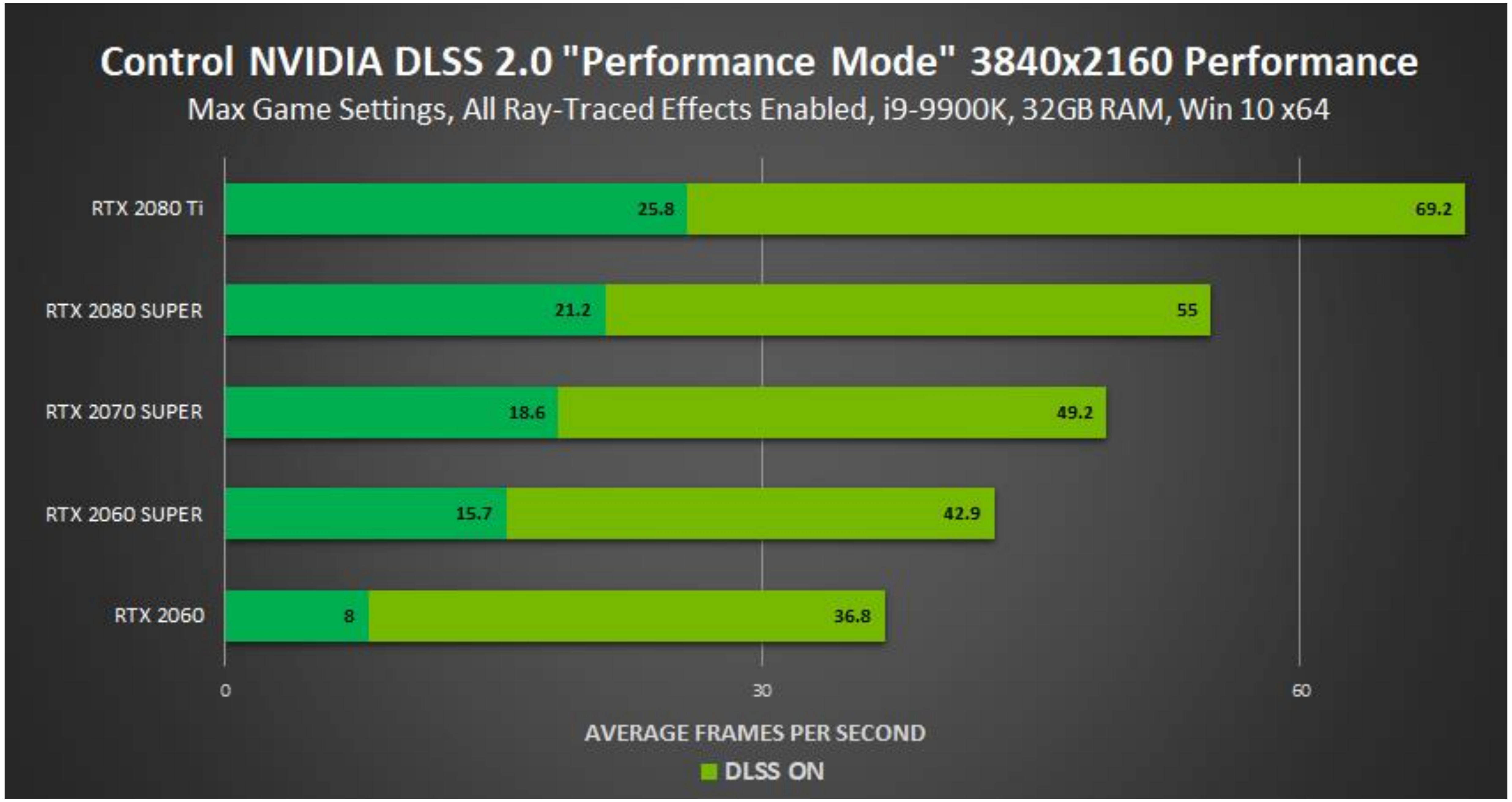
DLSS 2.0 PERFORMANCE BOOSTS

Quality Mode – 720p to 1080p



DLSS 2.0 PERFORMANCE BOOSTS

Performance Mode – 1080p to 4K



“DLSS is impressive to the point where I believe you'd be nuts not to use it.”

— Digital Foundry

“The upscaling power of this new AI driven algorithm is extremely impressive… it’s basically a free performance button.”

— Hardware Unboxed

SHIPPING IN THE FOLLOWING TITLES

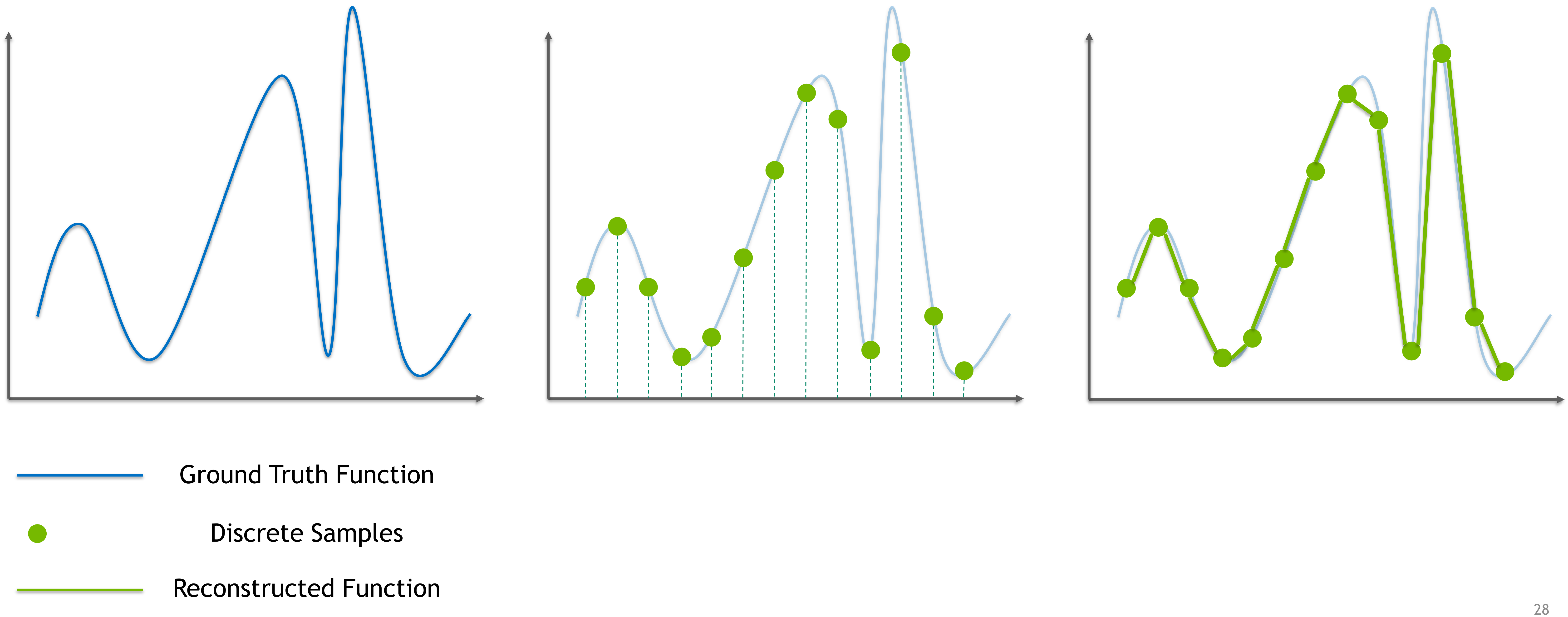
WOLFENSTEIN: YOUNGBLOOD &
DELIVER US THE MOON
AVAILABLE NOW

MECHWARRIOR 5 & CONTROL DLC
AVAILABLE THIS WEEK

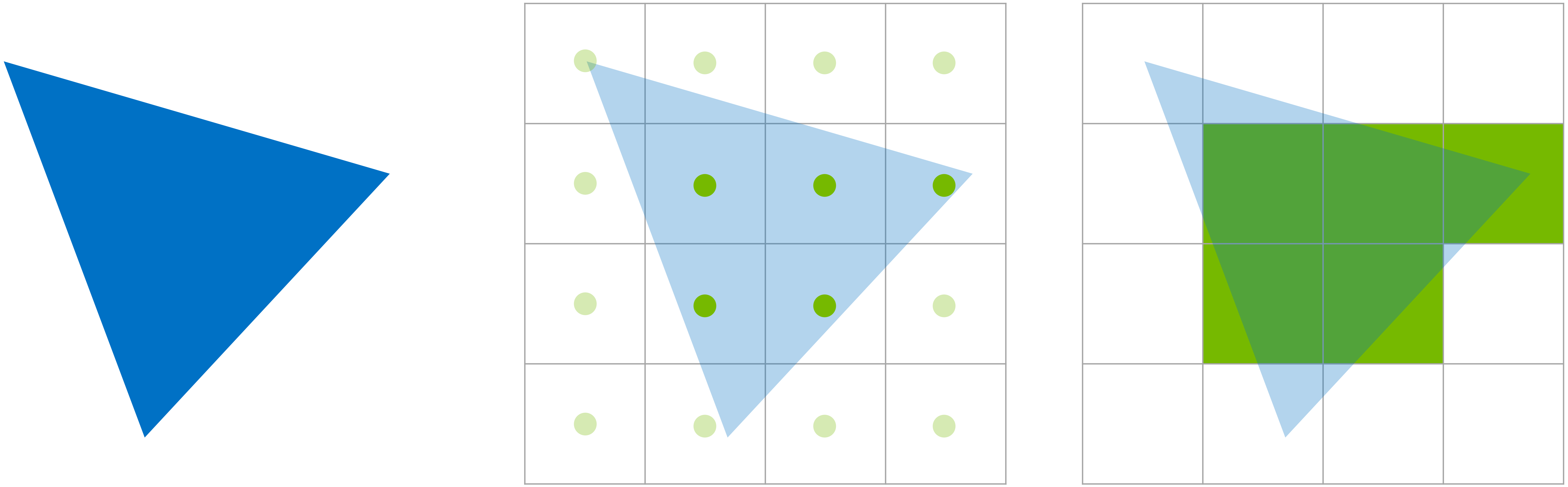
CHALLENGES IN IMAGE SUPER-RES FOR REAL-TIME RENDERING

RECONSTRUCTION 101

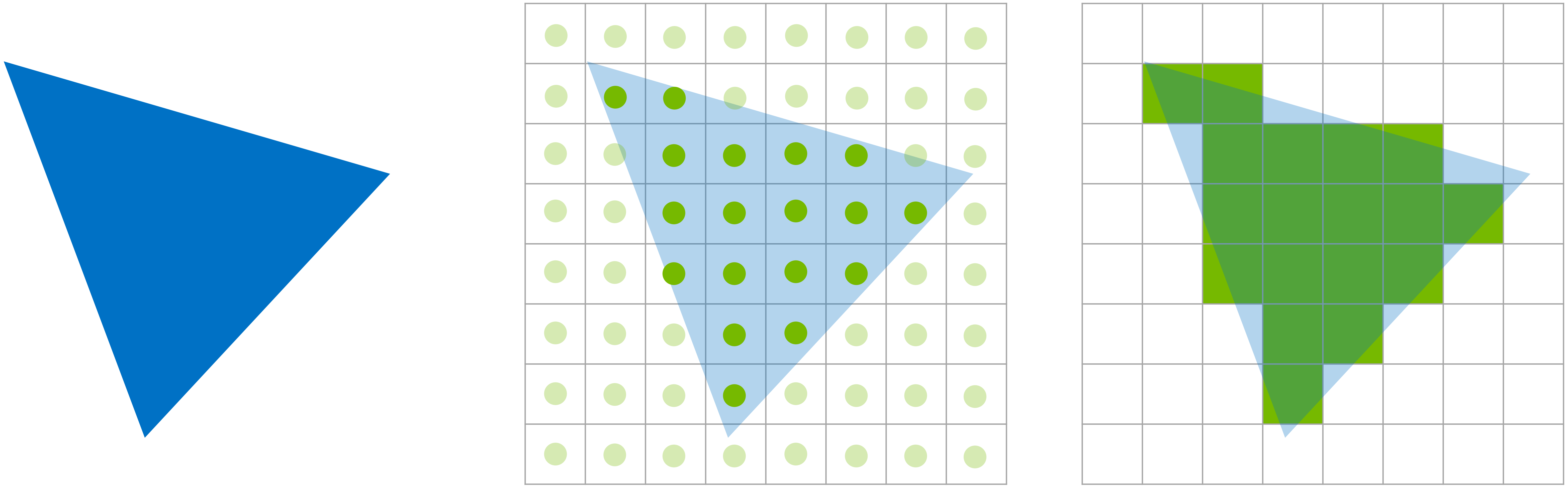
Double the sampling rate



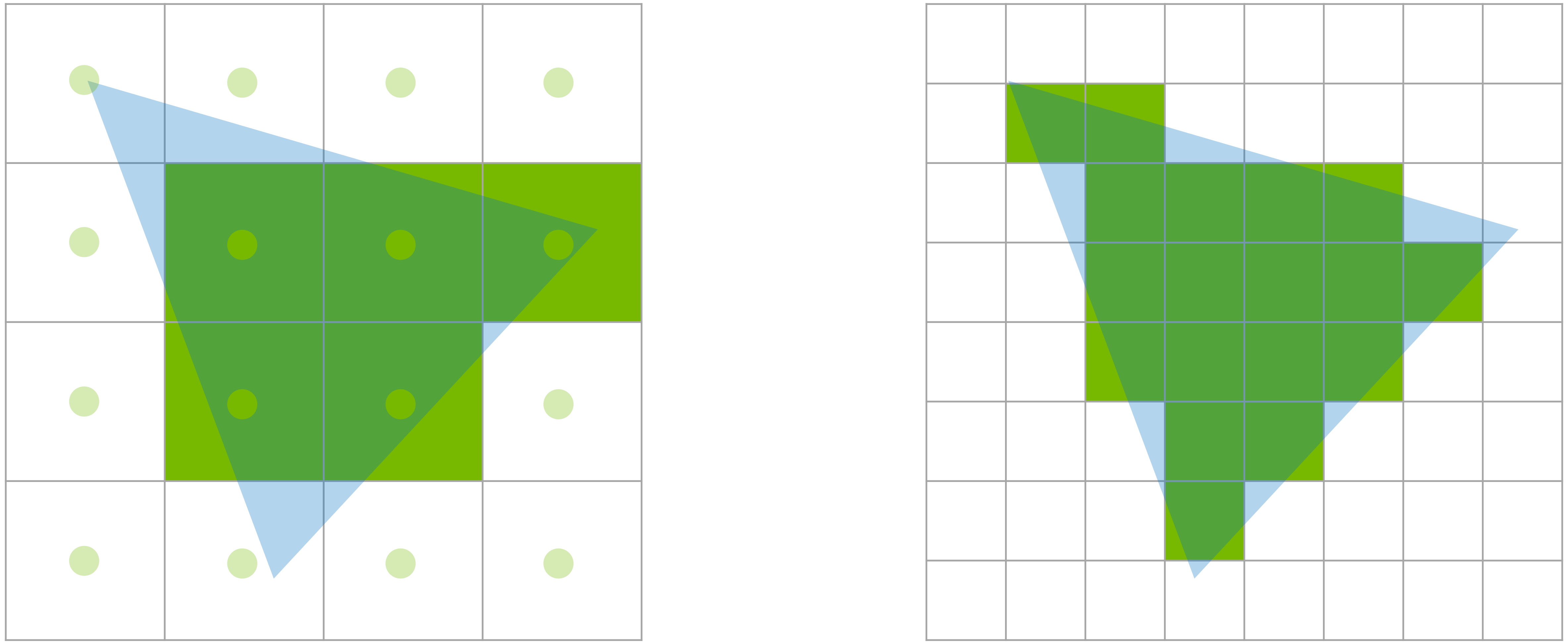
RECONSTRUCTION 101



RECONSTRUCTION 101

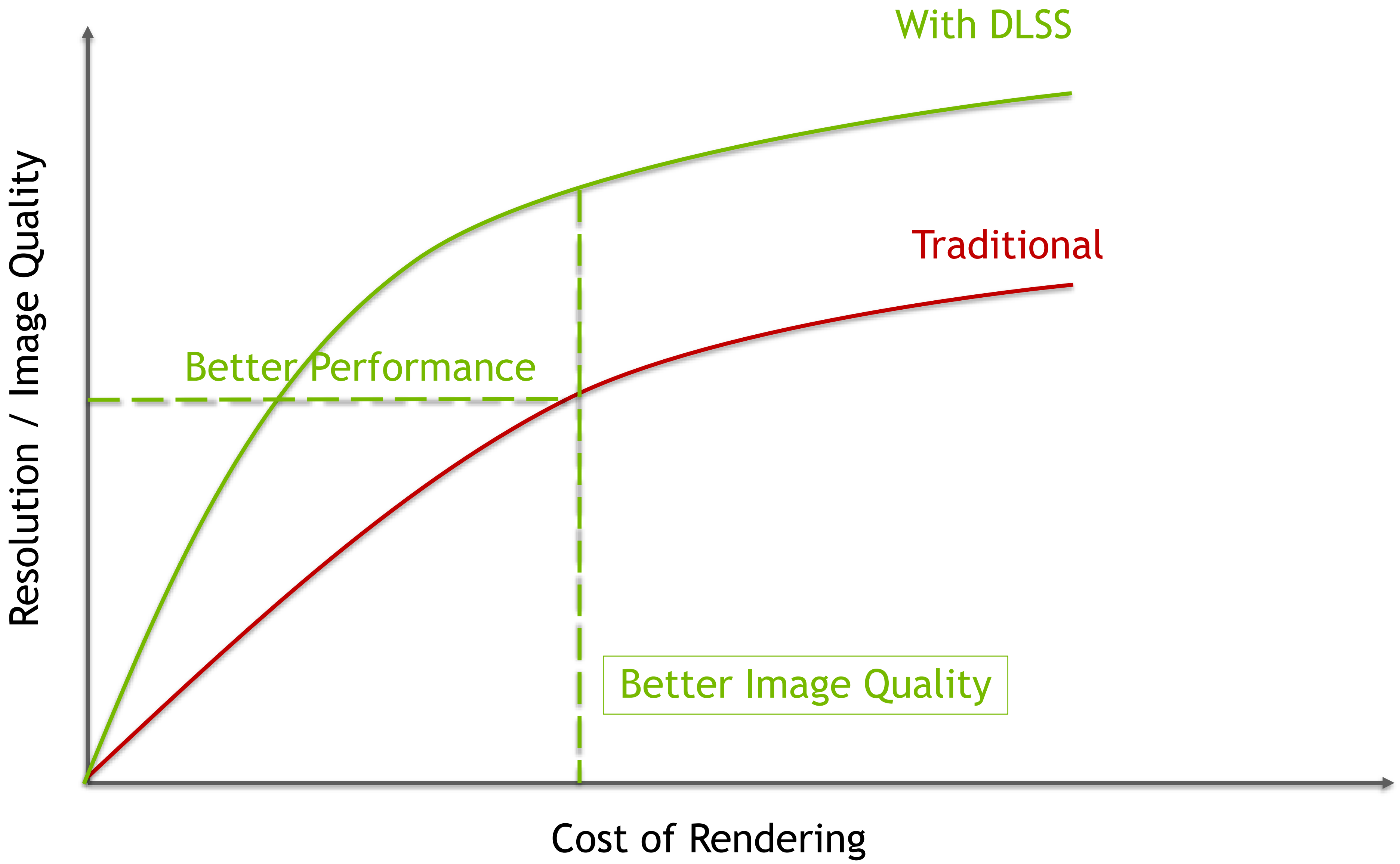


DLSS PROBLEM STATEMENT



Low resolution sampling rate → High resolution reconstruction

DLSS PROBLEM STATEMENT



SINGLE IMAGE SUPER-RES

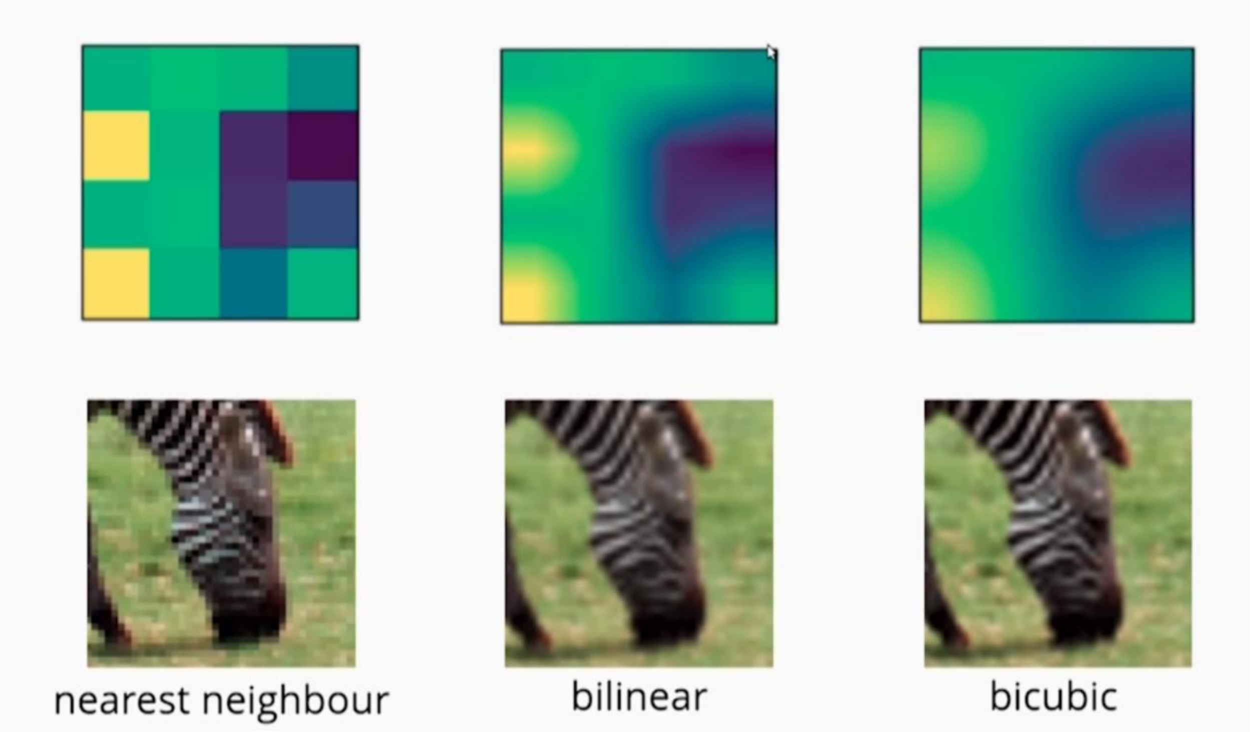
Previous work

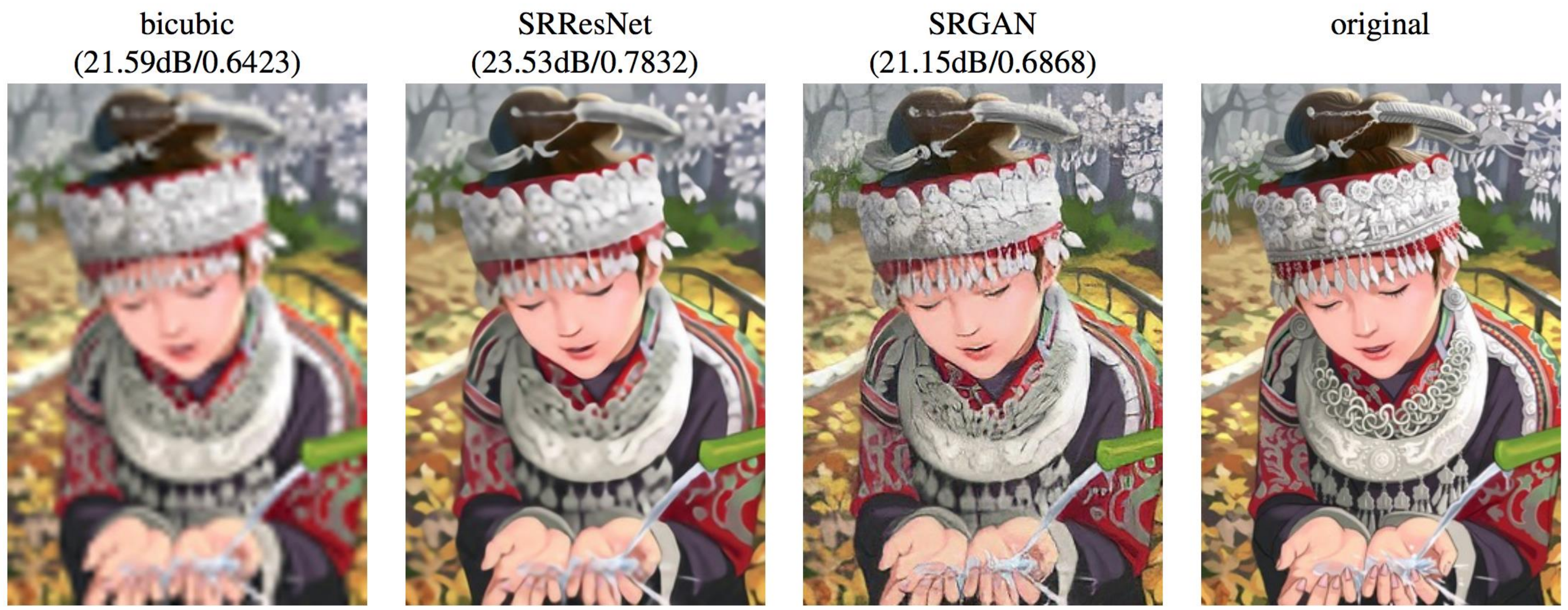
Reconstruct high resolution image by interpolating the low-resolution pixels

Common choices are bilinear, bicubic, lanczos

Contrast aware sharpening

deep neural networks can hallucinate new pixels conditioned on existing pixels based on priors or training data



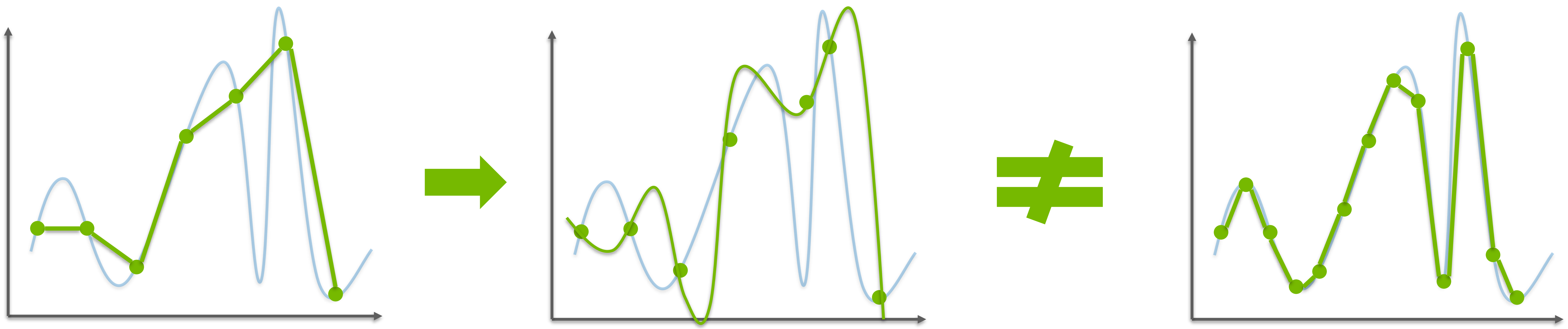


[Ledig et al. 2017]

SINGLE IMAGE SUPER-RES

Resulted images lack details compared to native high-resolution images

Images may be inconsistent with native rendering because of hallucination, and temporally unstable



Reconstructed with linear interpolation

Reconstructed with DL or other interpolation

High res samples

SINGLE IMAGE SUPER-RES

Resulted images lack details compared to native high-resolution images

Images may be inconsistent with native rendering because of hallucination, and temporally unstable

≠

DL Upscaled
720p to 1080p

Native
Rendering
1080p


1080p with TAA


540p to 1080p DLSS2.0


540p Bicubic Upsampled to 1080p


540p to 1080p with ESRGAN


540p
DLSS2.0
1080p TAA
540p
ESRGAN
540p TAA
Bicubic

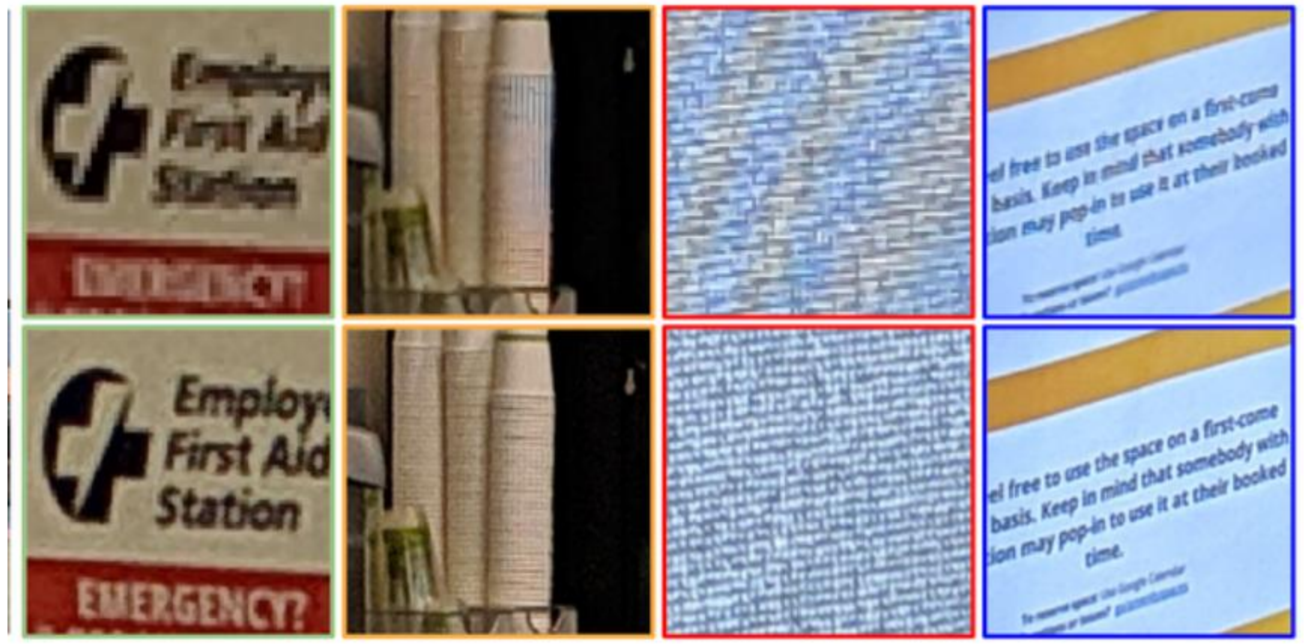
MULTI-FRAME SUPER-RES

Previous work

Less ill-posed than single image super-res, restore true optical details better

Designed for videos or burst mode photography, not leveraging rendering specific information

- Optical flow vs. geometric motion vector
- Pixels vs samples
- Using frames in the future



[Wronski et al. 2019]

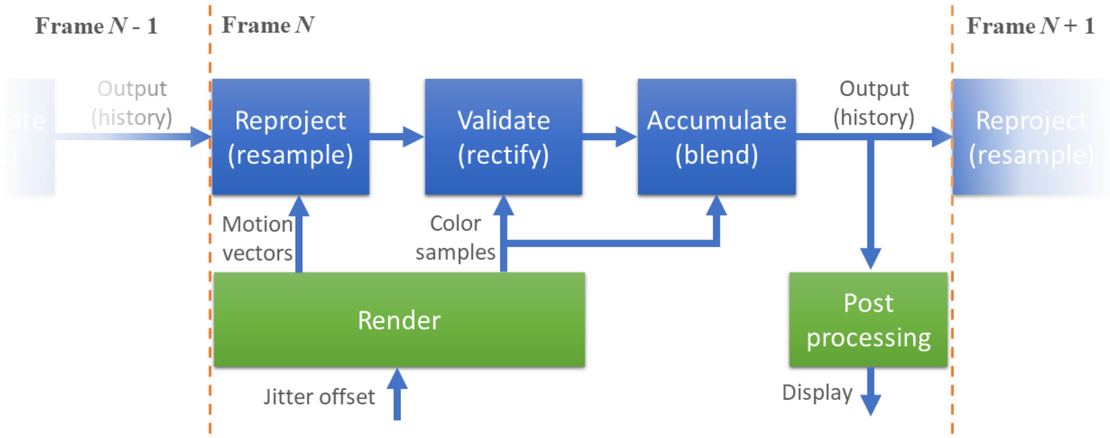
Bicubic Our result HR frame

[Sajjadi et al. 2018]

SPATIAL-TEMPORAL SUPER SAMPLING

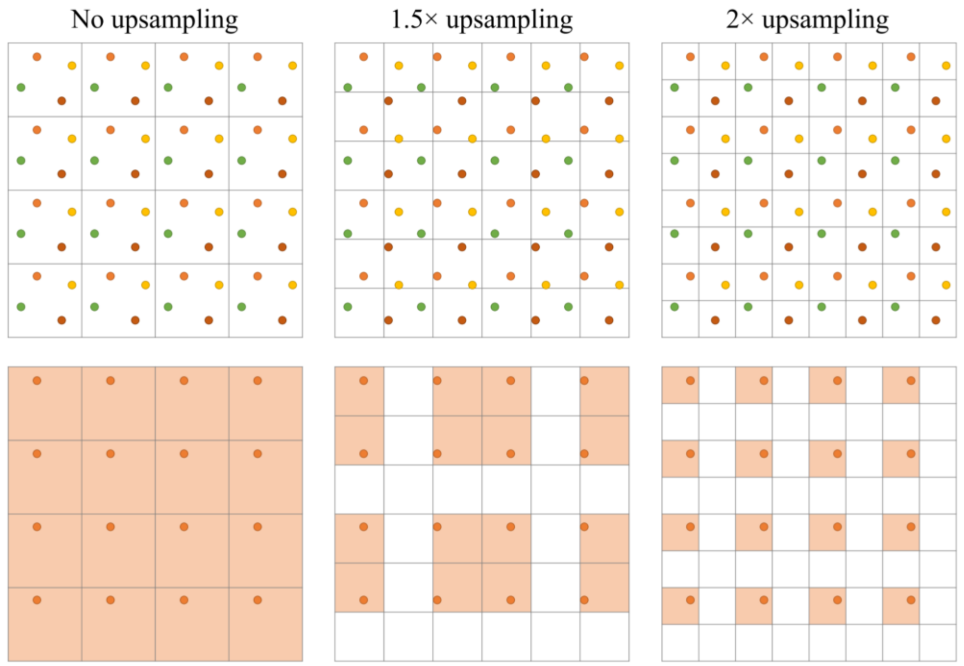
Previous work

Temporal Antilasing (TAA)



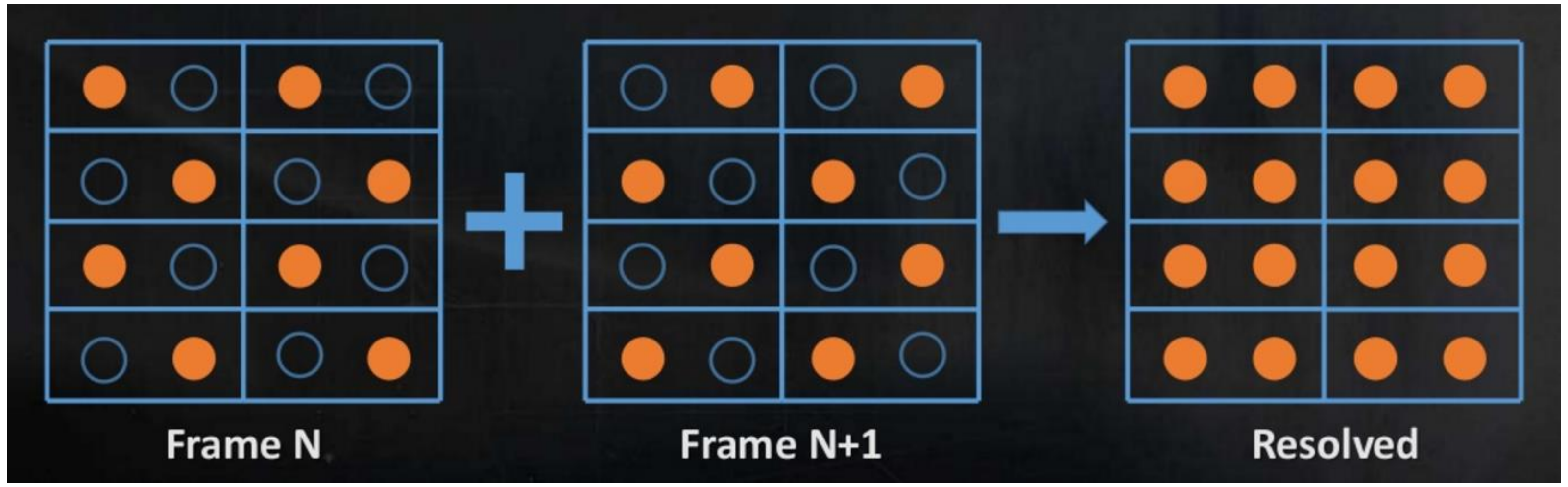
[Yang09, Lottes11, Sousa11, Karis14, Salvi16]

Temporal Upsampling



[Yang09, Herzog10, Malan12, Valient14, Aalto16, Epic18]

Checkerboard Rendering (CBR)



[ElMansouri16, Carpentier17, Wilidal17]

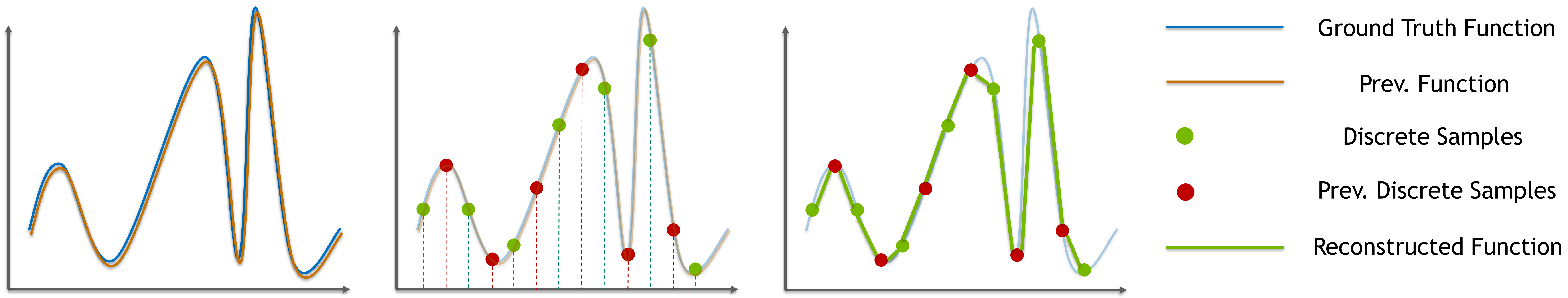
References can be found in <A Survey of Temporal Antialiasing Techniques>, Yang et al.

SPATIAL-TEMPORAL SUPER SAMPLING

Reconstruct high resolution image using samples from across multiple frames

Effective sampling rate drastically increased

Reconstructed image much closer to ground truth

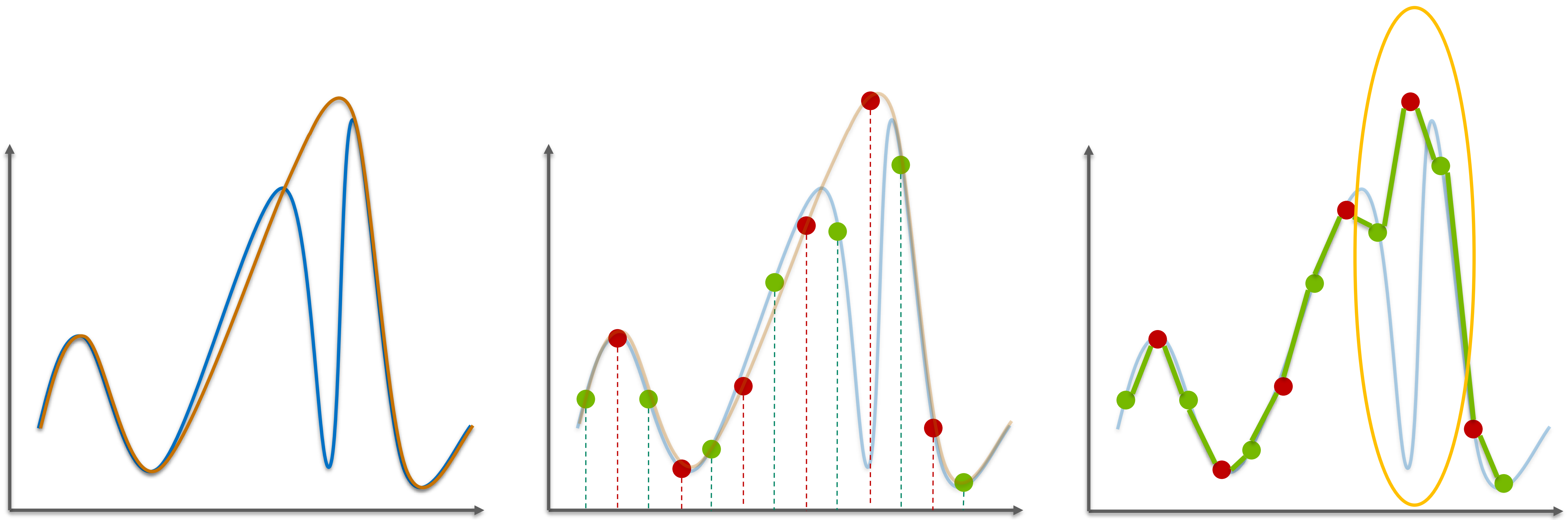


SPATIAL-TEMPORAL SUPER SAMPLING

Devils in the details

Samples from previous frames might no longer be correct due to content changes

Using samples from previous frame naively might lead to artifacts like ghosting



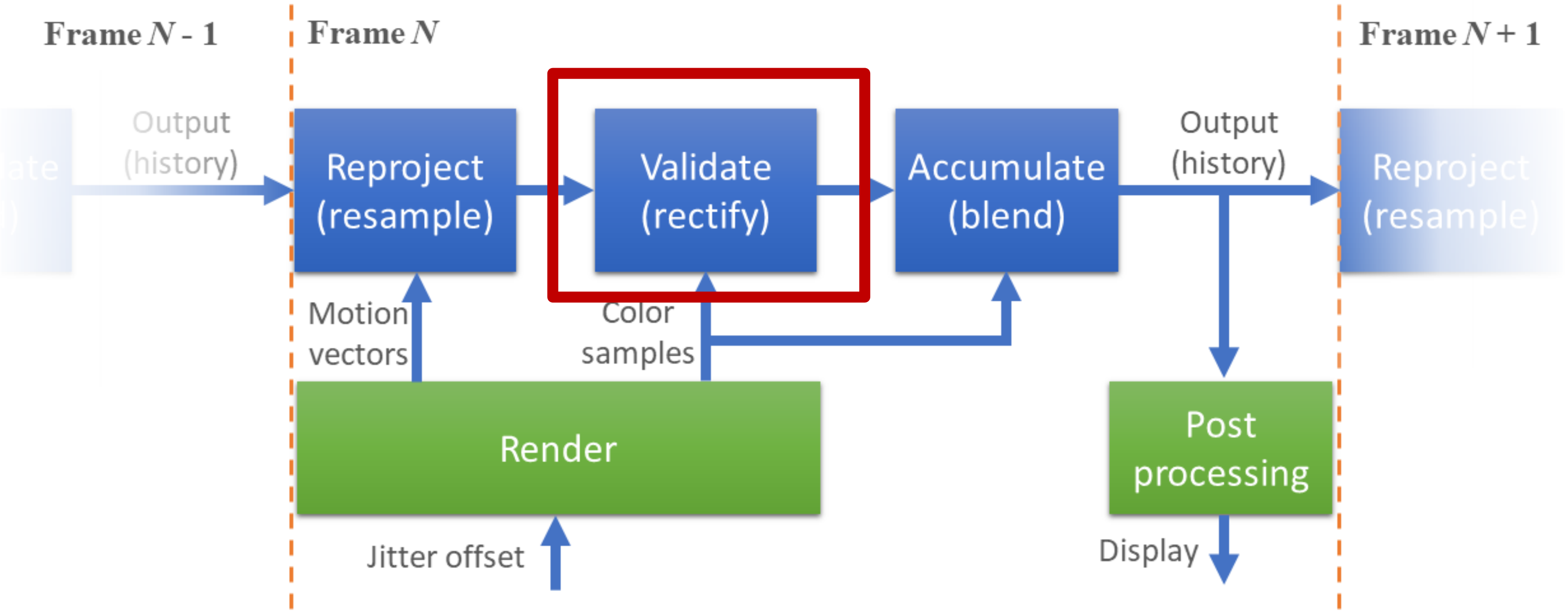
SPATIAL-TEMPORAL UPSAMPLING

History Rectification

Traditional spatial-temporal upsampling algorithms leverages heuristics to rectify invalid samples from previous frames

However common rectification heuristics often trade off between different artifacts:

Blurriness, temporal instability, even moire pattern vs. lagging and ghosting



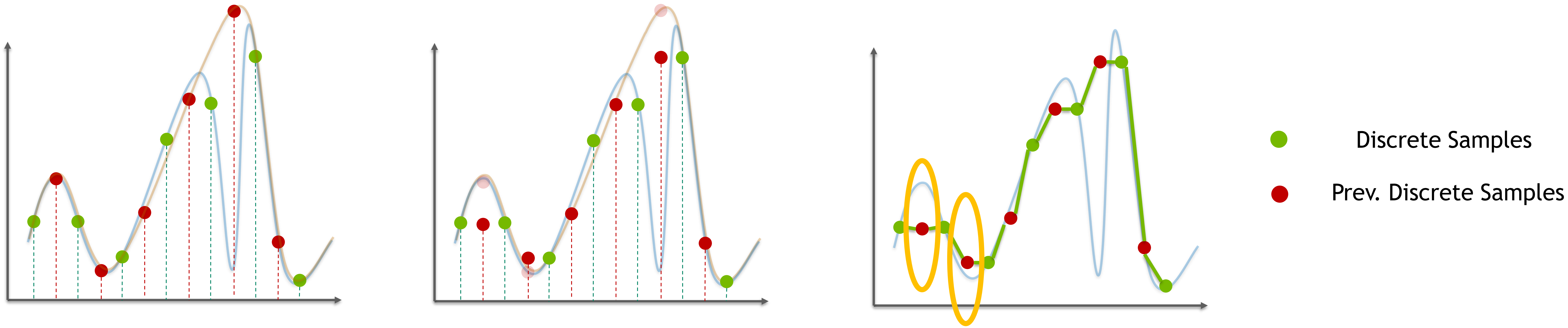
[Yang et al. 2020]

NEIGHBORHOOD CLAMPING

Most commonly used sample rectification technique [Karis14], [Salvi16]

Clamp previous frames samples to the min/max of the neighboring current frame samples

Resulted in loss in details in the reconstructed image



Ghosting Happens without History Rectification

NEIGHBORHOOD CLAMPING

Blurriness/Losing details

1spp Input

Reconstruction with clamping

Reconstruction without clamping

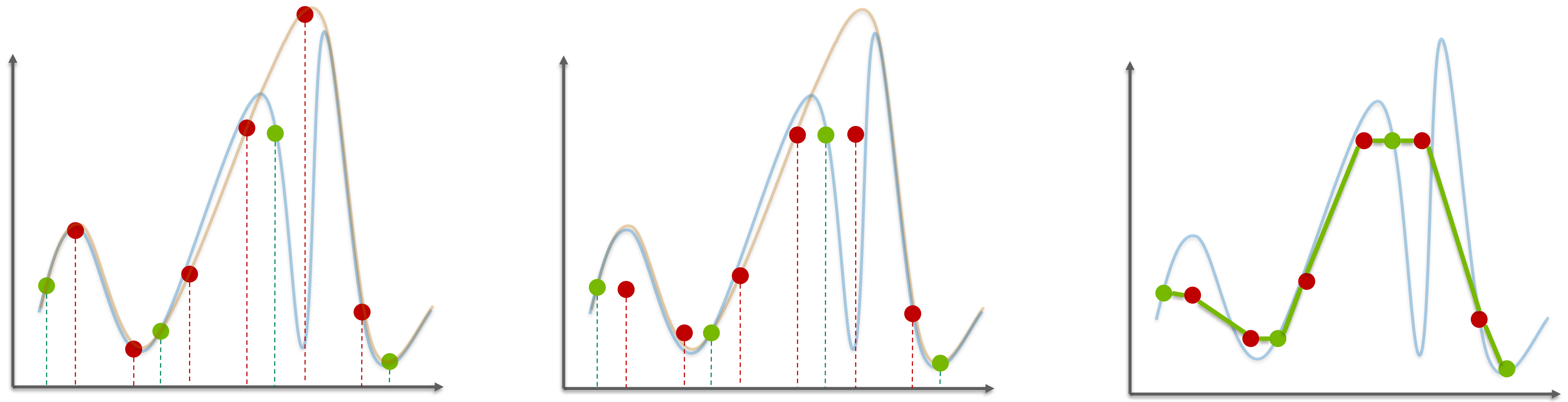
[Yang et al. 2020]

NEIGHBORHOOD CLAMPING

Blurriness/Losing detail

When perform temporal upsampling, clamping introduces more loss in detail

Since bounding boxes are calculated from a low-resolution image




Reconstruction with clamping,
¼ res input
Reconstruction w/o clamping,
¼ res input

Reconstruction with clamping, ¼ res input

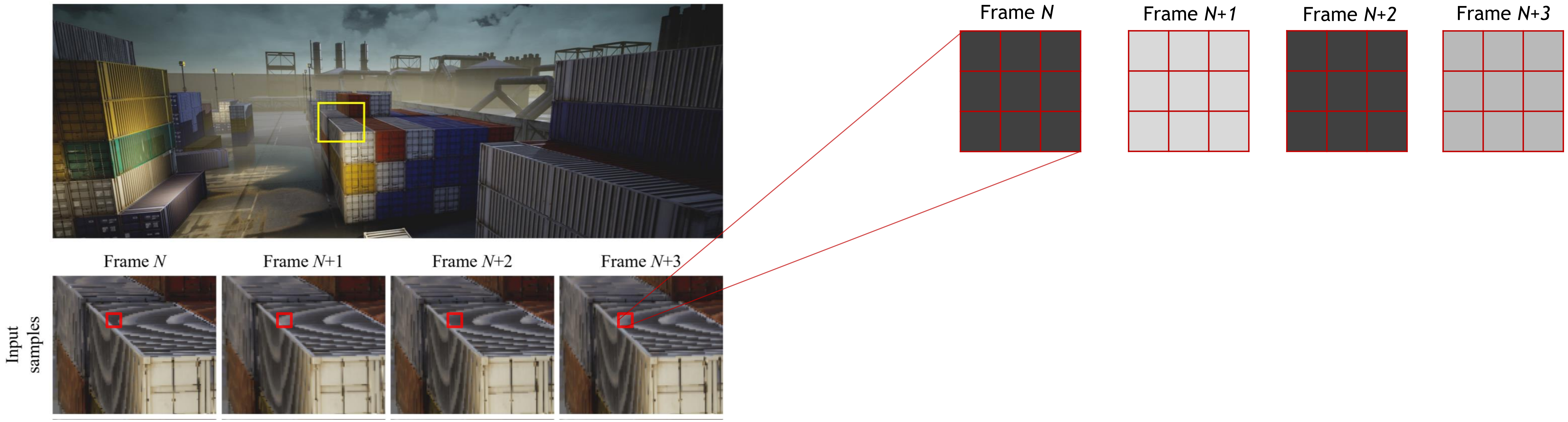
Reconstruction w/o clamping, ¼ res input

Temporal Instability
1080p TAA with clamping

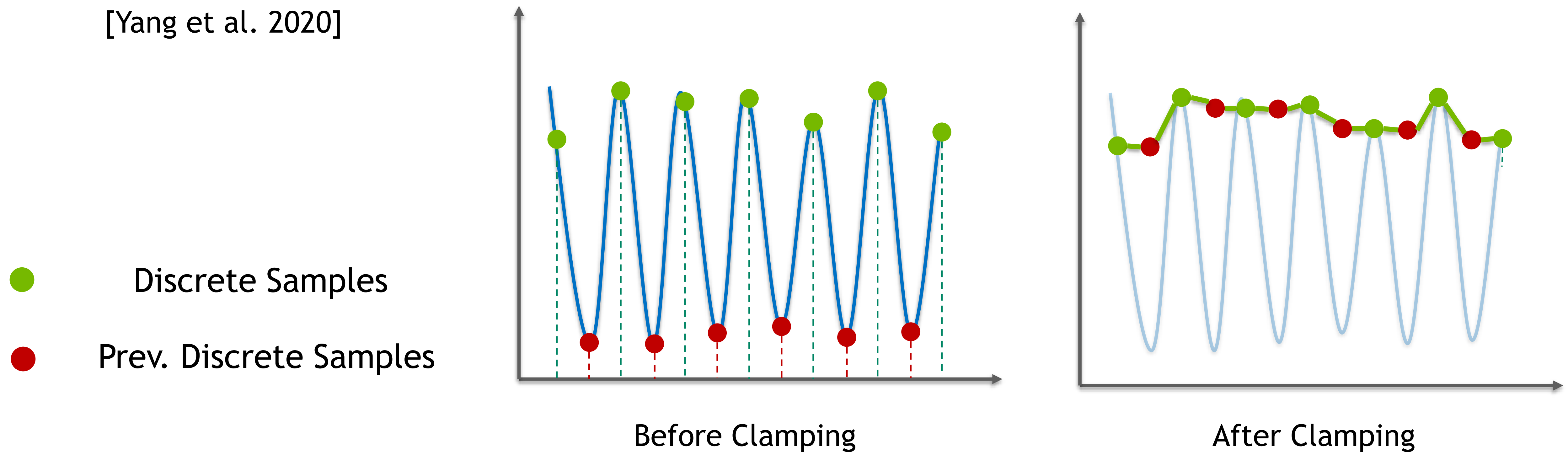
Temporal Instability
540p to 1080p TAAU with Clamping

NEIGHBORHOOD CLAMPING

Temporal Instability and Moire



[Yang et al. 2020]



540p to 1080p TAAU w/o Clamping

REAL-TIME SUPER-RES CHALLENGES

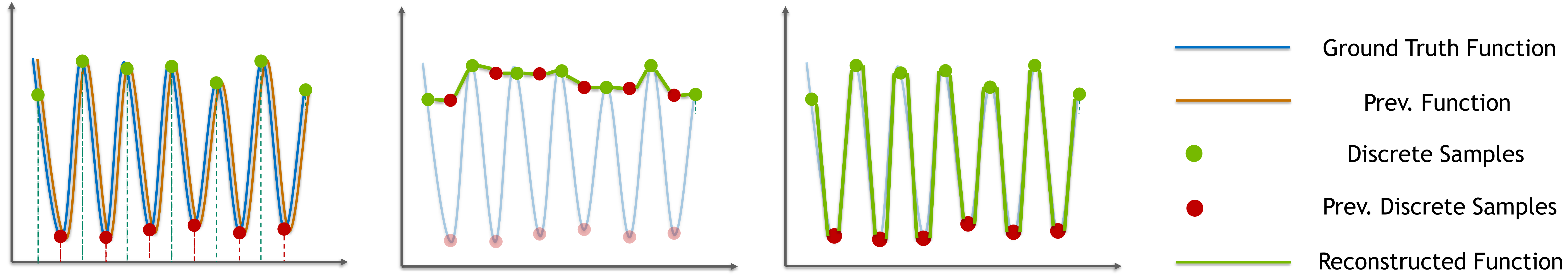
- Single frame approach
    - Blurry image quality
    - Inconsistent with native rendering
    - Temporally unstable
- Multi-frame approach
    - Heuristics to detect and rectifies changes across frames
    - Limitation in heuristics causing blurriness, temporal instability and ghosting

DLSS 2.0: DL BASED MULTI-FRAME RECONSTRUCTION

DLSS uses a neural network trained from tens of thousands of high-quality images

Neural networks are much more powerful than handcrafted heuristics

Much higher quality reconstructions using samples from multiple frames



Multi-frame samples and GT function

Non-DL reconstruction

DLSS reconstruction

DLSS 2.0
540p to 1080p


1080p TAA


540p to 1080p DLSS2.0

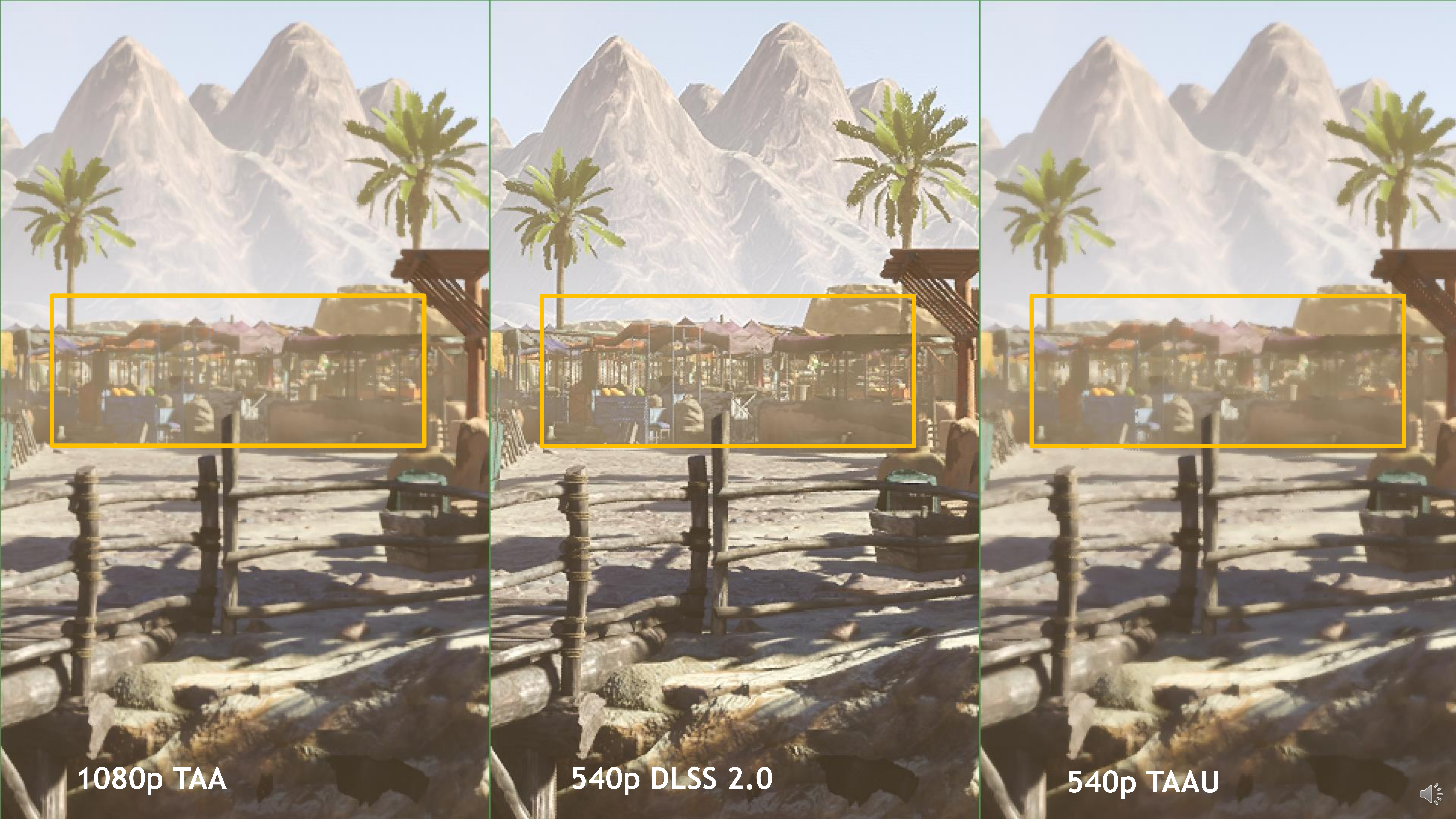

540p to 1080p TAAU


1080p TAA
540p DLSS 2.0
540p TAAU

1080p TAA


540p to 1080p DLSS2.0

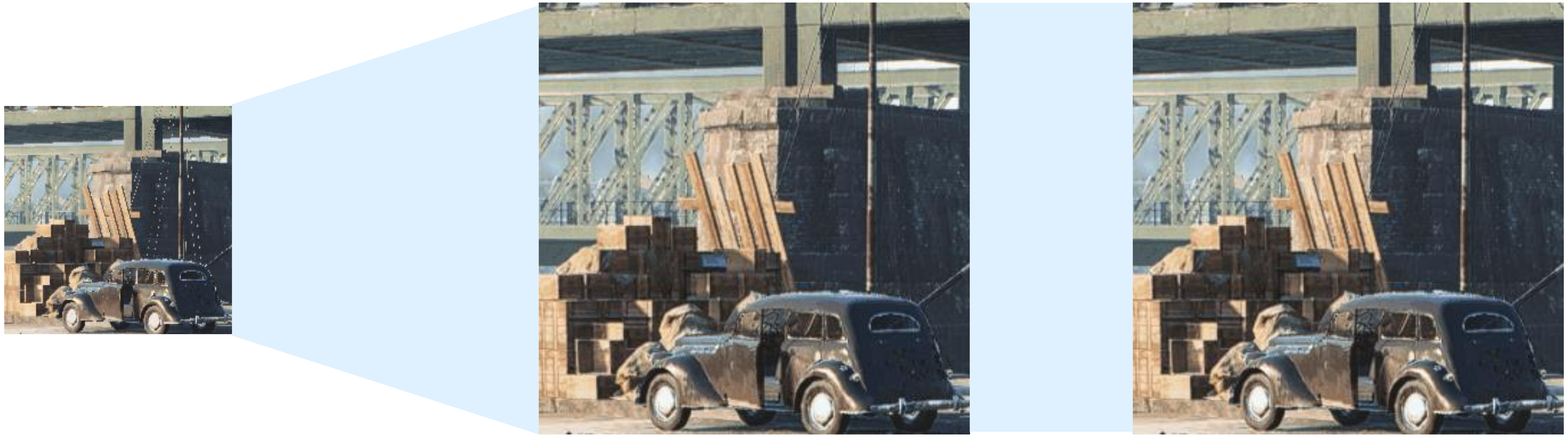

540p to 1080p TAAU


1080p TAA
540p DLSS 2.0
540p TAAU

ENGINE INTEGRATION

ENGINE INTEGRATION

Pipeline of DLSS 2.0



Input

DL Anti-Aliasing and Super Sampling

Post Process

Geometry/Shading
1080p

→

DL Upsampling
4K

→

Post (MB, Bloom, Tone map, …)

ENGINE INTEGRATION

Renderer adjustments

Input

DL Anti-Aliasing and Super Sampling

Post Process

Geometry/Shading 1080p → DL Upsampling 4K → Post (MB, Bloom, Tone map, …)

ENGINE INTEGRATION

Viewport Jitters

Viewport jittered similar to TAA

Halton sequence is recommended as jitter pattern

Number of phases should be linearly scaled with the area of low-resolution pixels

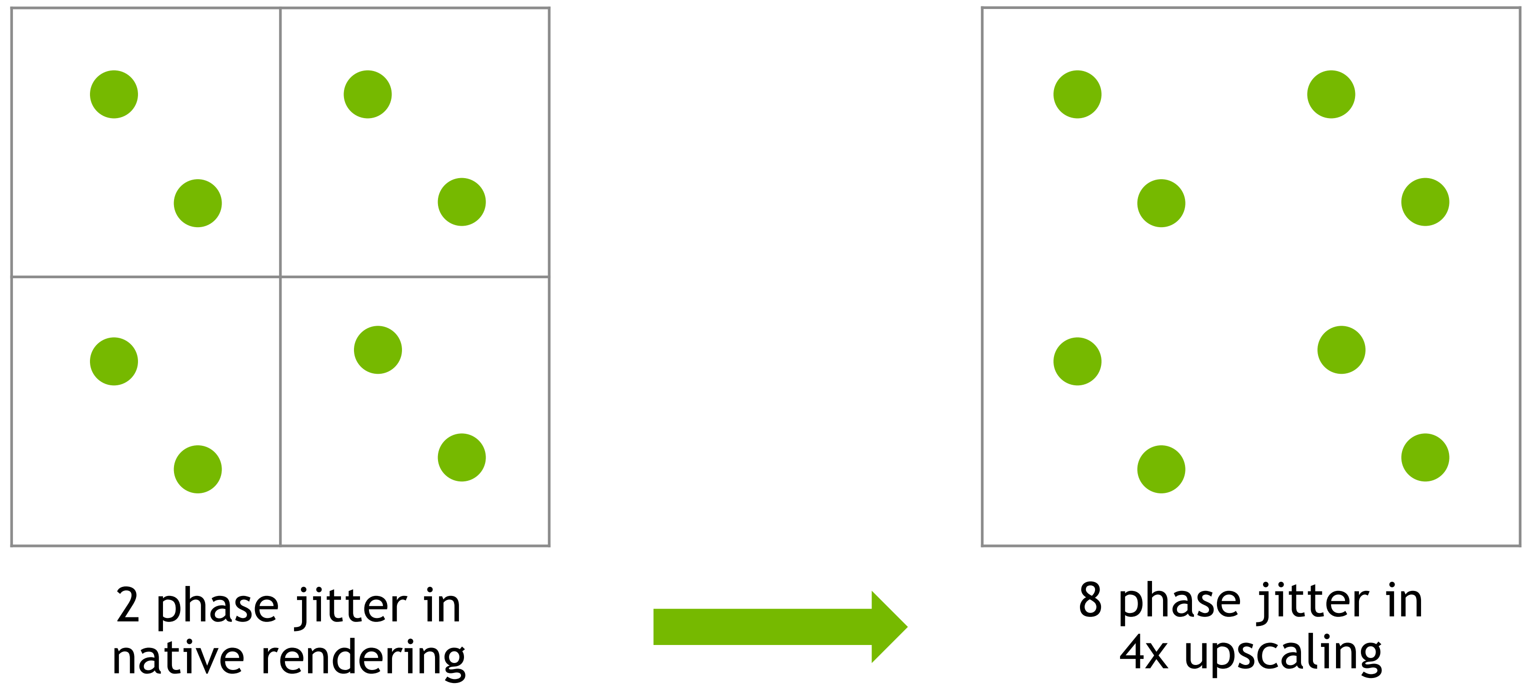
ENGINE INTEGRATION

Viewport Jitters

Viewport jittered similar to TAA

Halton sequence is recommended as jitter pattern

Number of phases should be linearly scaled with the area of low-resolution pixels



ENGINE INTEGRATION

Texture Mip Bias Adjustment

DLSS is not designed to enhance texture resolution

Mip bias should be set so textures have the same resolution as native rendering

- Mip bias = Native bias + Log2(Render Res/Native Res)

ENGINE INTEGRATION

Stochastic/Dithered Effects Adjustments

Effects requiring temporal dithering will need to increase the phase of dithering

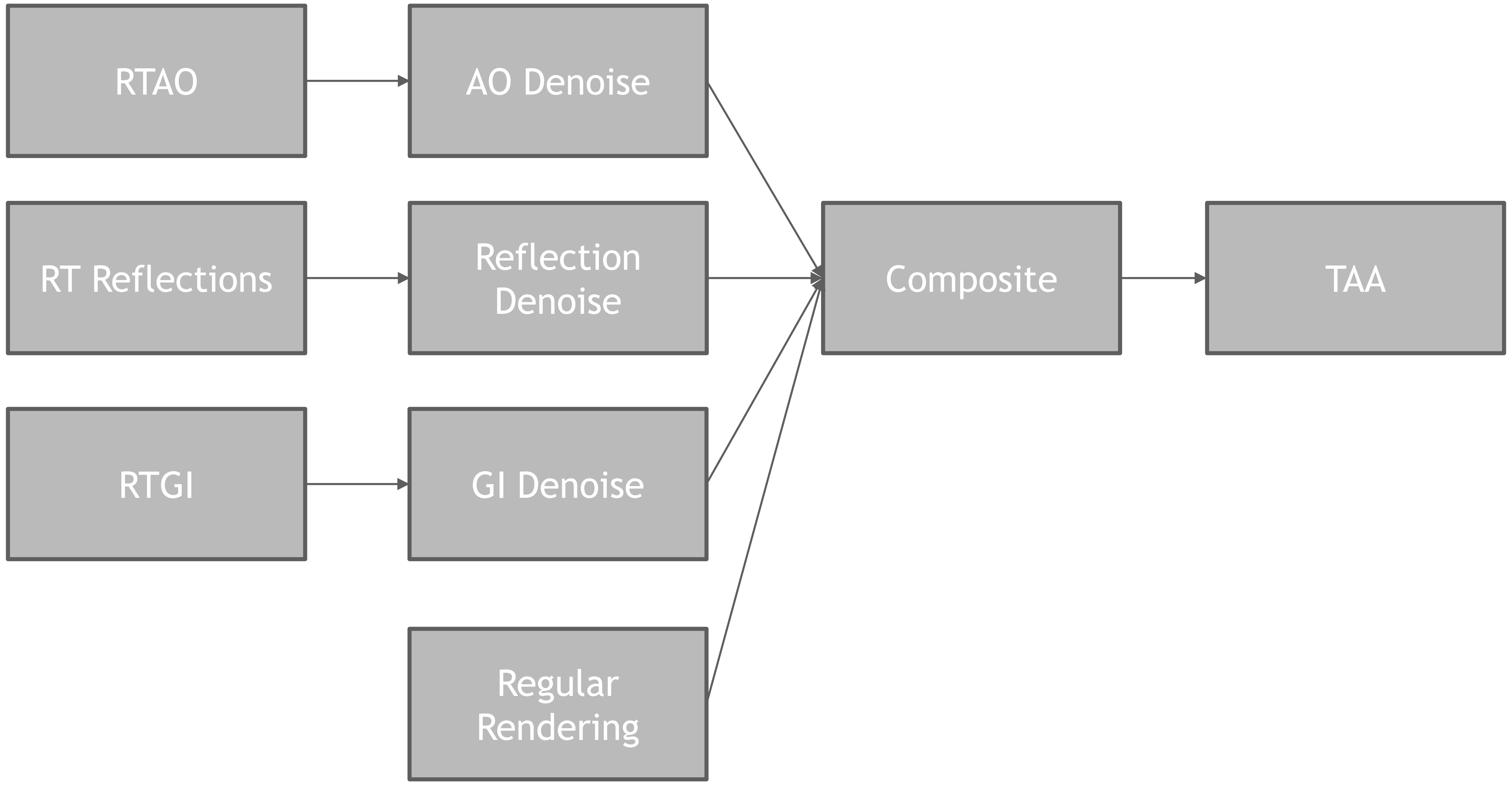
- Similar to viewport jitters
- Common effects include SSAO, SSR etc.

ENGINE INTEGRATION

Denoising for Ray Traced Effects

Since TAA is replaced by DLSS

Denoisers relying on TAA will need to improve denoisers or add a dedicated TAA pass after denoising

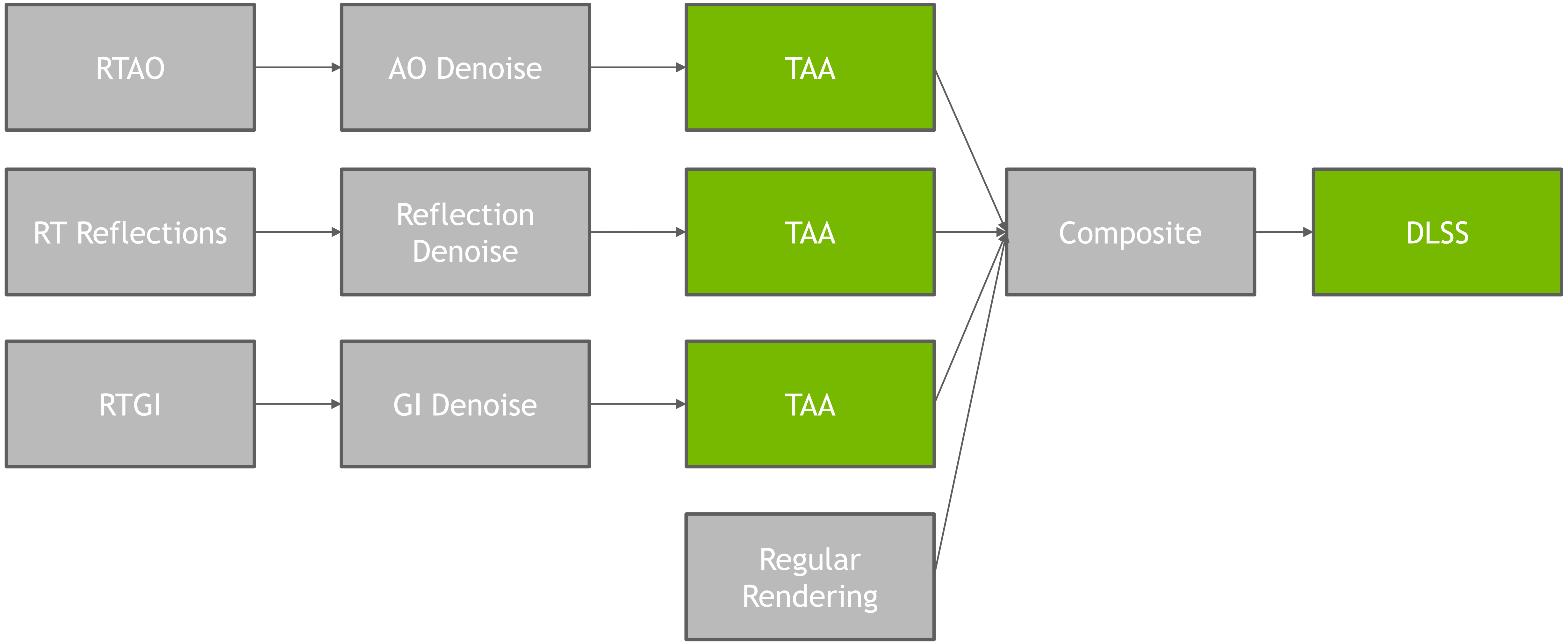


ENGINE INTEGRATION

Denoising for Ray Traced Effects

Since TAA is replaced by DLSS

Denoisers relying on TAA will need to improve denoisers or add a dedicated TAA pass after denoising



ENGINE INTEGRATION

Inserting DLSS into the pipeline

Input

DL Anti-Aliasing and Super Sampling

Post Process

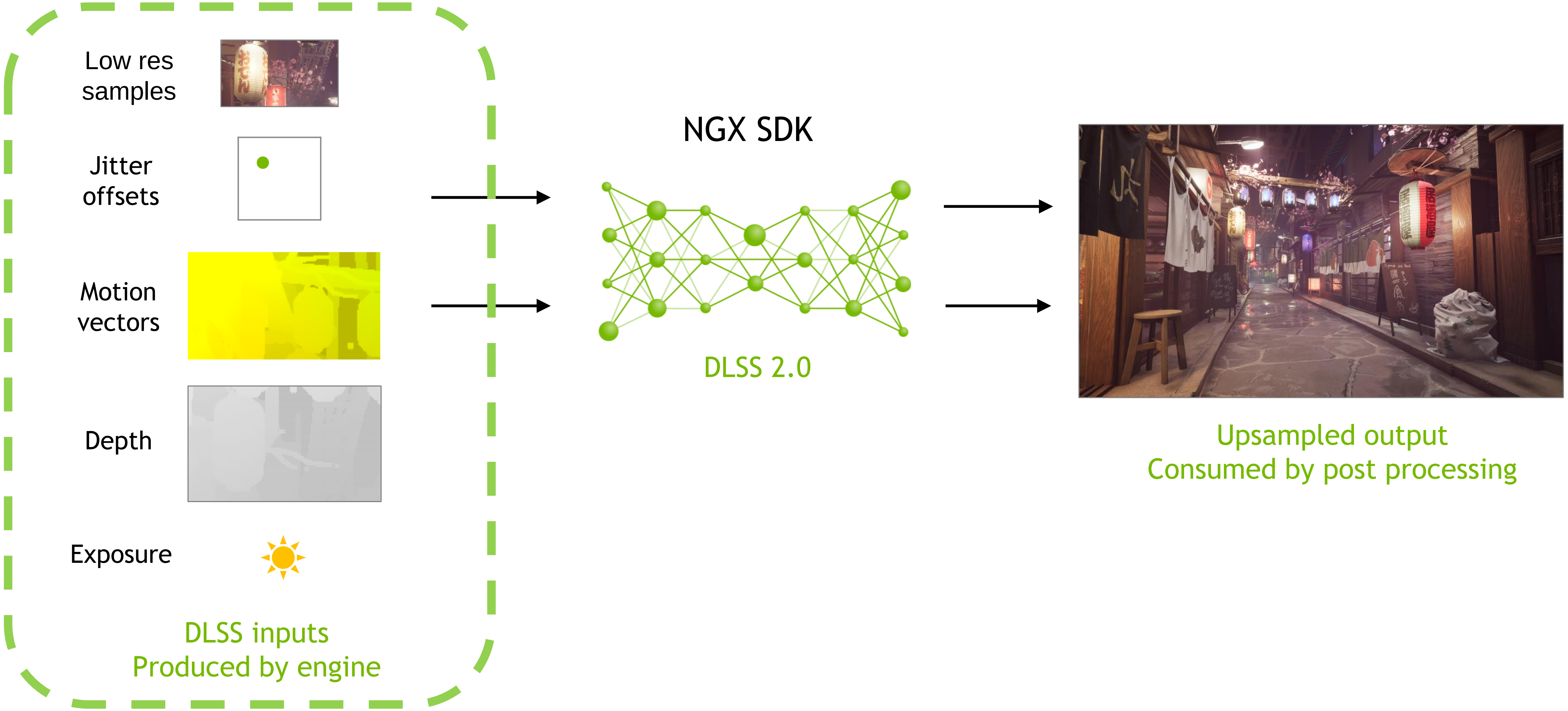
Geometry/Shading
1080p

DL Upsampling
4K

Post (MB, Bloom, Tone map, …)

ENGINE INTEGRATION

DLSS Inputs and Outputs



ENGINE INTEGRATION

Post Processing Adjustments

Input

DL Anti-Aliasing and Super Sampling

Post Process

Geometry/Shading
1080p

DL Upsampling
4K

Post (MB, Bloom, Tone map, …)

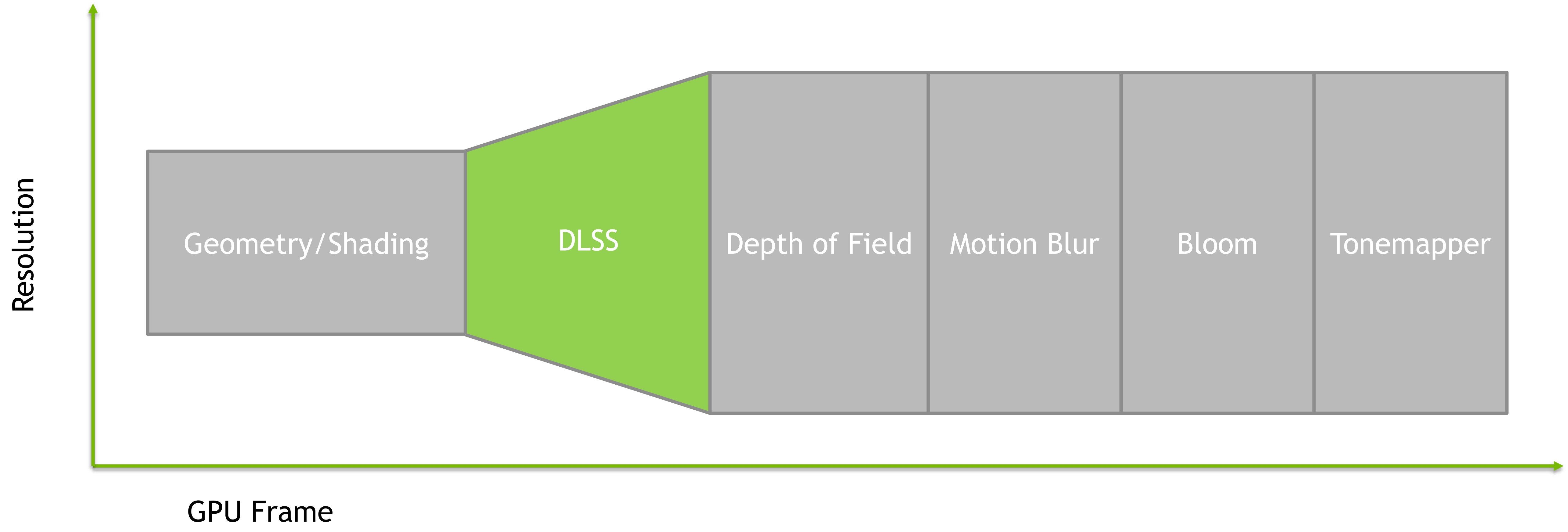
NVIDIA

ENGINE INTEGRATION

High Resolution Post Processing

Post processing rendered at upscaled resolution

Engine needs to handle post processing resolution being different from geometry and shading



UE4 DLSS 2.0 BRANCH

AVAILABLE THROUGH DLSS DEVELOPER PROGRAM

A common branch for DLSS2.0 and ray tracing for developers to experiment with.

https://developer.nvidia.com/dlss

Thank you!

: @edliu1105

知 : @文刀秋二

NVIDIA