

How to Decimate your textures

BCn compression tricks in
Horizon Forbidden West

hugh.malan@guerrilla-games.com



 Graphics Programming Conference, November 18-20, Breda

2025

HORIZON
FORBIDDEN WEST



NUBIS³

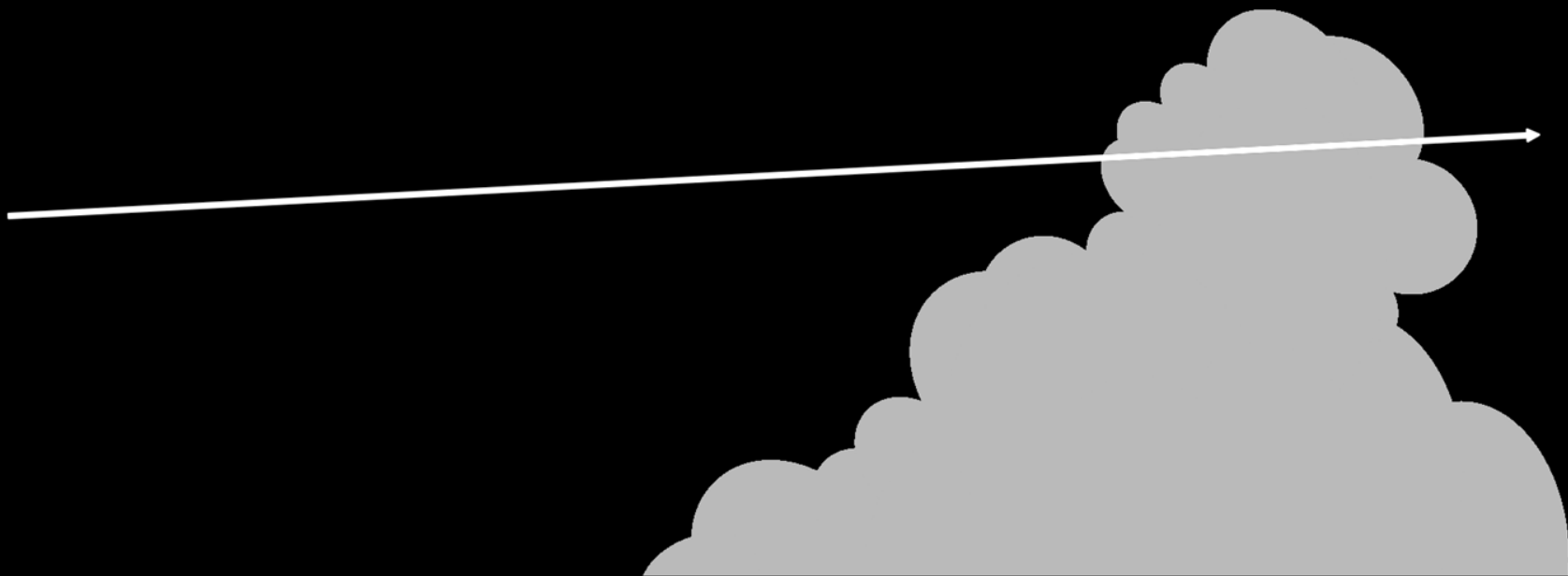
Methods (and madness) to model and render
immersive real-time voxel-based clouds.

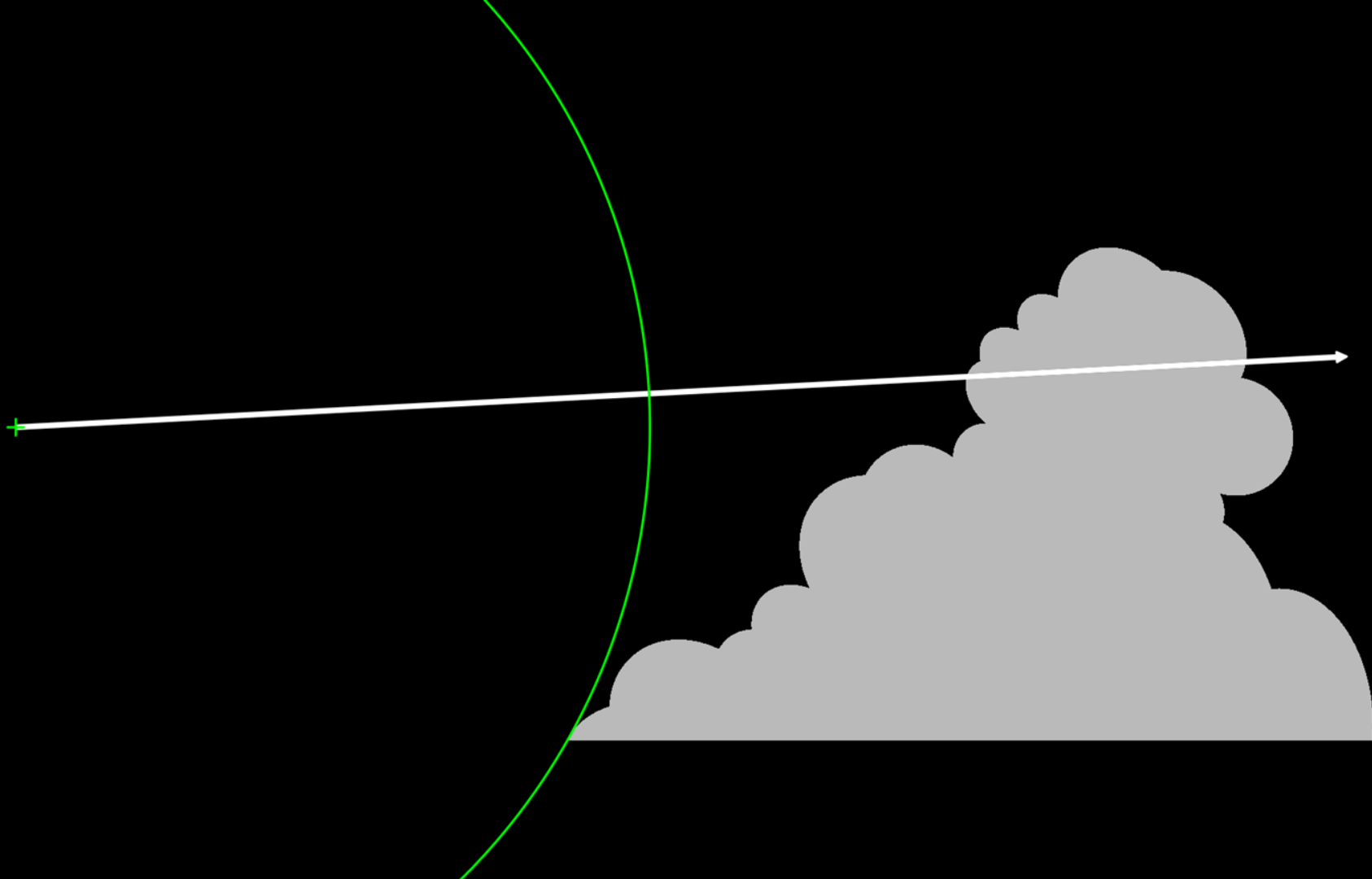
Andrew Schneider / Atmospherics Lead / Guerrilla

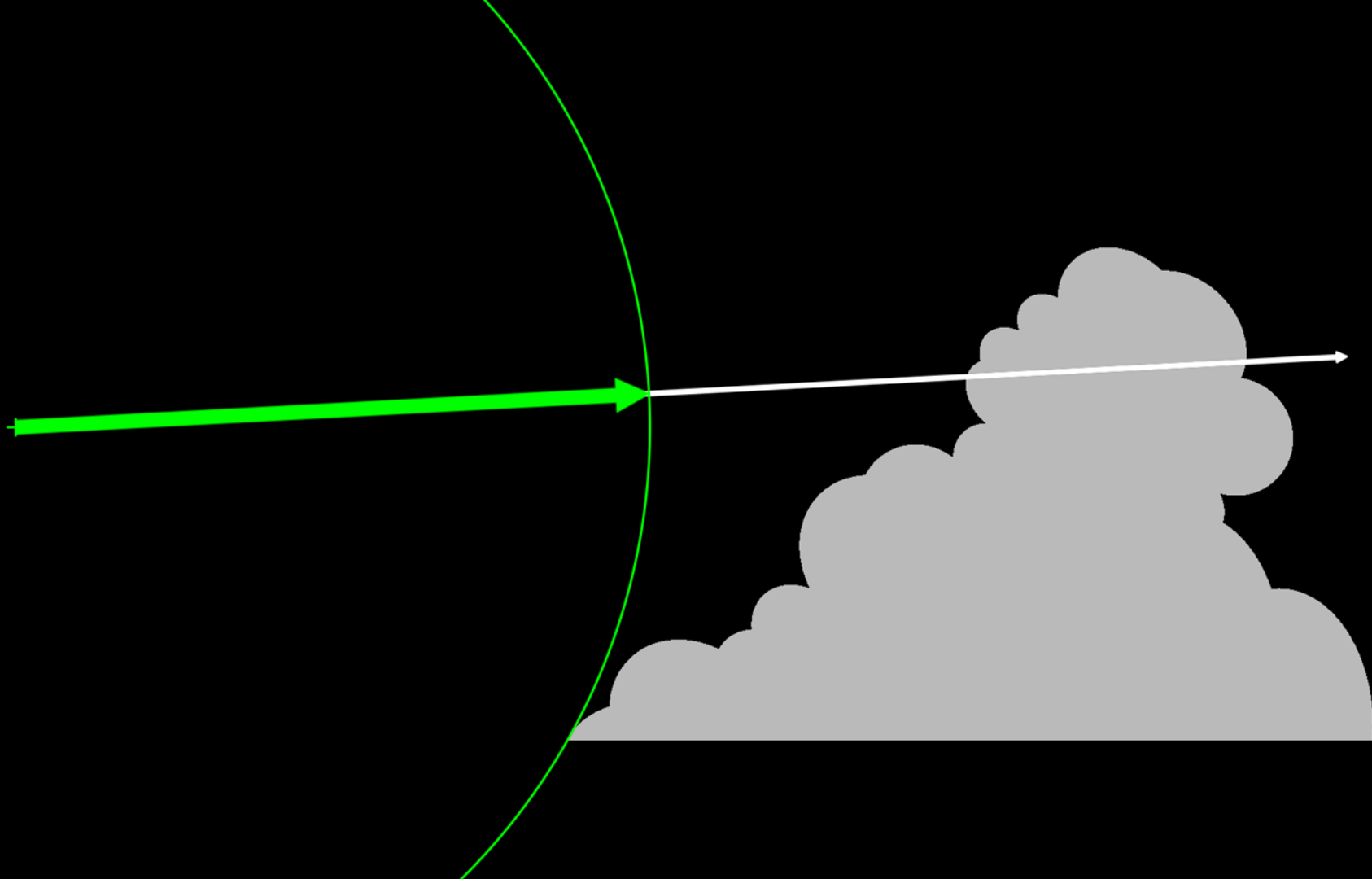
Advances in Real-Time Rendering in Games Course

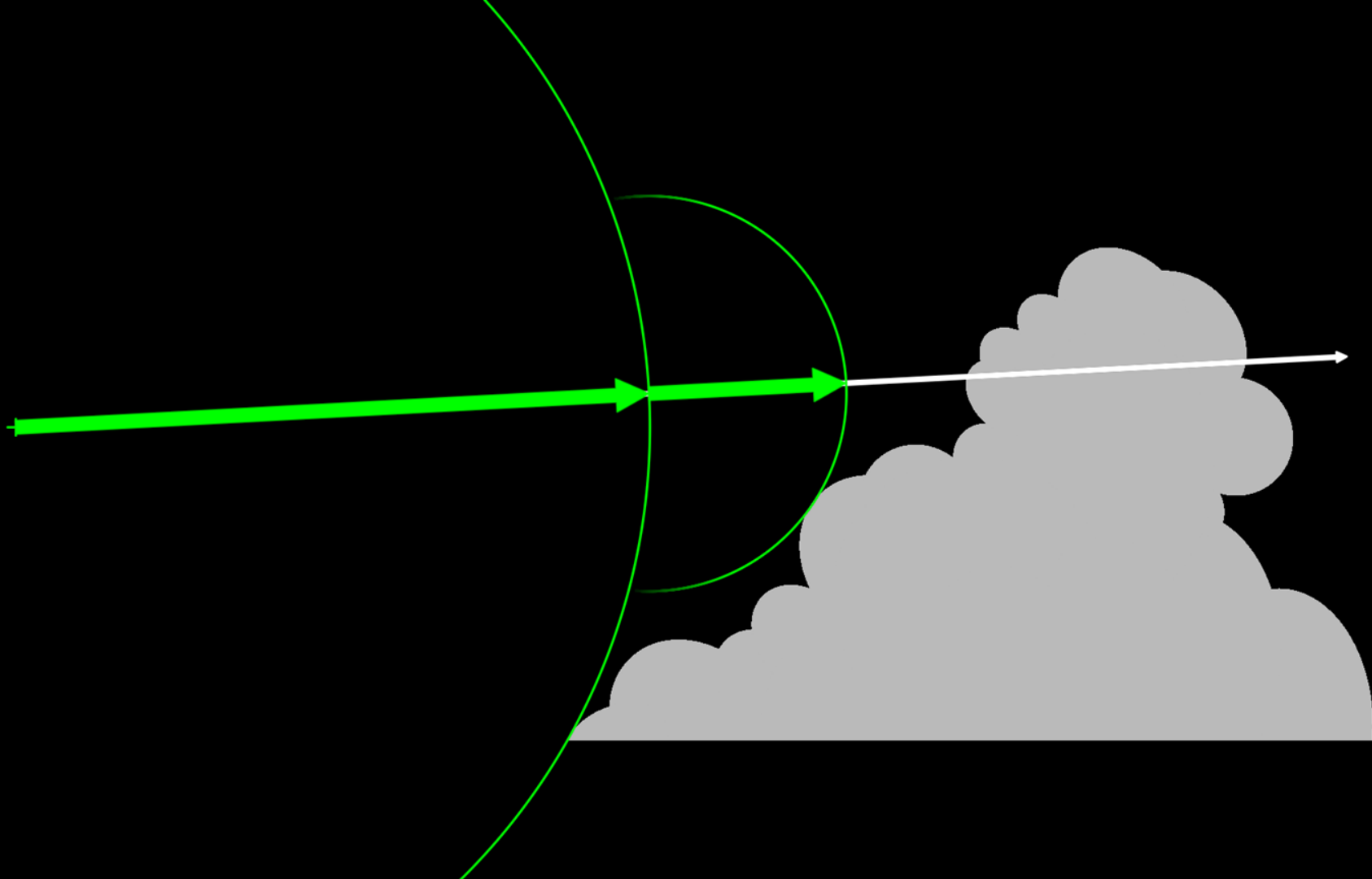


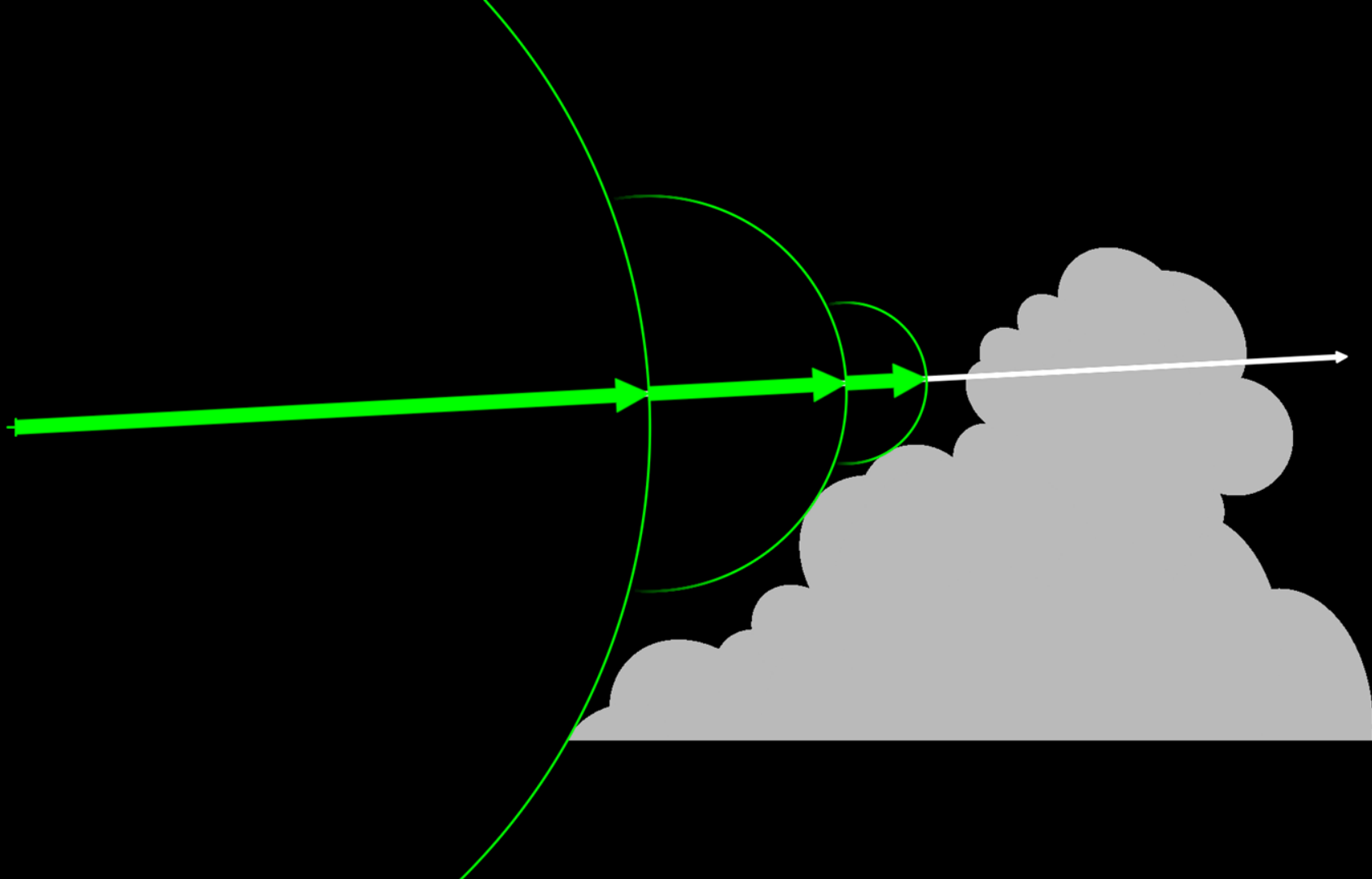
Decima Engine, 2023

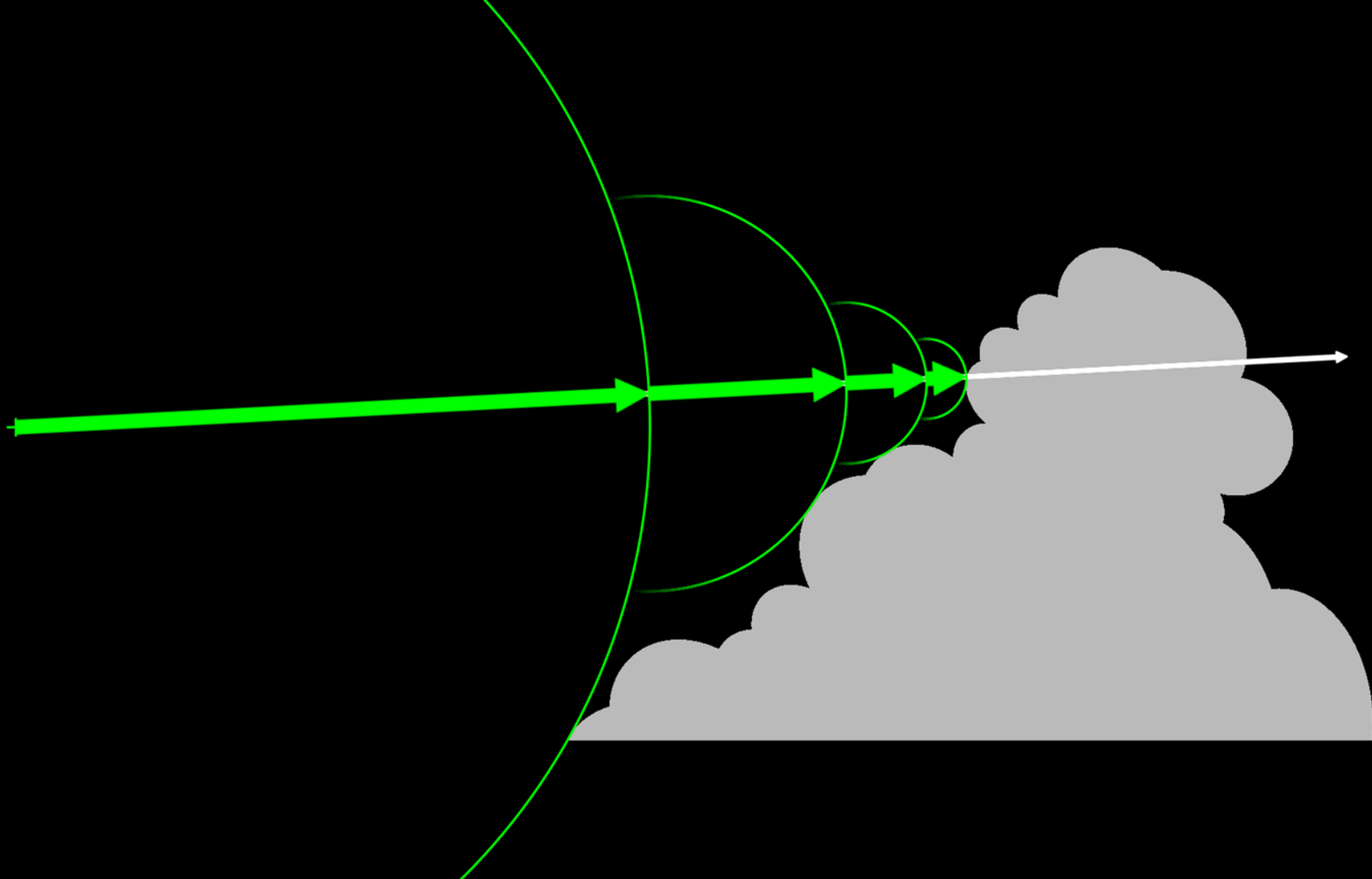




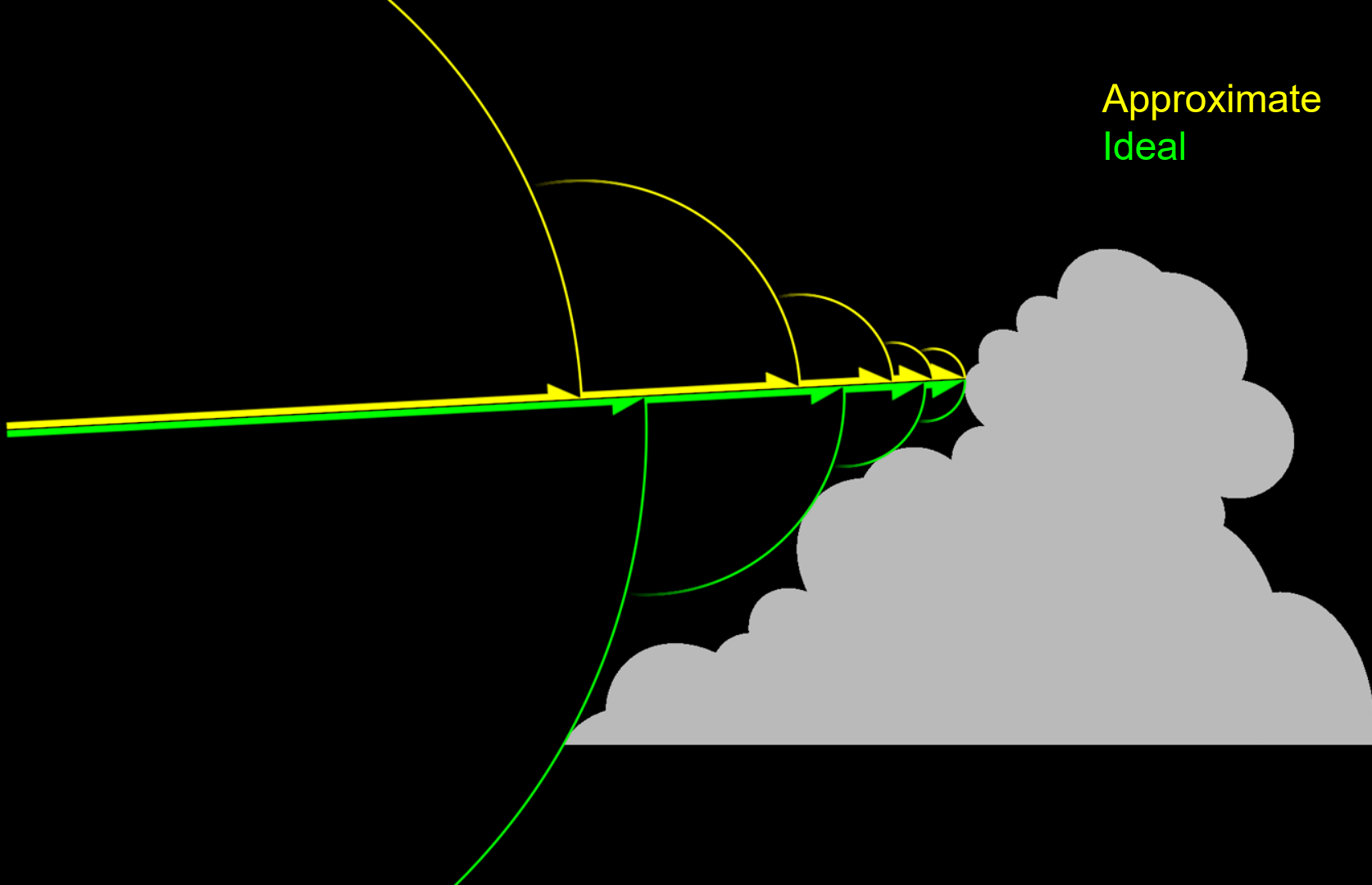


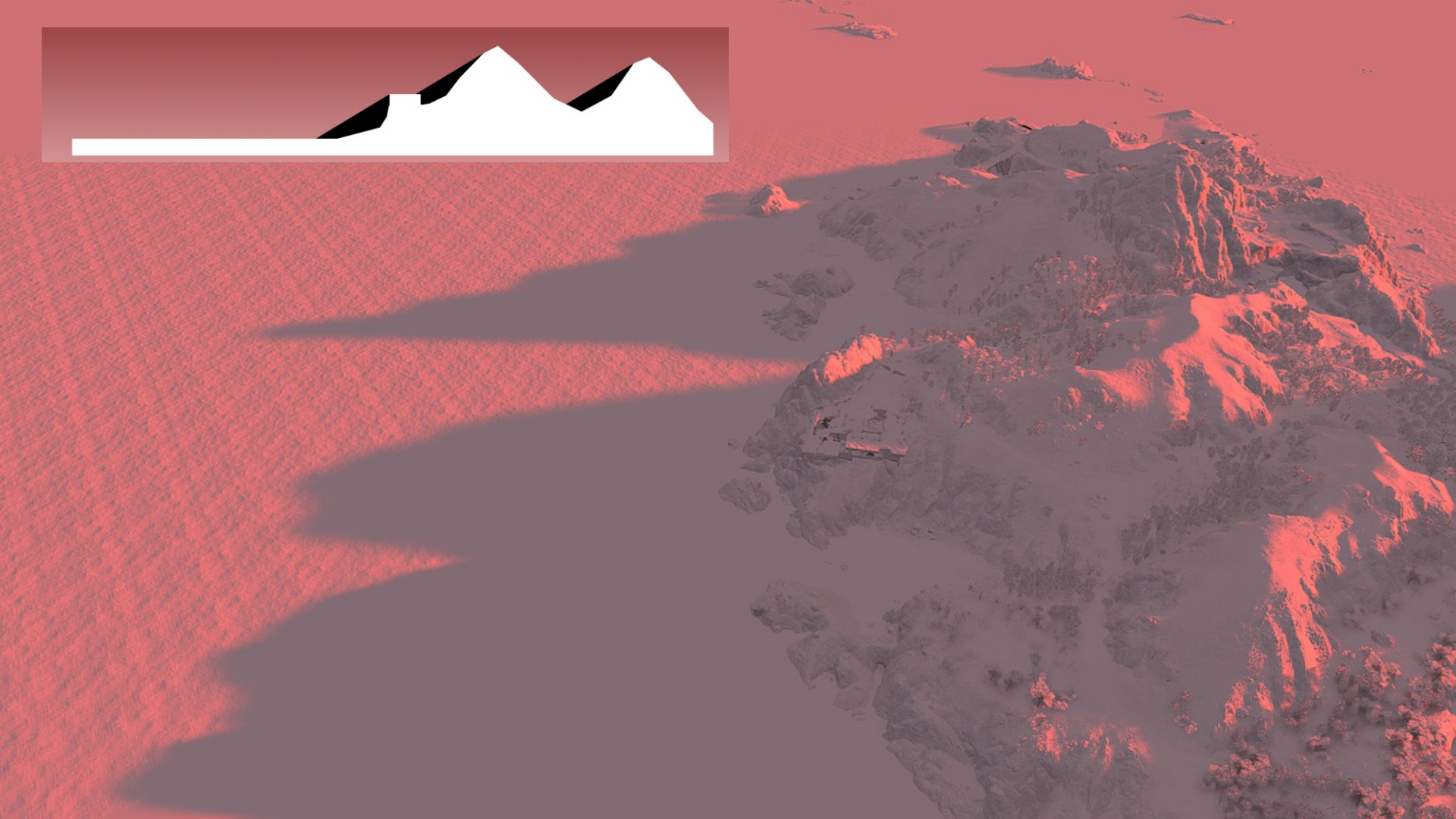




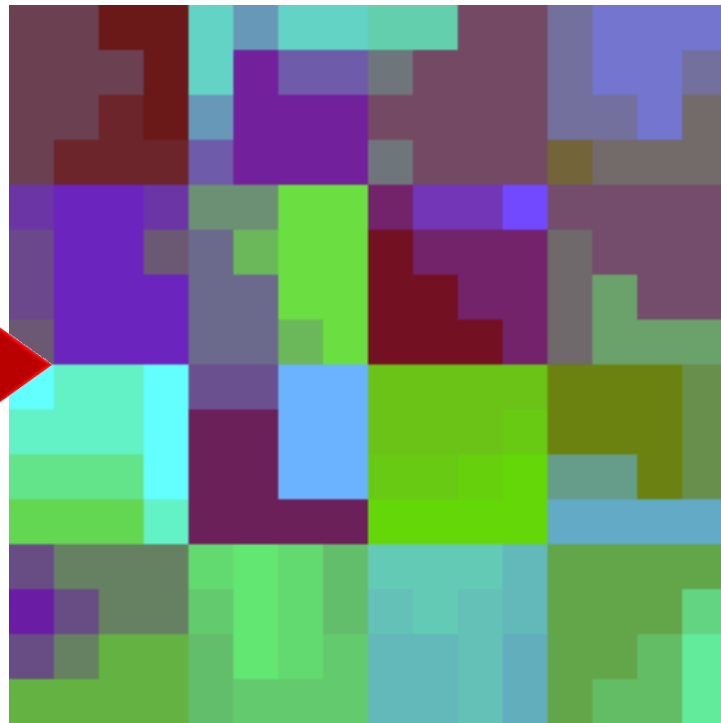
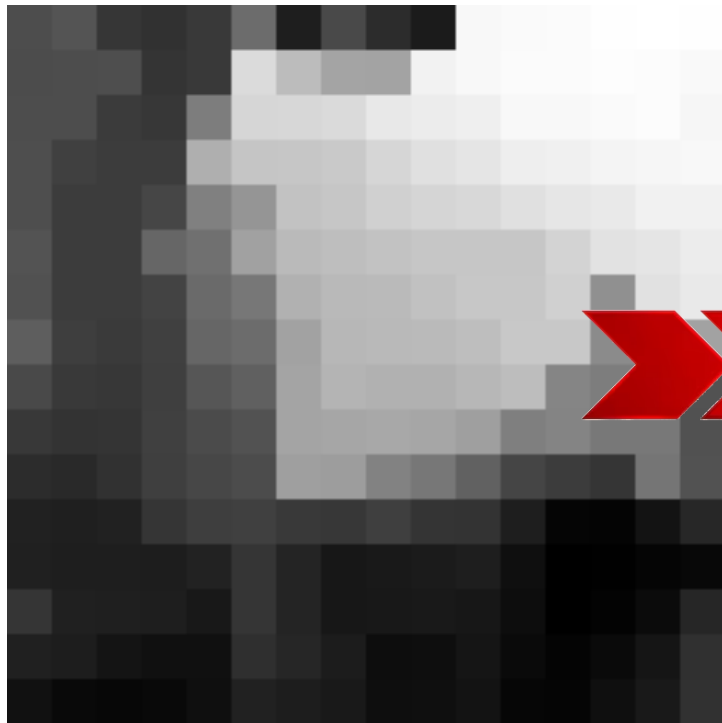


Approximate
Ideal





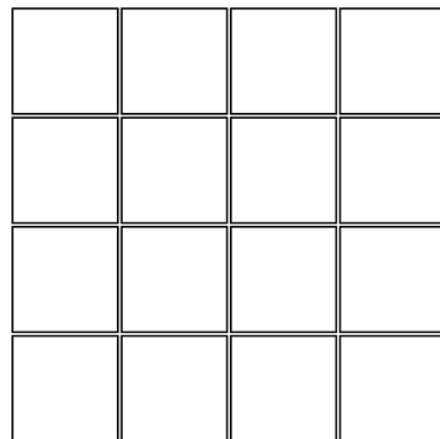
16bpp input $\geq$ 4bpp BC1 output



- Requirements
- BC1 overview
- GPU sampling
- RGB to scalar
- Encoder
- Results



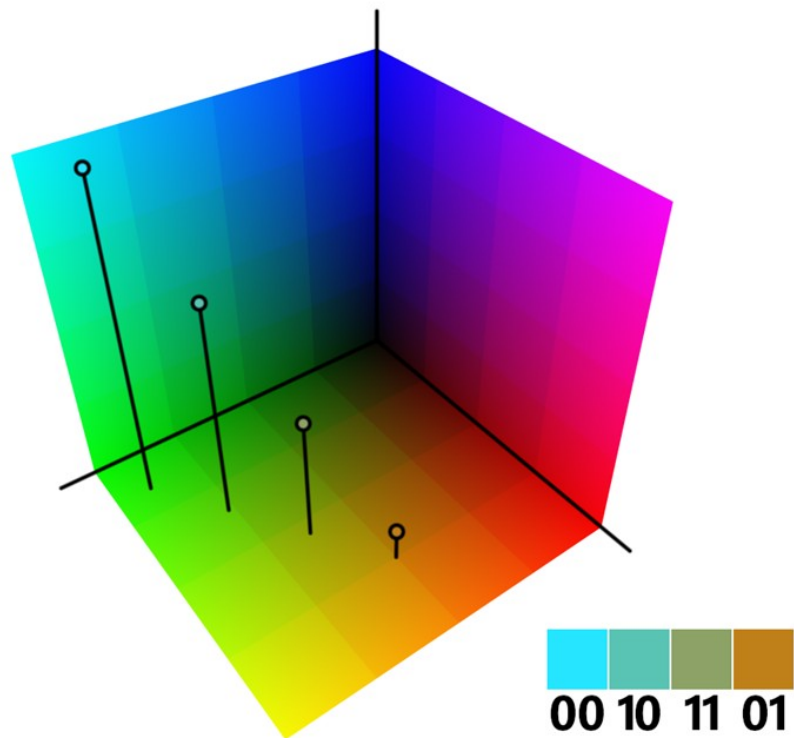
0	0	1	0	0	1	1	1	0	0	0	1	1	1	1	1	0	1	1	1	0	1	1	1	1	0	0	0	1	1	0	0	0	0	0	1	0	0	0	1	0	1	0	1	0	1	1	1	0	1	1	1	1	1	0	1	1	0	1	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

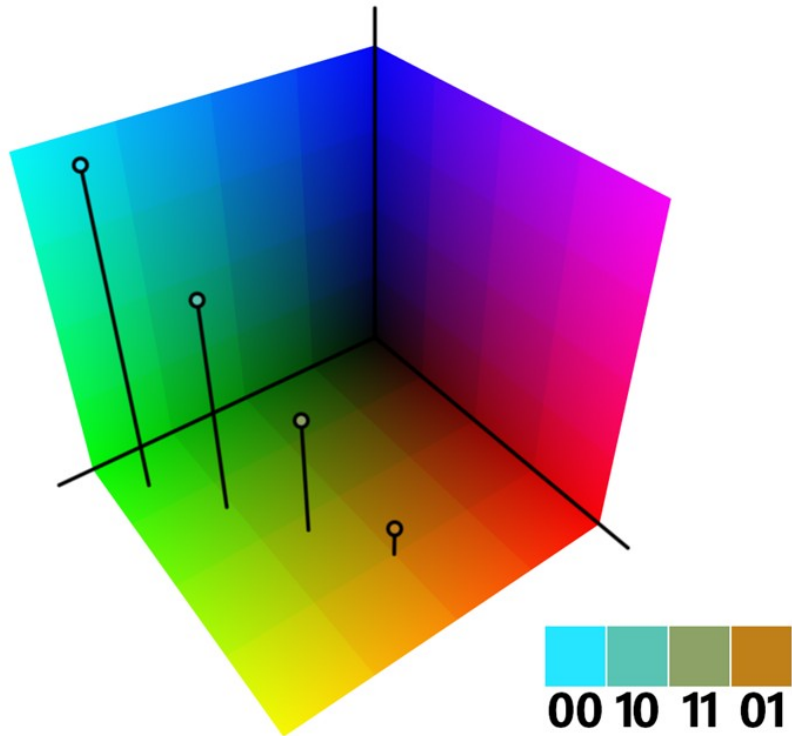


0010011110001111110111101111100011000000100010101111011111110110101

R R R R G G G G G B B B B B

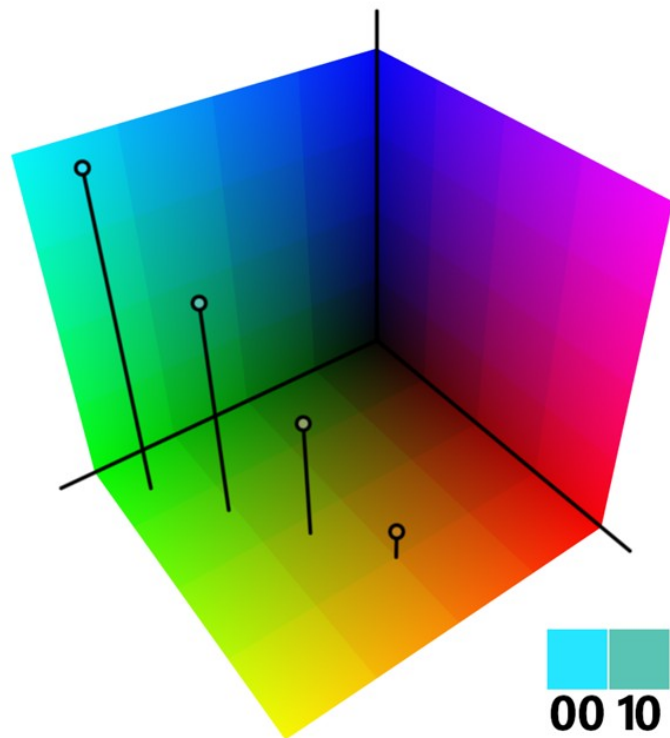
[illegible]

[illegible]

[illegible]

00	00	00	10
00	10	10	11
10	11	11	11
10	11	01	01

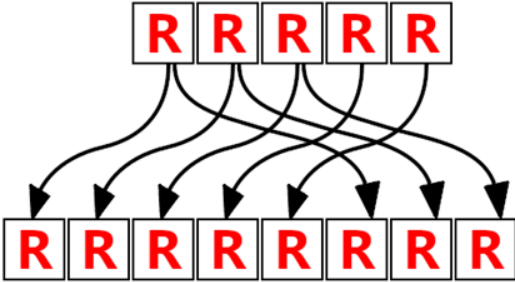
0	0	1	0	0	1	1	1	0	0	0	1	1	1	1	1	1	0	1	1	1	0	1	1	1	1	1	0	0	0	1	1	0	0	0	0	0	1	0	0	0	1	0	1	1	1	0	1	1	1	1	1	0	1	1	0	1							
R	R	R	R	R	G	G	G	G	G	B	B	B	B	B	R	R	R	R	R	G	G	G	G	G	B	B	B	B																																			



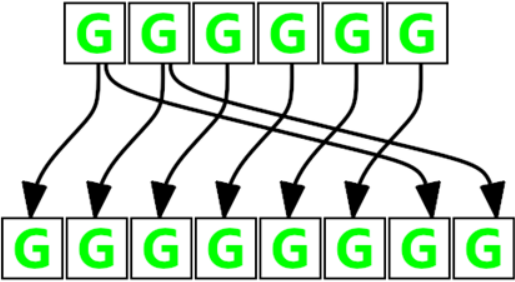
00	00	00	10
00	10	10	11
10	11	11	11
10	11	01	01

GPU sampling of endpoint colors

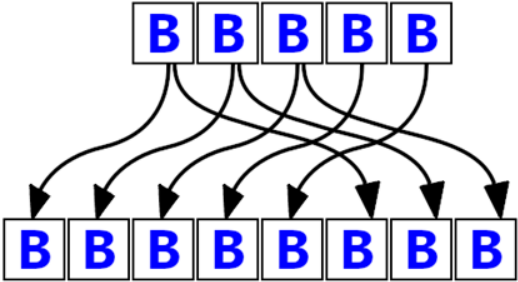
5-6-5 endpoint color



$r < 3 \mid r > 2$



$g < 2 \mid g > 4$



$b < 3 \mid b > 2$

Expanded to 8-8-8

GPU sampling of interpolated colors on AMD

```
if (interpolation_mode == FourInterpolatedColors)
{
    outColors[0] = color_0; // color_0 and color_1 are RGB888
    outColors[1] = color_1;
    outColors[2] = ((color_0 * 43) + (color_1 * 21) + IVec3(32, 32, 32)) / 64;
    outColors[3] = ((color_0 * 21) + (color_1 * 43) + IVec3(32, 32, 32)) / 64;
}
else // ThreeInterpolatedColorsAndTransparentBlack
{ ... }
```



RGB to scalar requirements

```
float sampled_value = dot(  
    EncodedBC1Texture.Sample(uv).rgb,  
    unpack_constant.xyz  
);
```

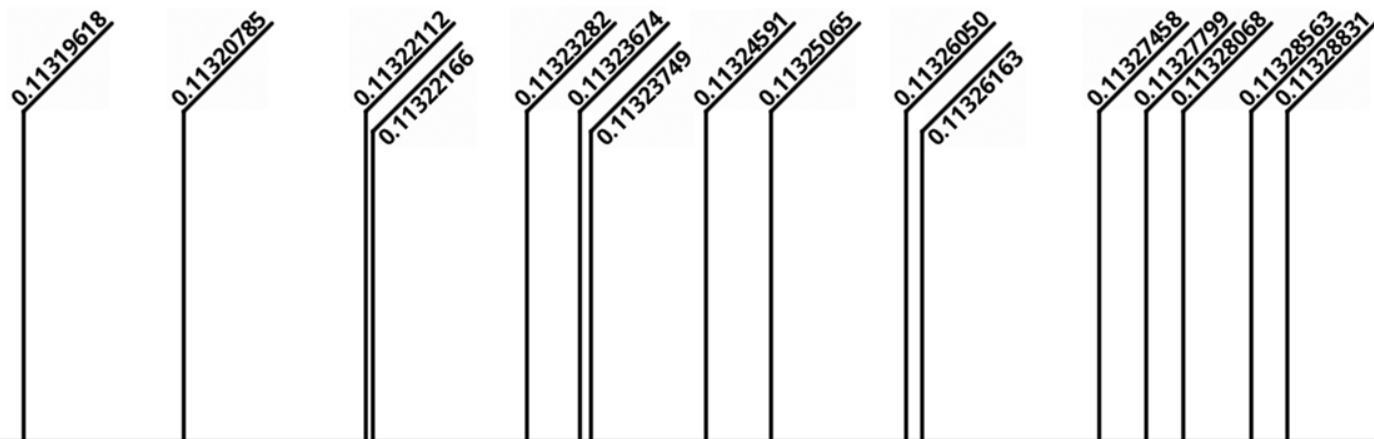


RGB to scalar dotproduct constant

```
float sampled_value = dot(  
    EncodedBC1Texture.Sample(uv).rgb,  
    float3(0.96414679, 0.03518212, 0.00067109) // 0.002% max error  
);
```

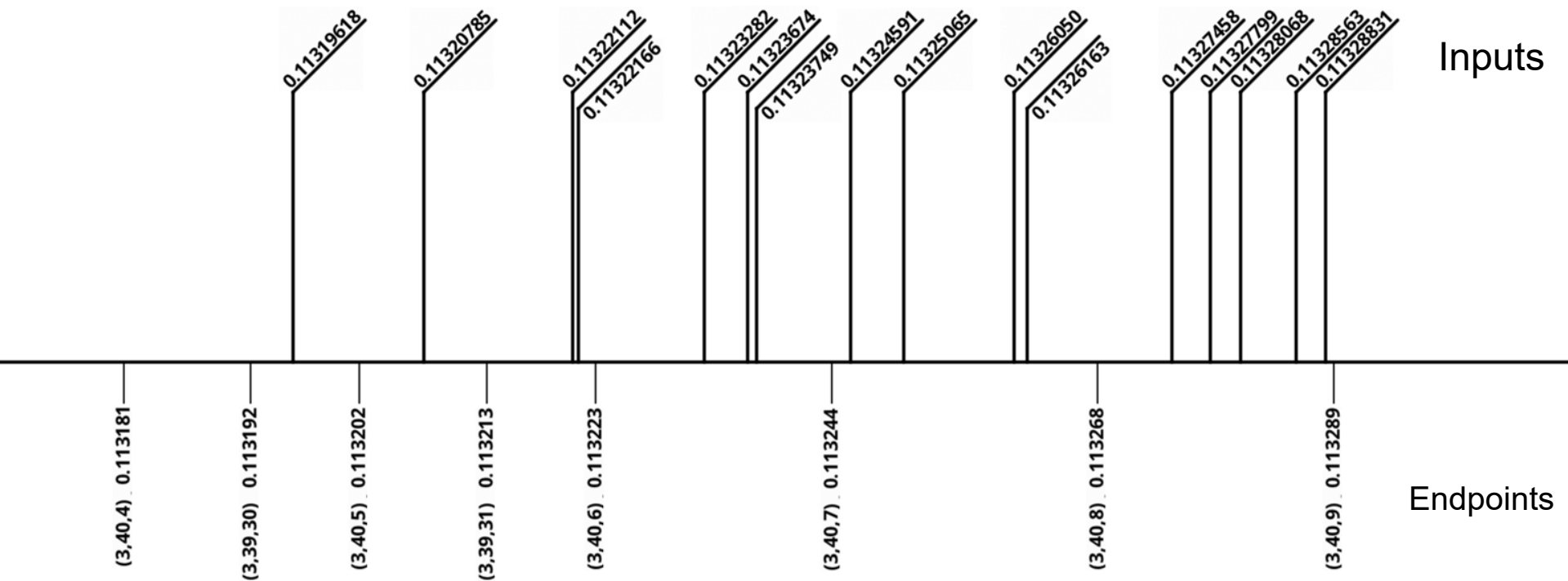


Encoder - 4x4 input values

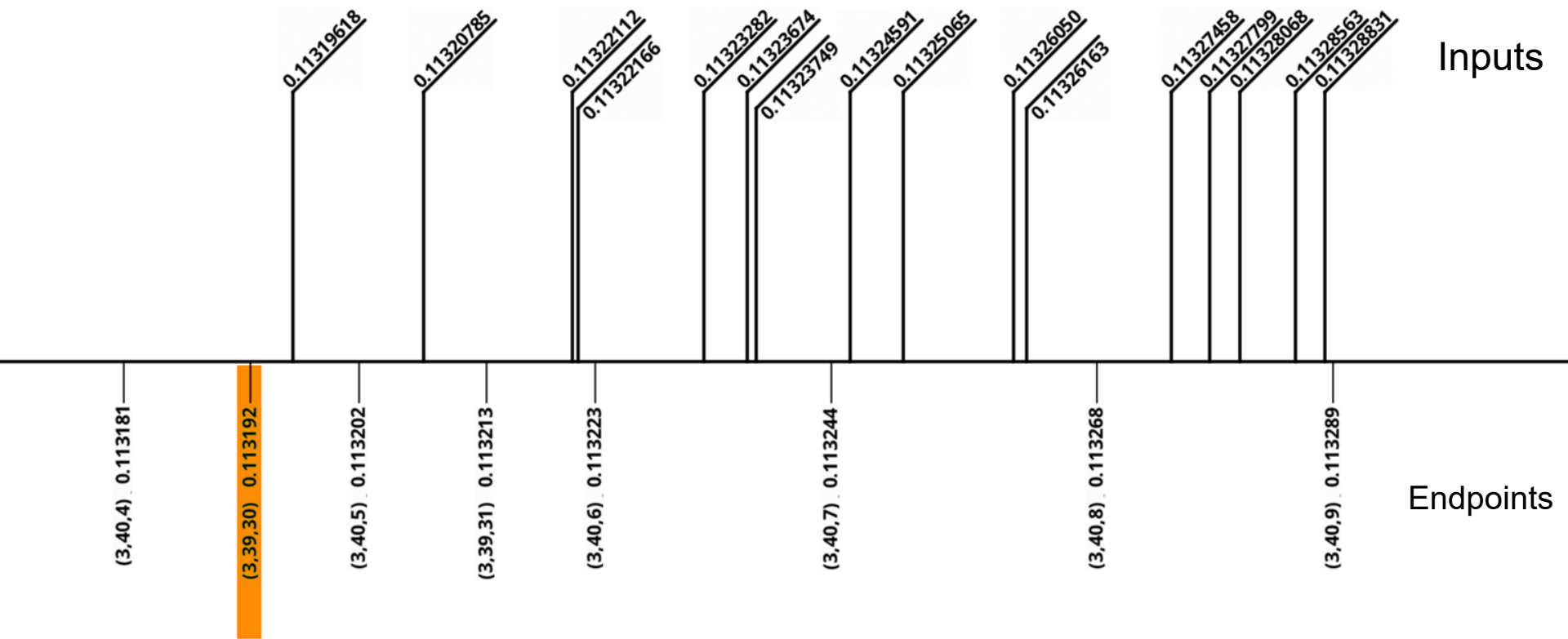


Inputs

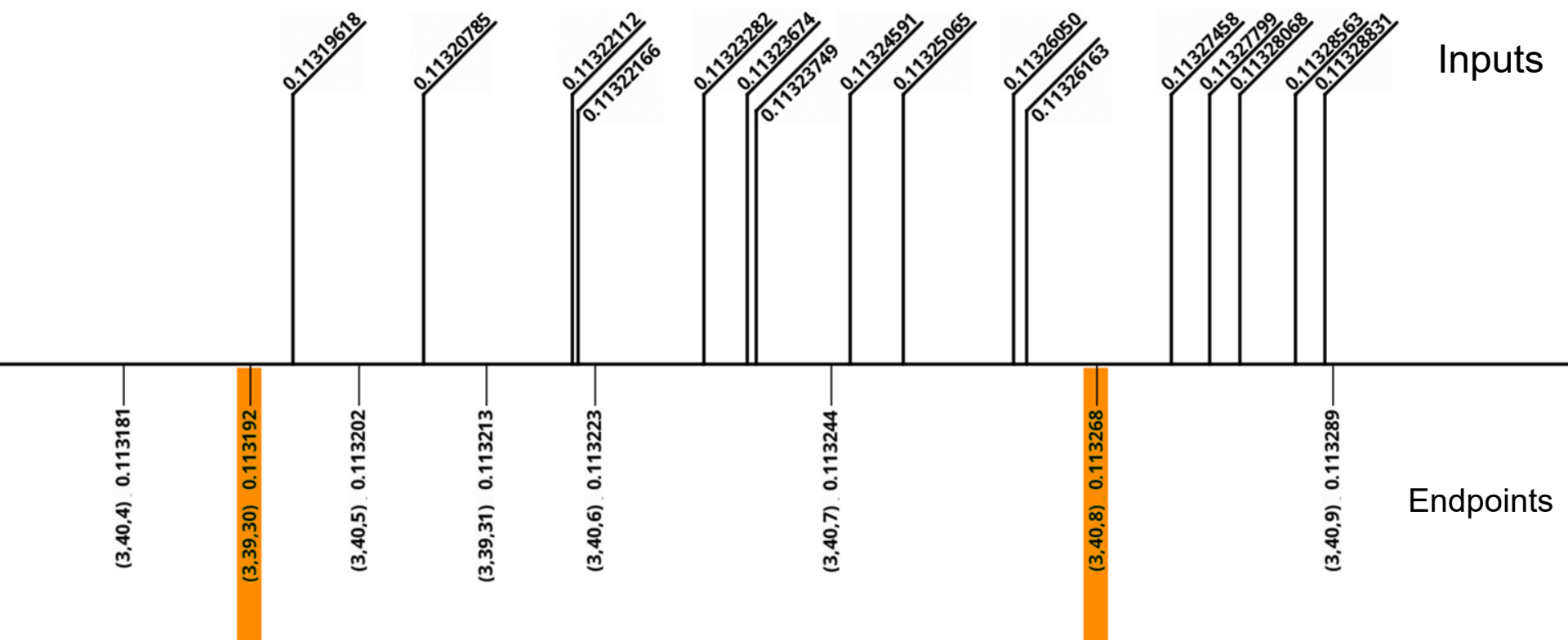
Encoder - possible endpoint colors and unpacked values



Encoder - Low endpoint color



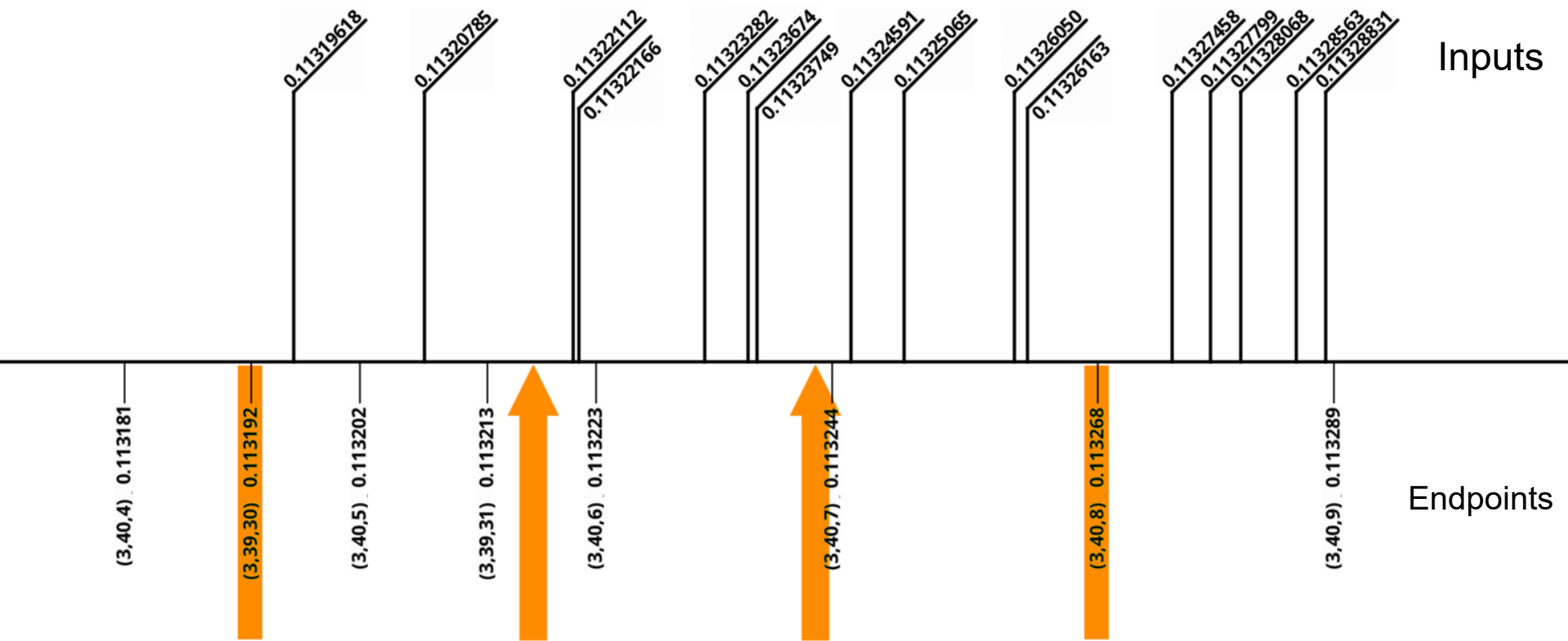
Encoder - Possible high endpoint



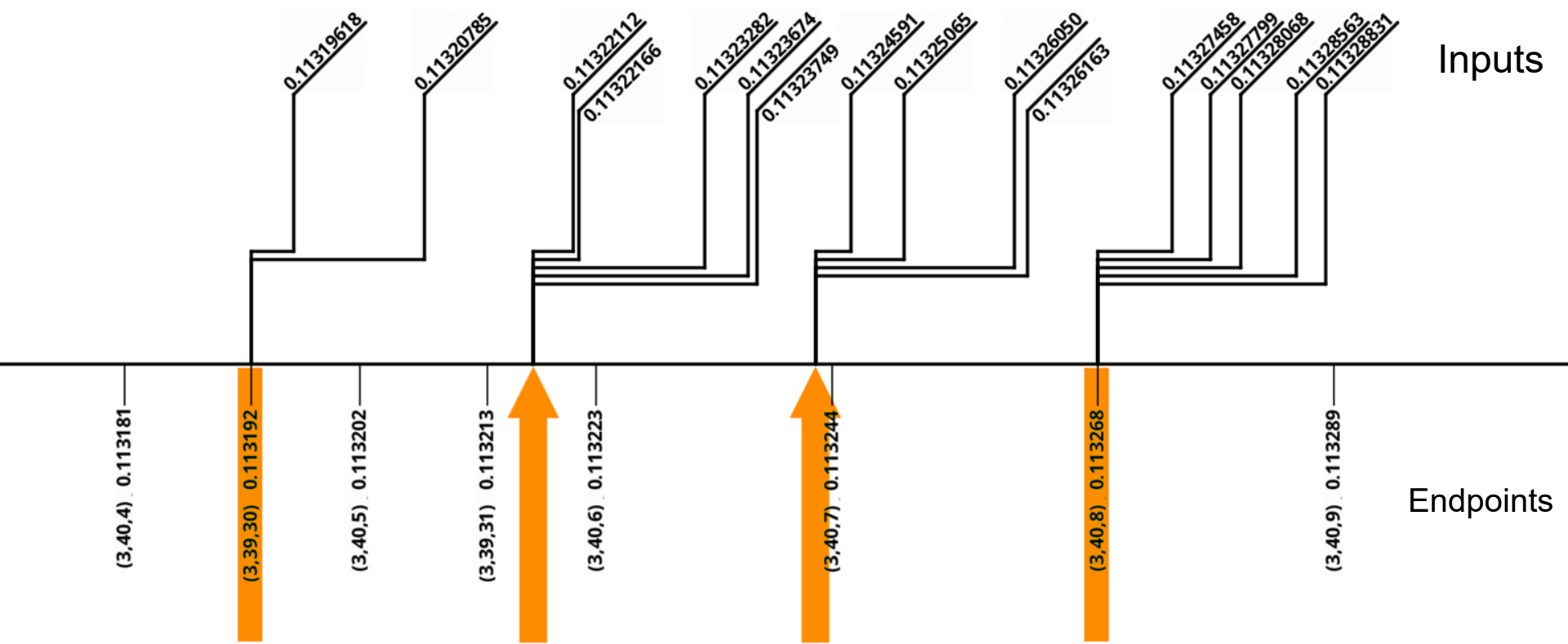
Inputs

Endpoints

Encoder - Possible high endpoint and interpolated colors



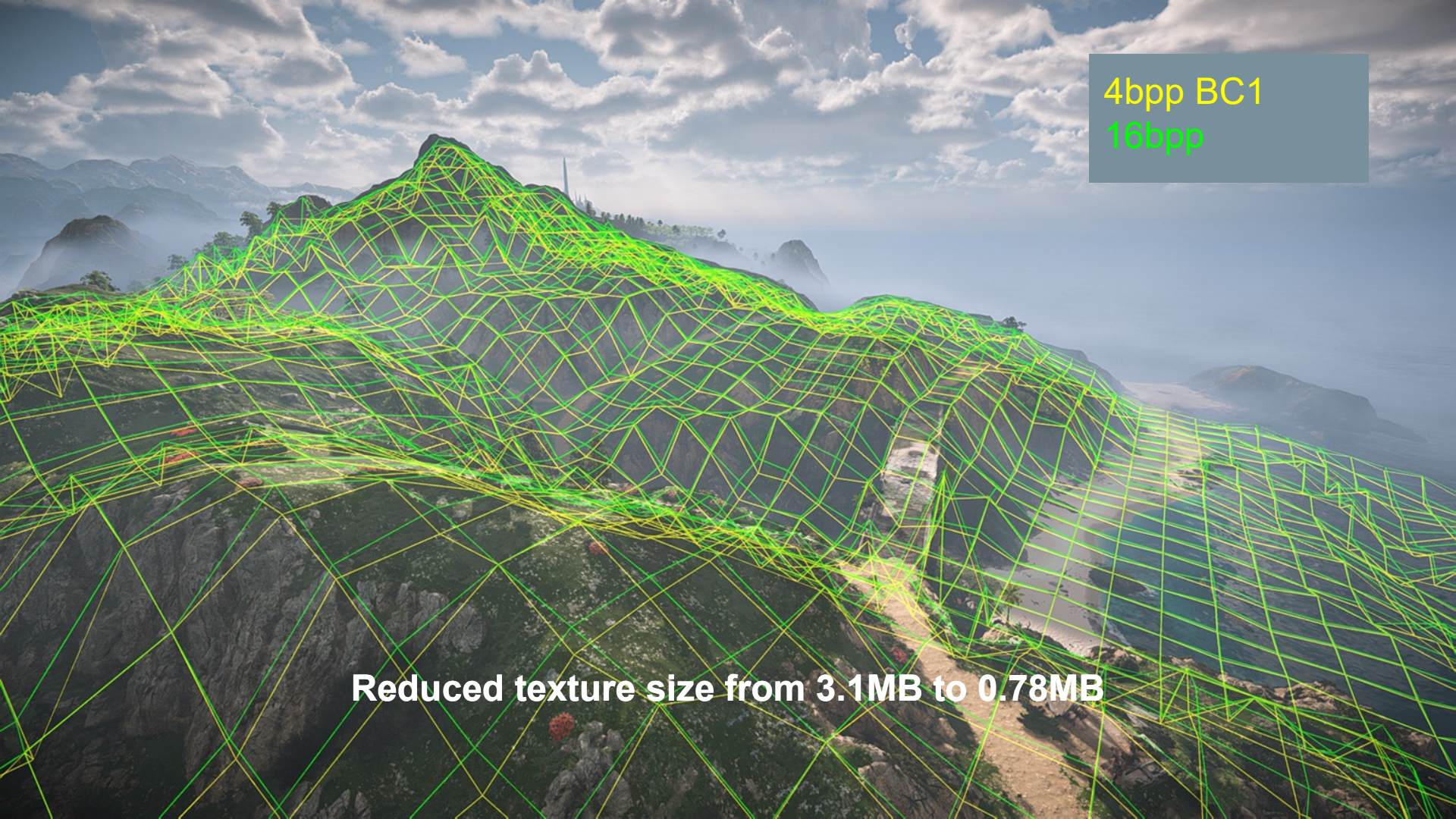
Encoder - Encoded value for each input



The image shows a blue sky with a large, fluffy white cloud in the center. Overlaid on the sky are several concentric, wavy contour lines. A yellow line follows the outer edge of the cloud, while a green line follows its inner edge. The lines are slightly offset from each other, creating a visual comparison. In the top right corner, the text '4bpp BC1' is written in yellow and '16bpp' is written in green. At the bottom center, white text states 'Reduced cloud rendering time by 30%'.

4bpp BC1
16bpp

Reduced cloud rendering time by 30%



4bpp BC1
16bpp

Reduced texture size from 3.1MB to 0.78MB

<https://fgiesen.wordpress.com/2021/10/04/gpu-bcn-decoding/>

“[This] level of precision is unlikely to ever matter to anyone.”

Fabian “ryg” Giesen

Questions?



References

“GPU BCn decoding”, Fabien Giesen (2021) <https://fgiesen.wordpress.com/2021/10/04/gpu-bcn-decoding/>

“GPU DXT Decompression”, Ignacio Castaño (2009) <https://www.ludicon.com/castano/blog/2009/03/gpu-dxt-decompression/>

EXT_texture_compression_dxt1 definition (2008)

https://registry.khronos.org/OpenGL/extensions/EXT/EXT_texture_compression_dxt1.txt

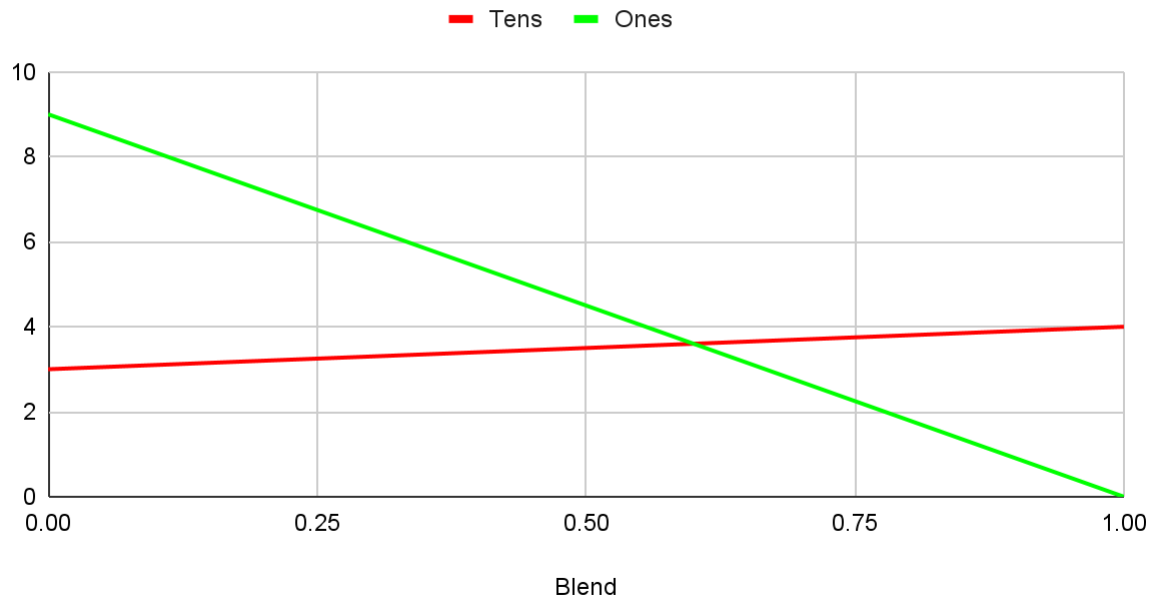
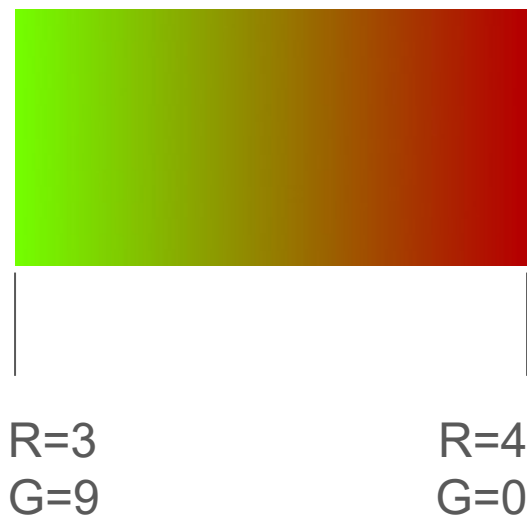


BONUS SLIDES

Failure cases with the basic endpoint search

Bilinear sampling

Linear interpolation from 3&9 to 4&0



BC5