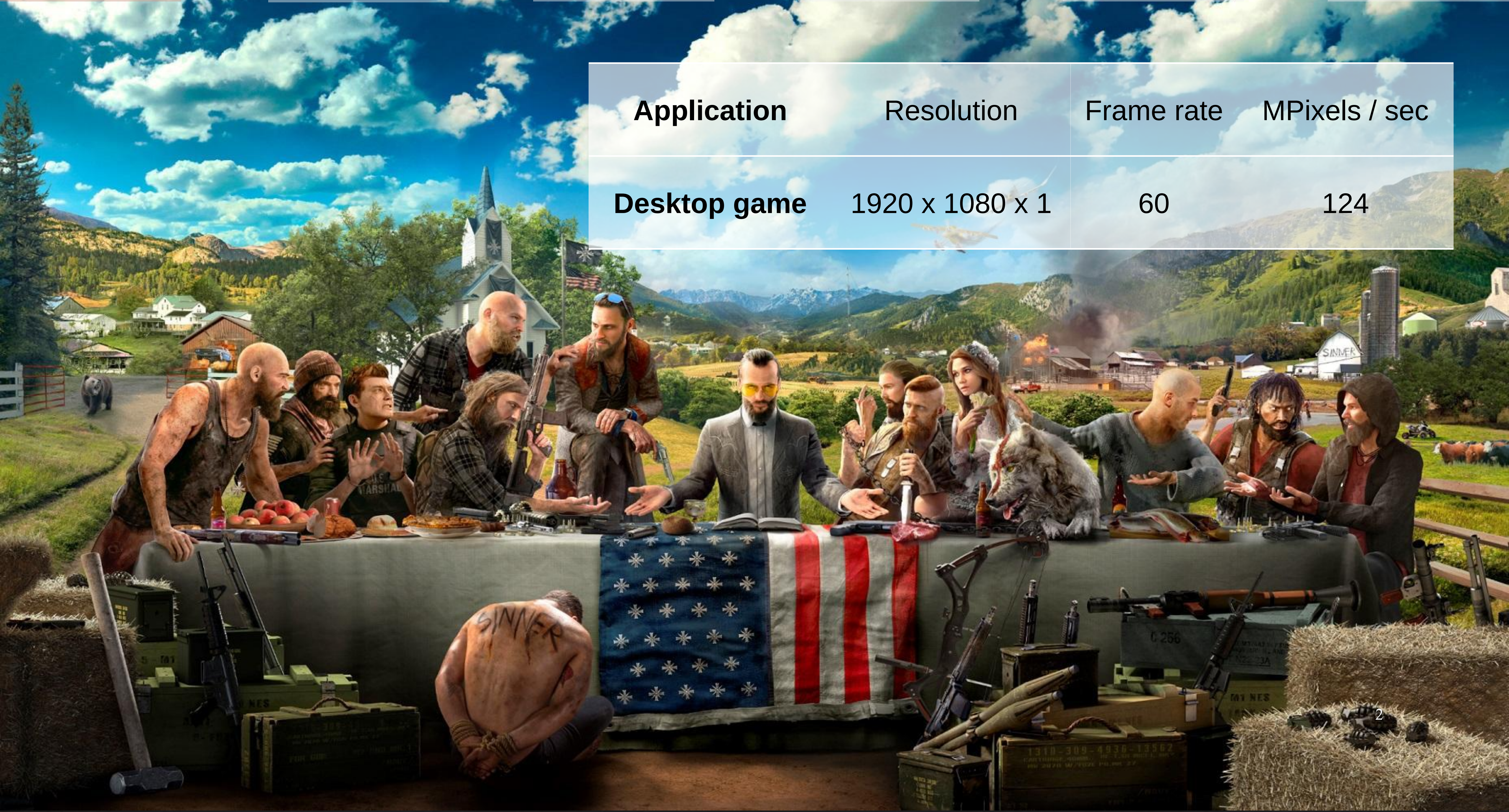
Kernel Foveated Rendering

Xiaoxu Meng, Ruofei Du, Matthias Zwicker and Amitabh Varshney
Augmentarium | UMIACS
University of Maryland, College Park

ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games 2018

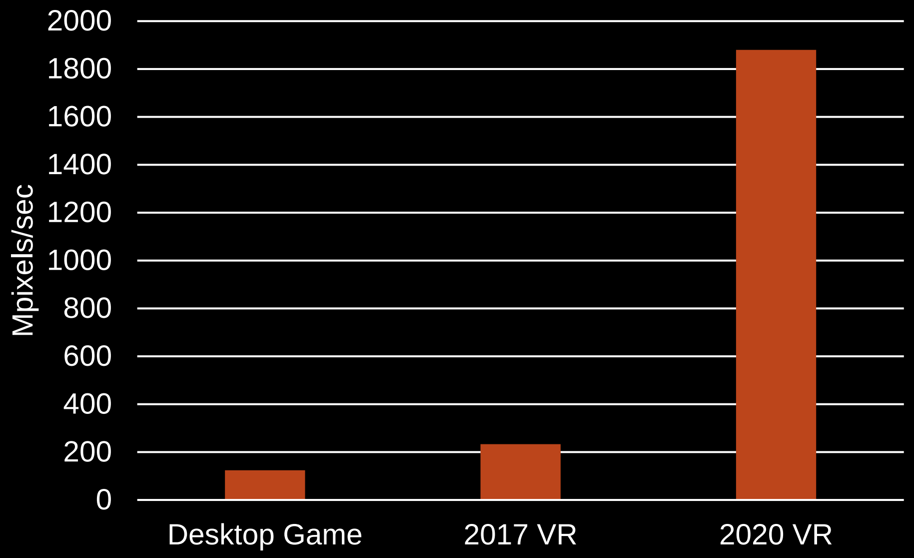
Application	Resolution	Frame rate	MPixels / sec
Desktop game	1920 x 1080 x 1	60	124



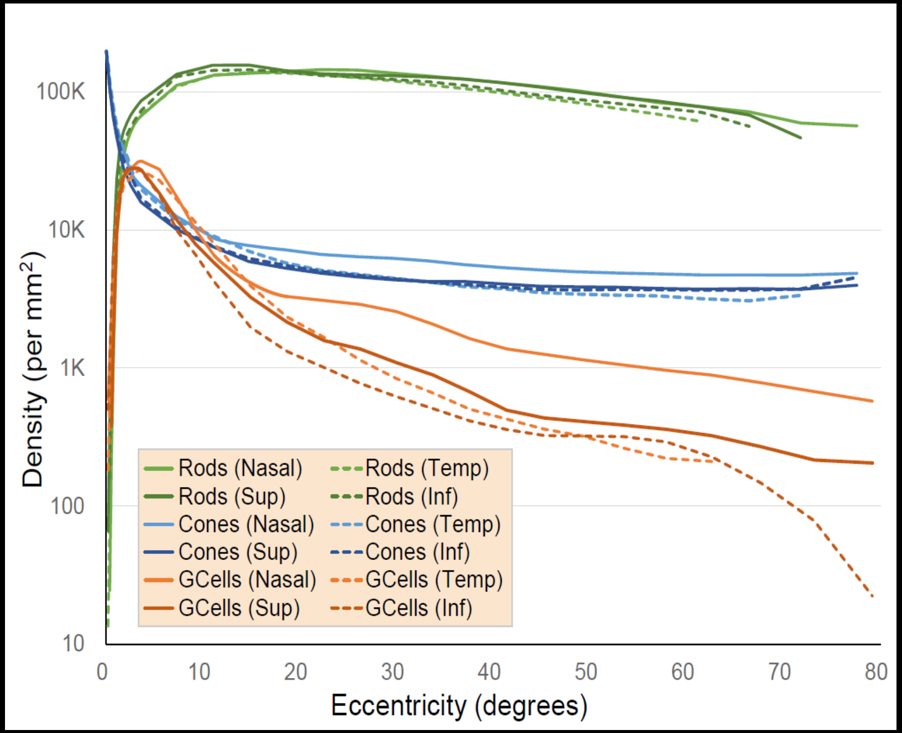
Application	Resolution	Frame rate	MPixels / sec
Desktop game	1920 x 1080 x 1	60	124
2018 VR (HTC Vive PRO)	1440 x 1600 x 2	90	414

Application	Resolution	Frame rate	MPixels / sec
Desktop game	1920 x 1080 x 1	60	124
2018 VR (HTC Vive PRO)	1440 x 1600 x 2	90	414
2020 VR *	4000 x 4000 x 2	90	2,880

* Data from Siggraph Asia 2016, Prediction by Michael Abrash, October 2016

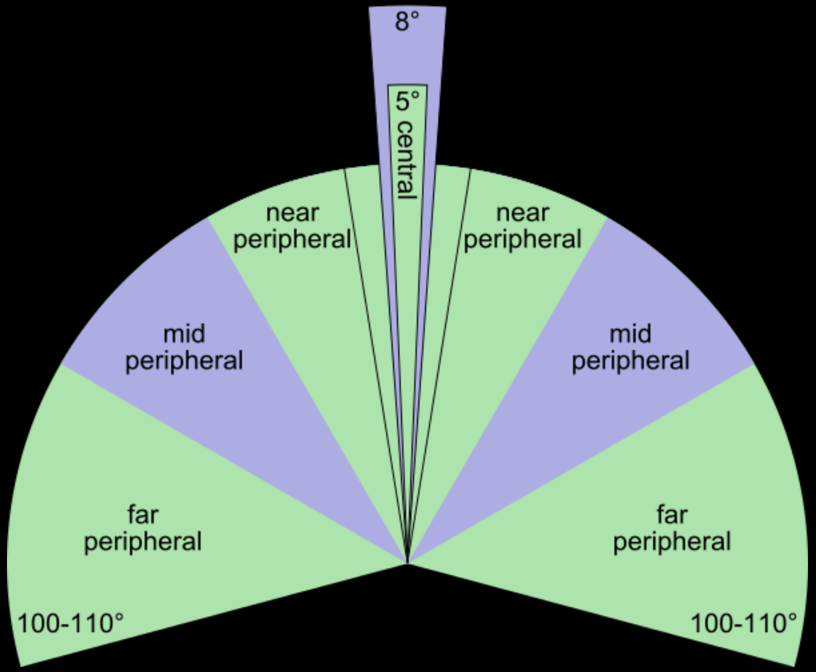


- Virtual reality is a challenging workload



fovea:
the center of the retina
corresponds to the center of the vision field

- Virtual reality is a challenging workload
- Most VR pixels are peripheral

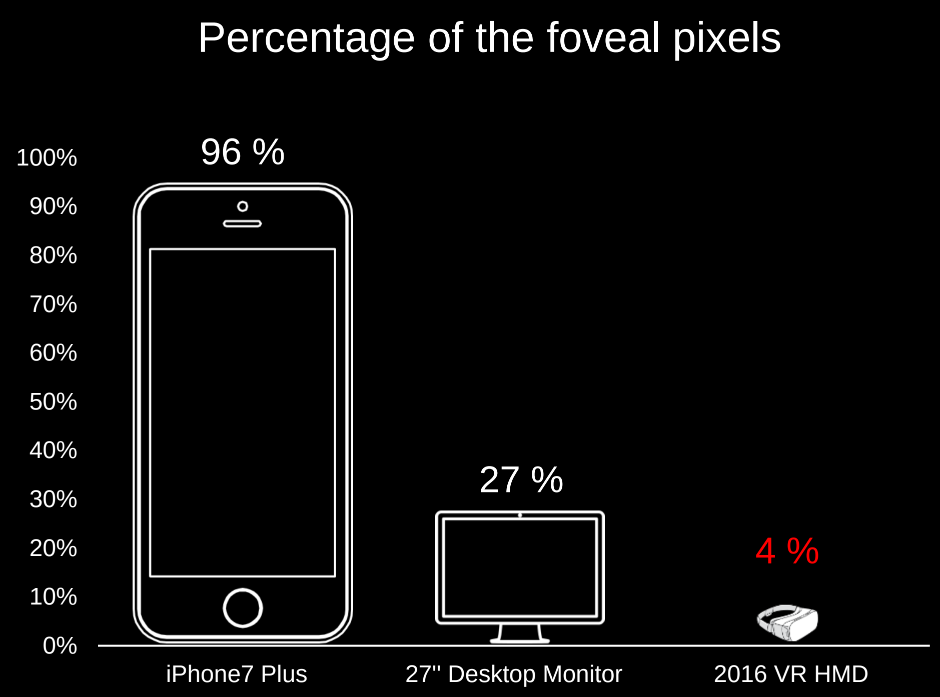


foveal region:
the human eye detects significant detail
peripheral region:
the human eye detects little high fidelity detail

- Virtual reality is a challenging workload
- Most VR pixels are peripheral

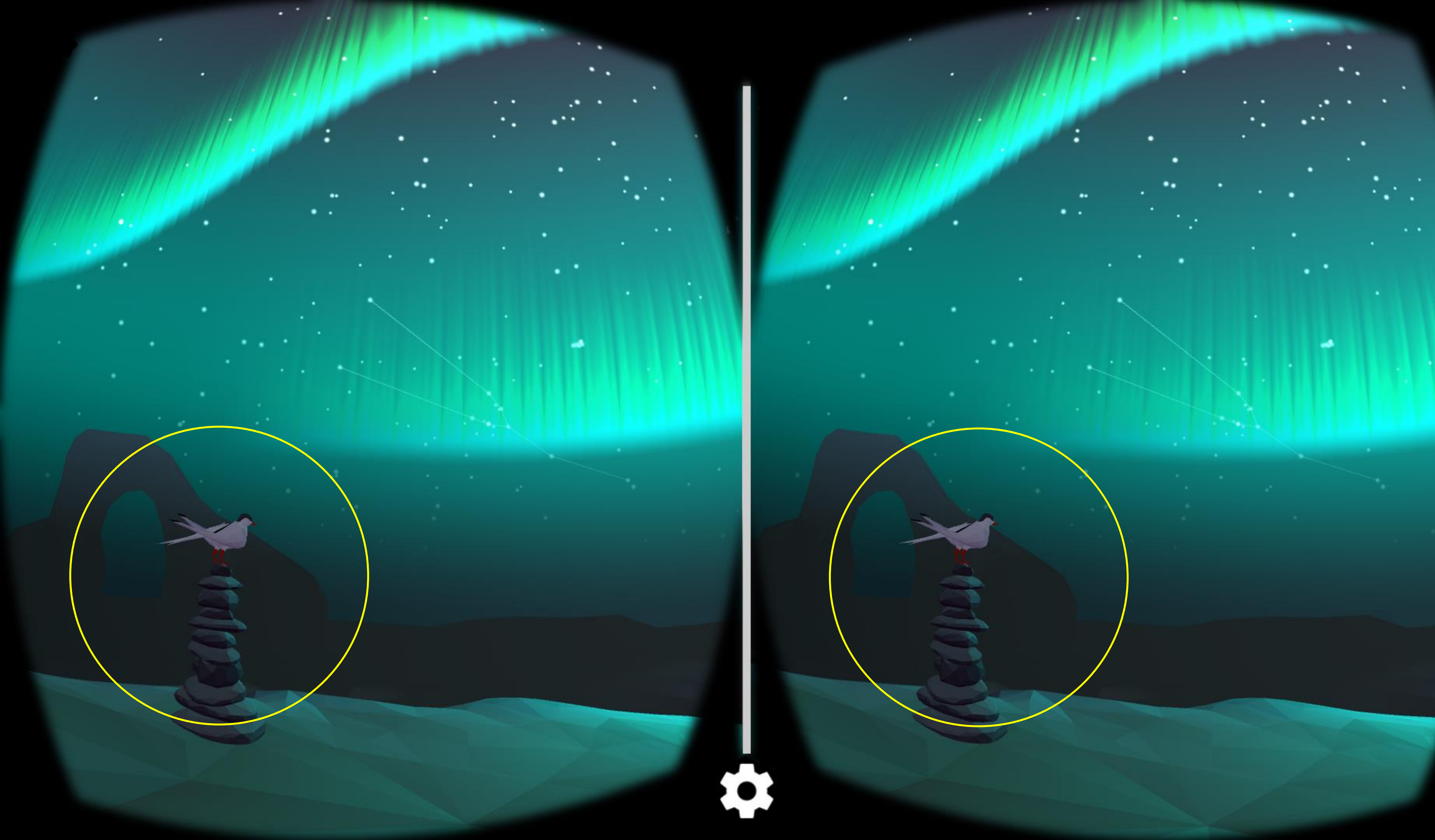
foveal region:
the human eye detects significant detail
peripheral region:
the human eye detects little high fidelity detail

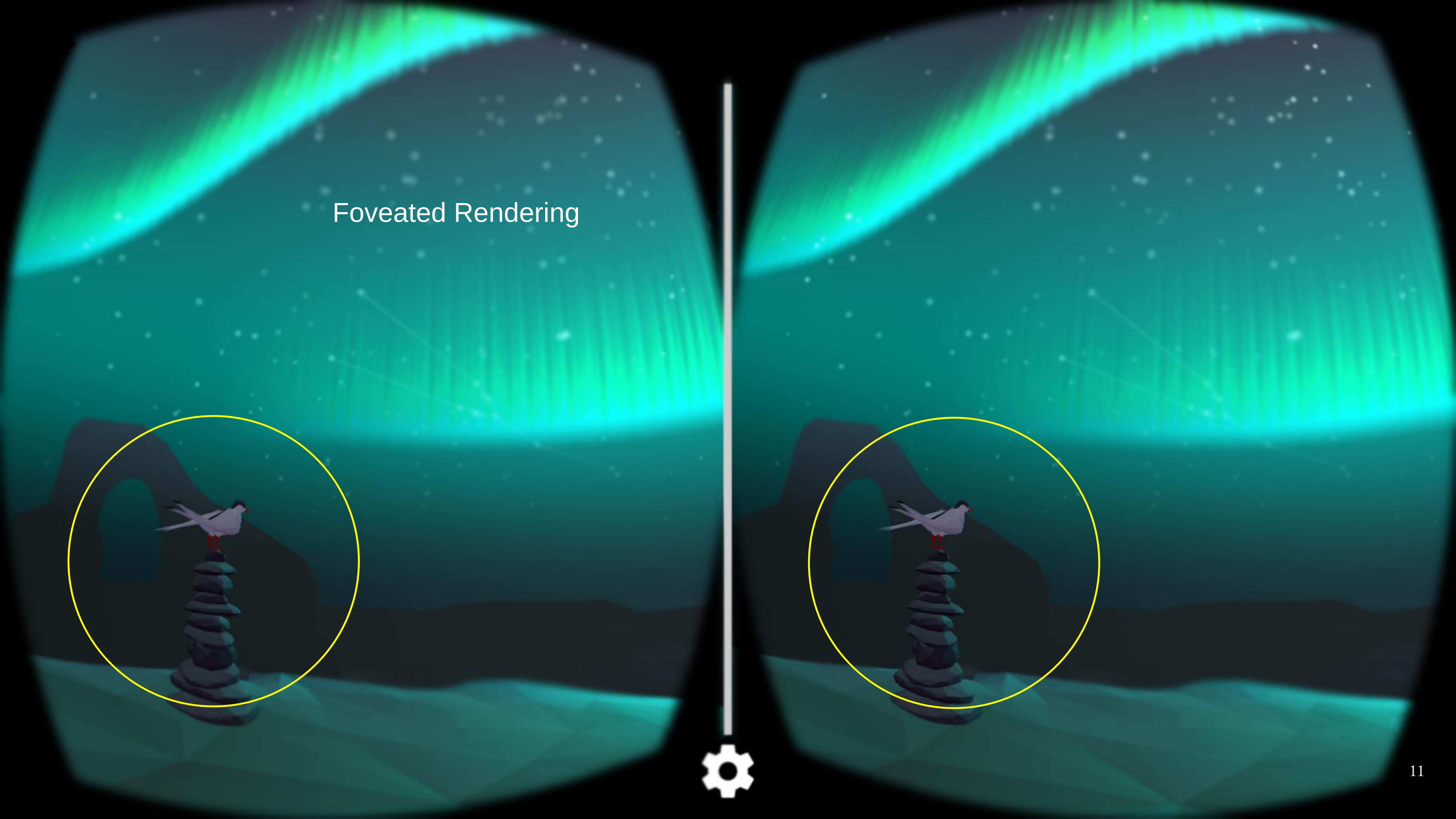
- Virtual reality is a challenging workload
- Most VR pixels are peripheral



- Virtual reality is a challenging workload
- Most VR pixels are peripheral

* Data from Siggraph 2017, by Anjul Patney, August 2017

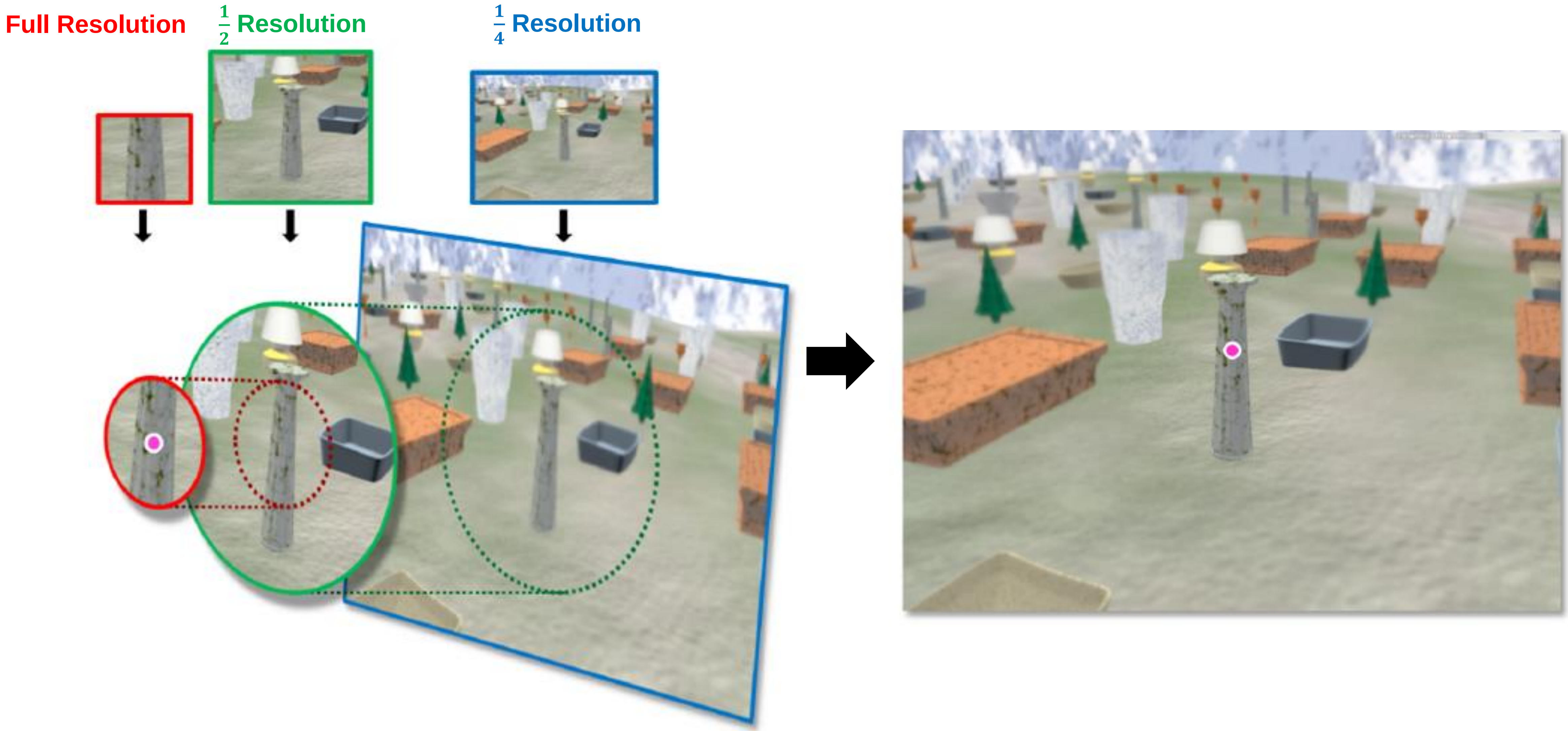



Foveated Rendering

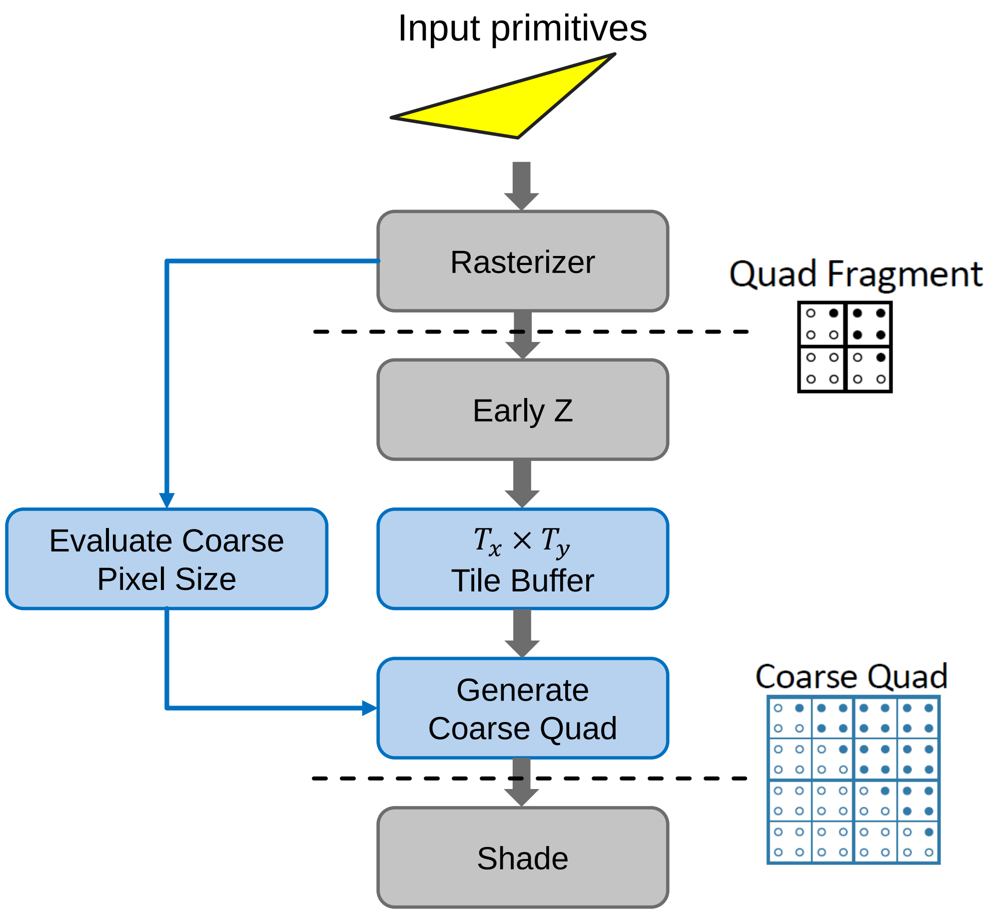
- Virtual reality is a challenging workload
- Most VR pixels are peripheral
- Eye tracking technology available

Foveate
Brian Guenter
Mark Finch
High Performance Graphics (2014)
Jonathan Ragan-Kelley and Ingo Wald (Editors)
Coarse Pixel Shading
PS
CPS
UPSCALING
Abstract
1 Introduction
DOI: 10.1111/cgf.12956
Eurographics Symposium on Rendering 2016
E. Eisemann and E. Fiume
(Guest Editors)
Adaptiv
Gaze-Con
Michael Stengel
Figure 1: Gaze-contingent Rendering Pipeline.
Abstract
1. Introduction
Towards Foveated Rendering for Gaze-Tracked Virtual Reality
Anjul Patney*
Marco Salvi
Joohwan Kim
David Luebke
Anton Kaplanyan
Aaron Lefohn
Chris Wyman
Nir Benty
NVIDIA
Fovea
Periphery
Fovea
Periphery
Abstract
1 Introduction

Multi-Pass Foveated Rendering [Guenter et al. 2012]



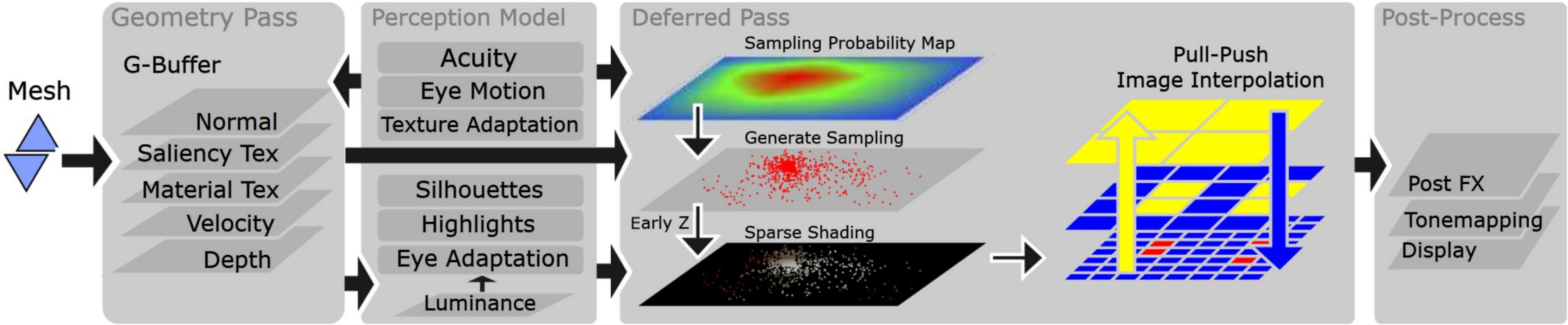
Coarse Pixel Shading (CPS) [Vaidyanathan et al. 2014]



CPS with TAA & Contrast Preservation [Patney et al. 2016]

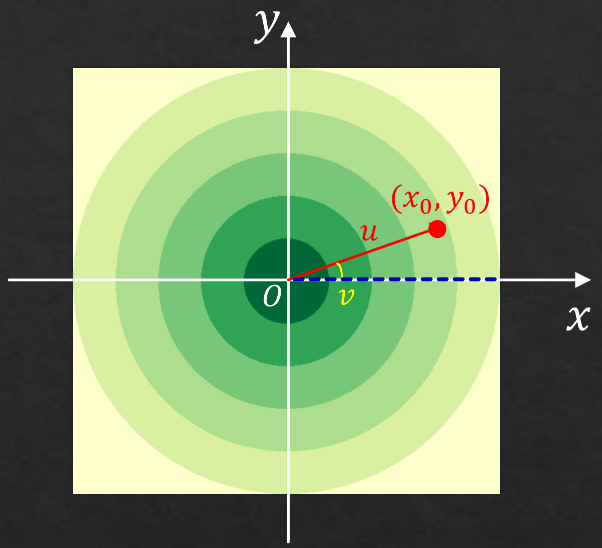
Can we change the resolution gradually?

Perceptual Foveated Rendering [Stengel et al. 2016]

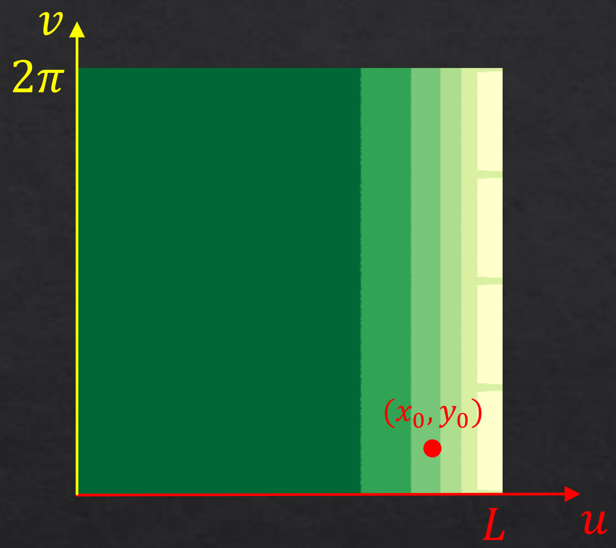


Is there a foveated rendering approach *without* the expensive pixel interpolation?

Log-polar mapping [Araujo and Dias 1996]



Cartesian coordinates
(x, y)



Log-polar coordinates
(u, v)

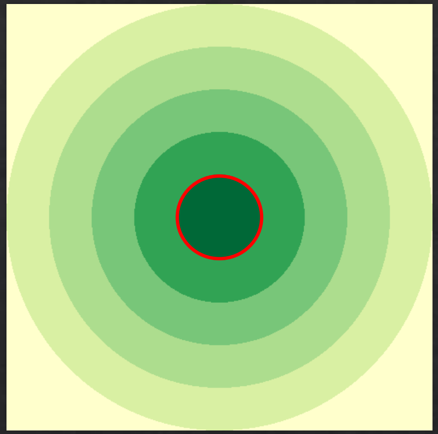
Log-polar Mapping

$$u = \frac{\log\sqrt{x^2 + y^2}}{L} \cdot w$$

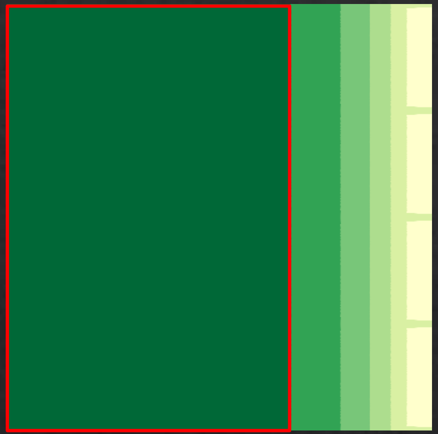
$$v = \frac{(\arctan\frac{y}{x} + \mathbf{1}\,[y - 0] \cdot 2\pi)}{2\pi} \cdot h$$

- $W: screen\ width$ $\quad H: screen\ height$ $\quad w: buffer\ width$ $\quad h: buffer\ height$
- $\mathbf{1}\,[y < 0] = \begin{cases} 1 & y < 0 \\ 0 & y > 0 \end{cases}$
- $L = \log\sqrt{W^2 + H^2}$

Log-polar mapping [Araujo and Dias 1996]



Cartesian coordinates
(x, y)



Log-polar coordinates
(u, v)

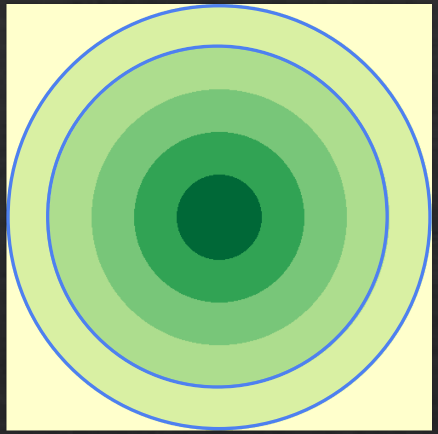
Log-polar Mapping

$$u = \frac{\log\sqrt{x^2 + y^2}}{L} \cdot w$$

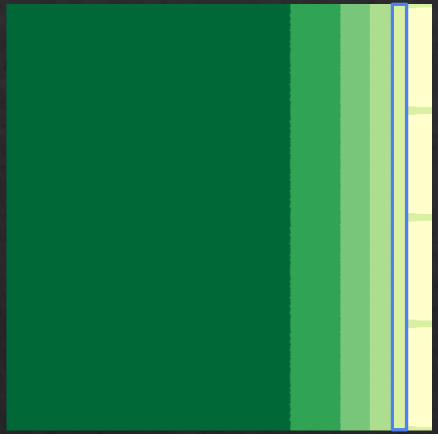
$$v = \frac{(\arctan\frac{y}{x} + \mathbf{1}\,[y - 0] \cdot 2\pi)}{2\pi} \cdot h$$

- $W: screen\ width \quad H: screen\ height \quad w: buffer\ width \quad h: buffer\ height$
- $\mathbf{1}\,[y < 0] = \begin{cases} 1 & y < 0 \\ 0 & y > 0 \end{cases}$
- $L = \log\sqrt{W^2 + H^2}$

Log-polar mapping [Araujo and Dias 1996]



Cartesian coordinates
(x, y)



Log-polar coordinates
(u, v)

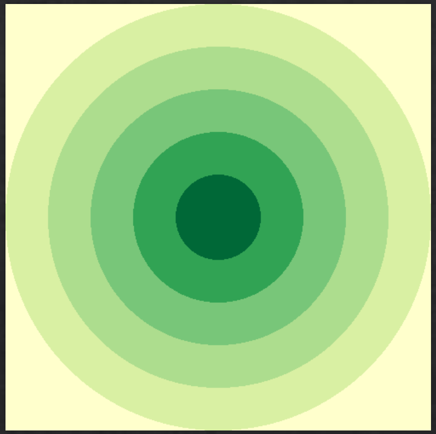
Log-polar Mapping

$$u = \frac{\log\sqrt{x^2 + y^2}}{L} \cdot w$$

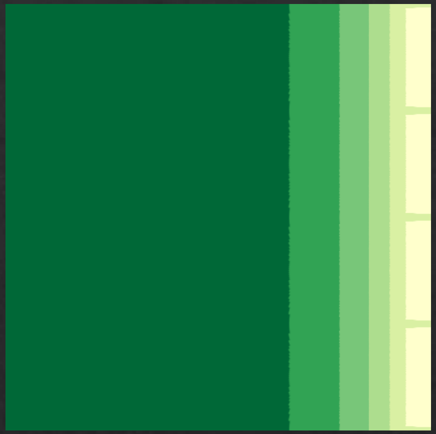
$$v = \frac{(\arctan\frac{y}{x} + \mathbf{1}\,[y - 0] \cdot 2\pi)}{2\pi} \cdot h$$

- $W: screen\ width \quad H: screen\ height \quad w: buffer\ width \quad h: buffer\ height$
- $\mathbf{1}\,[y < 0] = \begin{cases} 1 & y < 0 \\ 0 & y > 0 \end{cases}$
- $L = \log\sqrt{W^2 + H^2}$

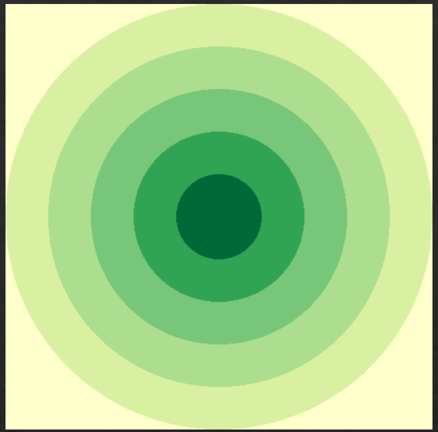
Log-polar mapping [Araujo and Dias 1996]



Cartesian coordinates
(x, y)



Log-polar coordinates
(u, v)



Cartesian coordinates
(x, y)

Log-polar Mapping

$$u = \frac{\log\sqrt{x^2 + y^2}}{L} \cdot w$$

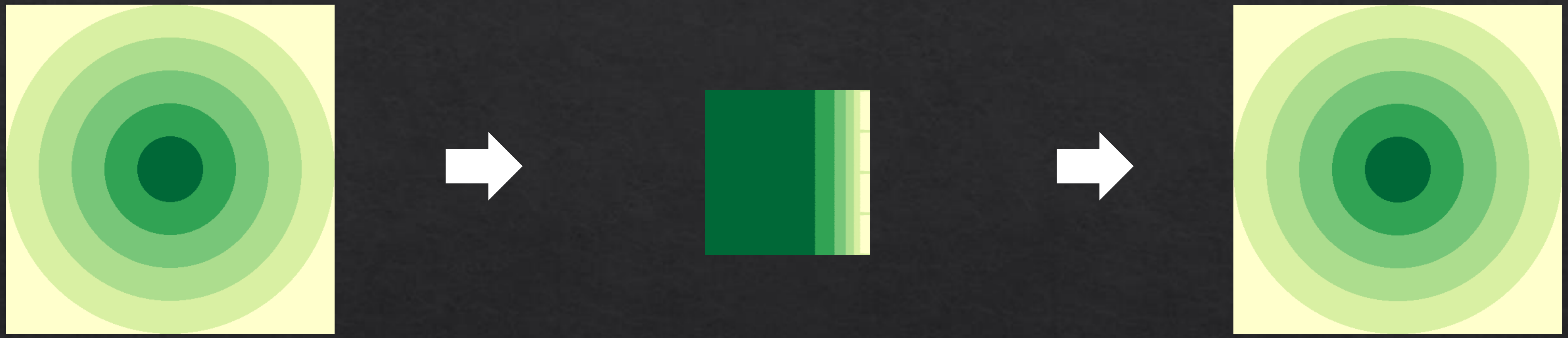
$$v = \frac{(\arctan\frac{y}{x} + \mathbf{1}\,[y - 0] \cdot 2\pi)}{2\pi} \cdot h$$

$$x = e^{L \cdot \frac{u}{w}} \cos\left(v \cdot \frac{2\pi}{h}\right)$$

$$y = e^{L \cdot \frac{u}{w}} \sin\left(v \cdot \frac{2\pi}{h}\right)$$

- W: *screen width* H: *screen height* w: *buffer width* h: *buffer height*
- $\mathbf{1}\,[y < 0] = \begin{cases} 1 & y < 0 \\ 0 & y > 0 \end{cases}$
- $L = \log\sqrt{W^2 + H^2}$

Log-polar mapping [Araujo and Dias 1996]



Cartesian coordinates (x, y)

Log-polar coordinates (u, v)

Cartesian coordinates (x, y)

Log-polar Mapping

$$u = \frac{\log\sqrt{x^2 + y^2}}{L} \cdot w$$

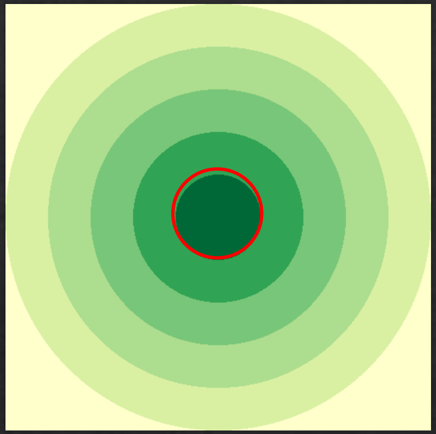
$$x = e^{L \cdot \frac{u}{w}} \cos\left(v \cdot \frac{2\pi}{h}\right)$$

$$v = \frac{\left(\arctan\frac{y}{x} + \mathbf{1}\,[y - 0] \cdot 2\pi\right)}{2\pi} \cdot h$$

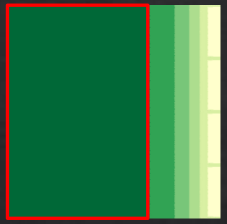
$$y = e^{L \cdot \frac{u}{w}} \sin\left(v \cdot \frac{2\pi}{h}\right)$$

- W: *screen width* H: *screen height* w: *buffer width* h: *buffer height*
- $\mathbf{1}\,[y < 0] = \begin{cases} 1 & y < 0 \\ 0 & y > 0 \end{cases}$
- $L = \log\sqrt{W^2 + H^2}$

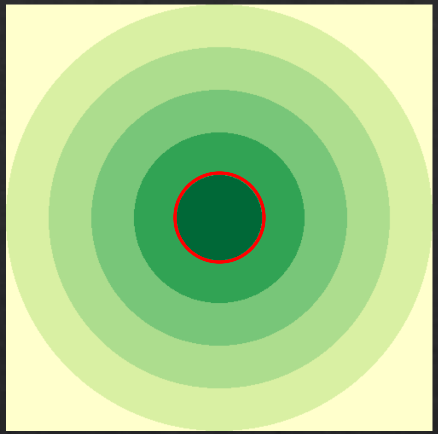
Log-polar mapping [Araujo and Dias 1996]



Cartesian coordinates
(x, y)



Log-polar coordinates
(u, v)



Cartesian coordinates
(x, y)

Log-polar Mapping

$$u = \frac{\log\sqrt{x^2 + y^2}}{L} \cdot w$$

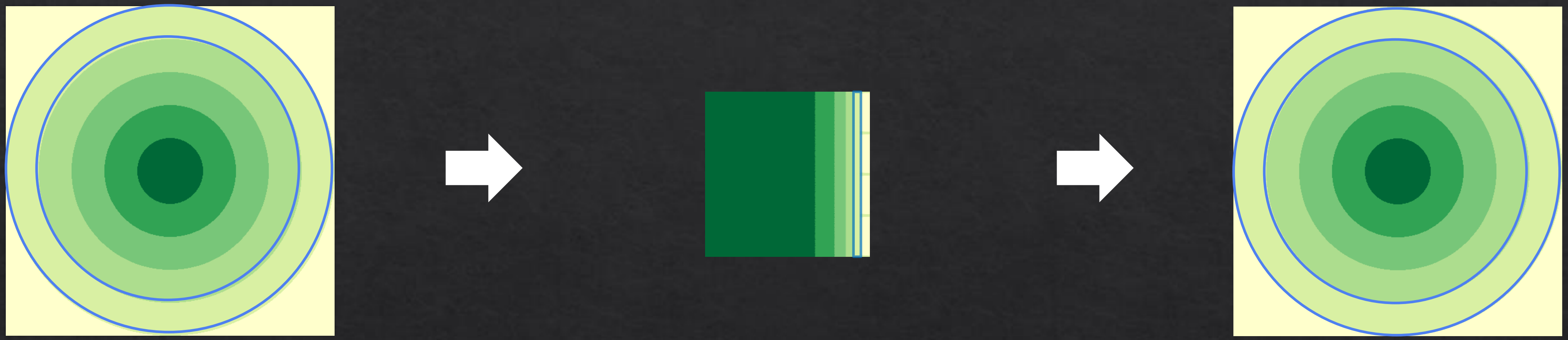
$$v = \frac{\left(\arctan\frac{y}{x} + \mathbf{1}\,[y - 0] \cdot 2\pi\right)}{2\pi} \cdot h$$

$$x = e^{L \cdot \frac{u}{w}} \cos\left(v \cdot \frac{2\pi}{h}\right)$$

$$y = e^{L \cdot \frac{u}{w}} \sin\left(v \cdot \frac{2\pi}{h}\right)$$

- $W: screen\ width \quad H: screen\ height \quad w: buffer\ width \quad h: buffer\ height$
- $\mathbf{1}\,[y < 0] = \begin{cases} 1 & y < 0 \\ 0 & y > 0 \end{cases}$
- $L = \log\sqrt{W^2 + H^2}$

Log-polar mapping [Araujo and Dias 1996]



Cartesian coordinates (x, y) → *Log-polar coordinates* (u, v) → *Cartesian coordinates* (x, y)

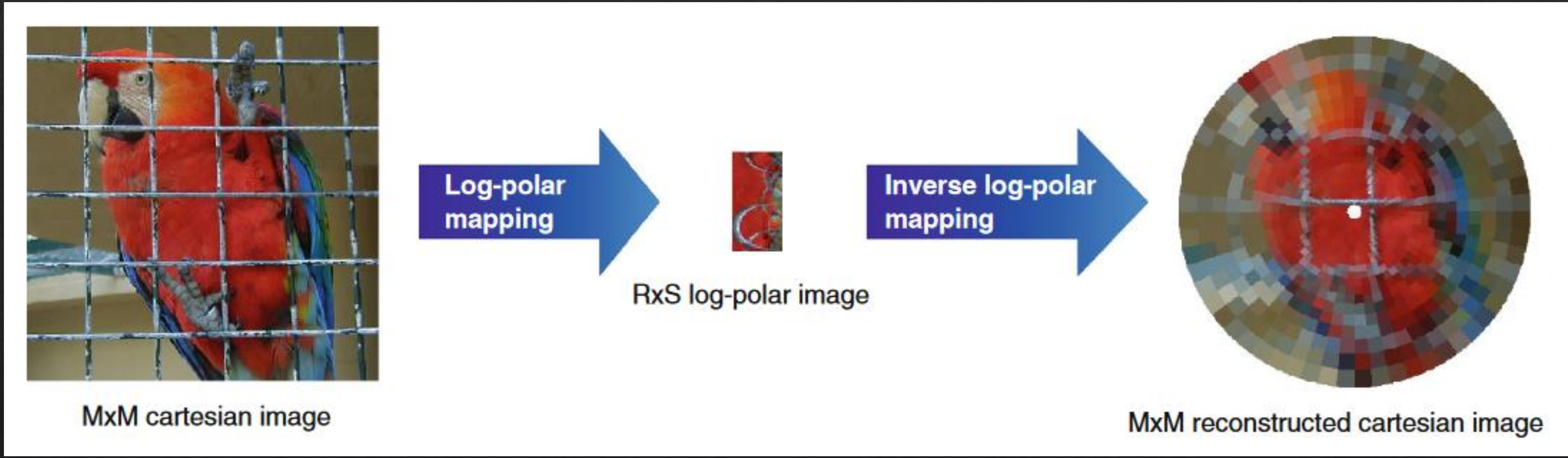
Log-polar Mapping

$$u = \frac{\log\sqrt{x^2 + y^2}}{L} \cdot w \qquad x = e^{L \cdot \frac{u}{w}} \cos\left(v \cdot \frac{2\pi}{h}\right)$$

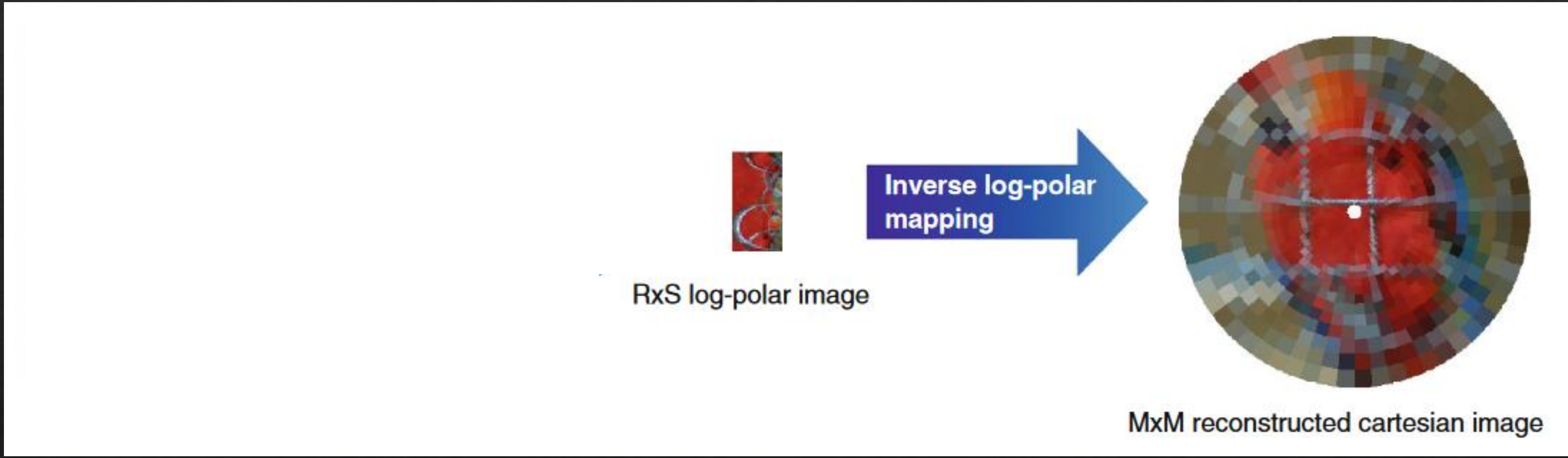
$$v = \frac{\left(\arctan\frac{y}{x} + \mathbf{1}\,[y - 0] \cdot 2\pi\right)}{2\pi} \cdot h \qquad y = e^{L \cdot \frac{u}{w}} \sin\left(v \cdot \frac{2\pi}{h}\right)$$

- $W\!: screen\ width \quad H\!: screen\ height \quad w\!: buffer\ width \quad h\!: buffer\ height$
- $\mathbf{1}\,[y < 0] = \begin{cases} 1 & y < 0 \\ 0 & y > 0 \end{cases}$
- $L = \log\sqrt{W^2 + H^2}$

Log-polar Mapping for 2D Image [Antonelli et al. 2015]

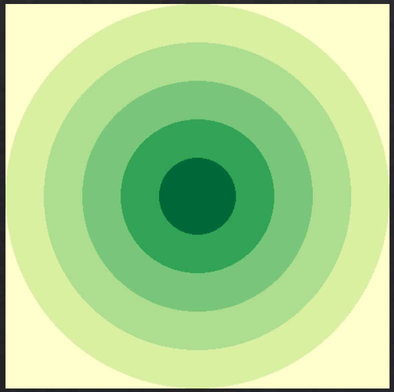


Log-polar Mapping for 2D Image



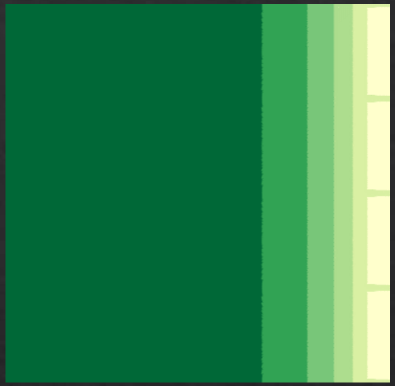
Our Approach

Kernel Log-polar Mapping

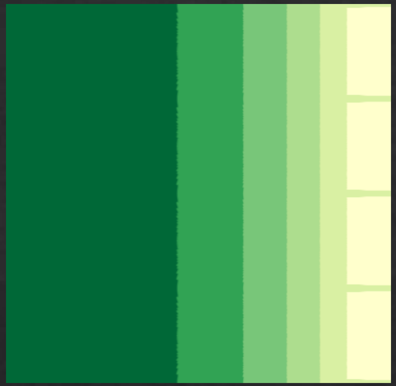


$$u = K^{-1}\left(\frac{\log\sqrt{x^2+y^2}}{L}\right)\cdot w$$

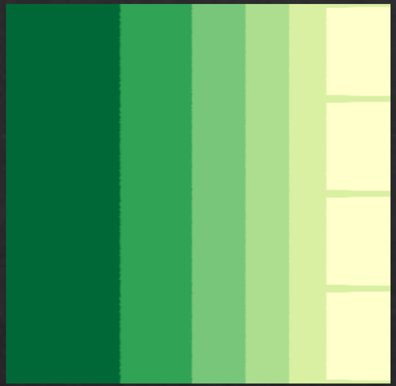
range: [0,1]



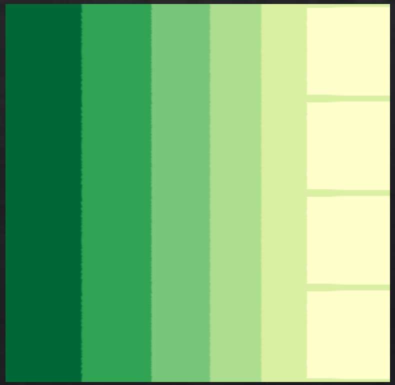
$K(x) = x$



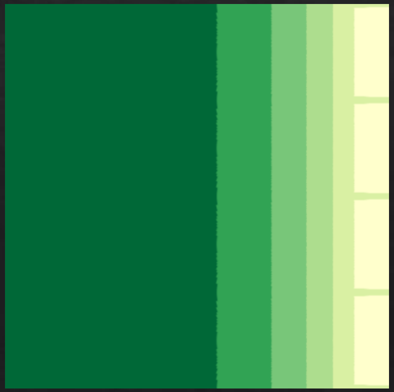
$K(x) = x^2$



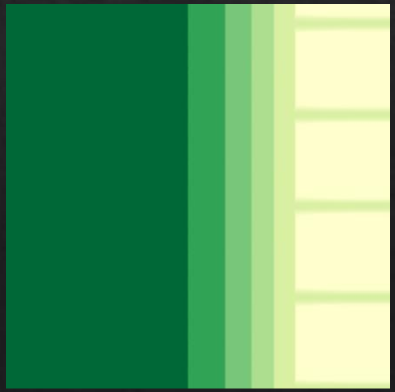
$K(x) = x^3$



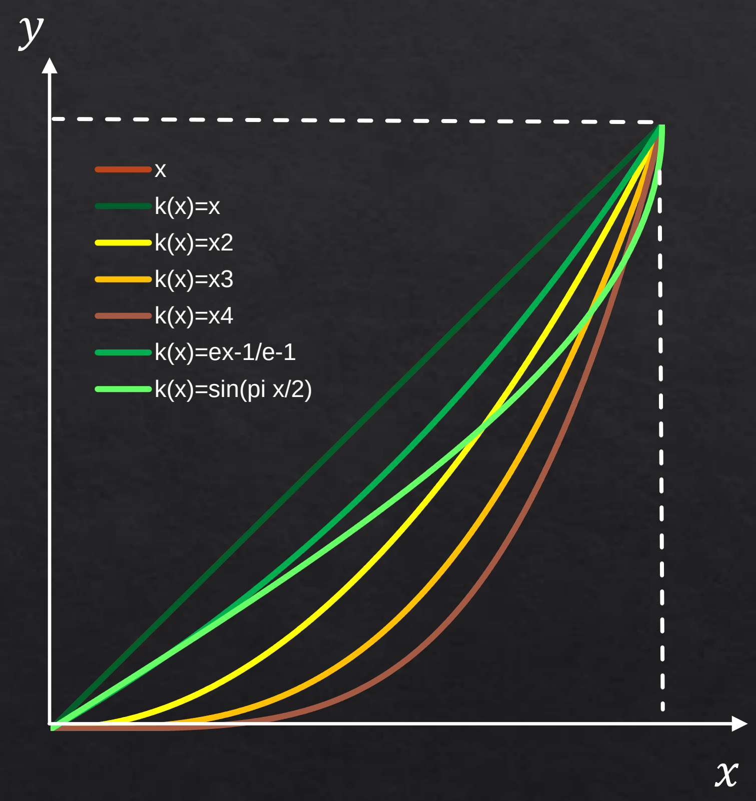
$K(x) = x^4$

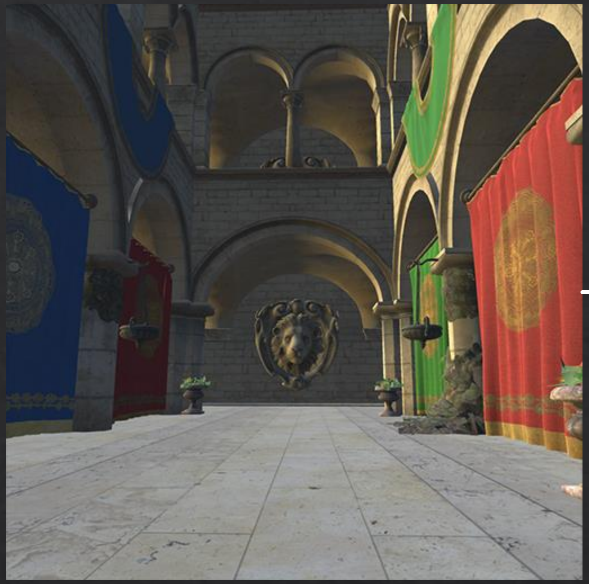


$K(x) = \frac{e^x - 1}{e - 1}$

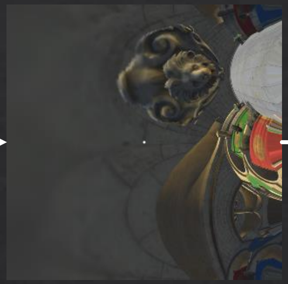


$K(x) = \sin(\frac{\pi}{2}x)$

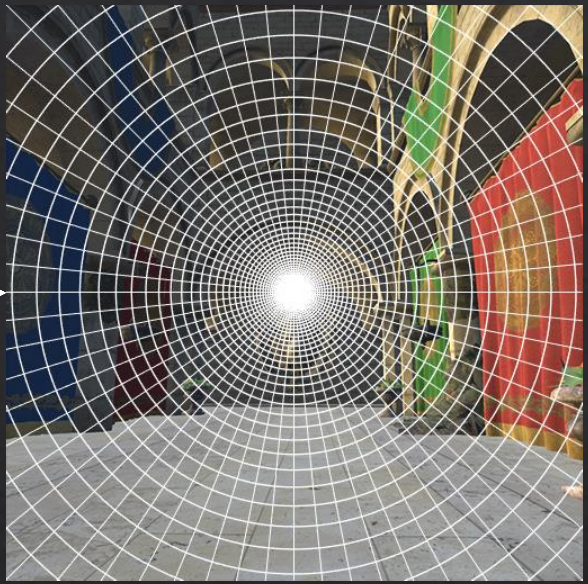




Cartesian coordinates
(x, y)



Log-polar coordinates
(u, v)



Cartesian coordinates
(x, y)

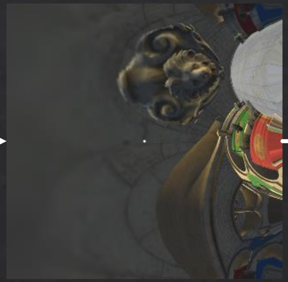
Log-polar Mapping

$$u = \frac{\log\sqrt{x^2 + y^2}}{L} \cdot w \qquad x = e^{L \cdot \frac{u}{w}} \cos\left(v \cdot \frac{2\pi}{h}\right)$$

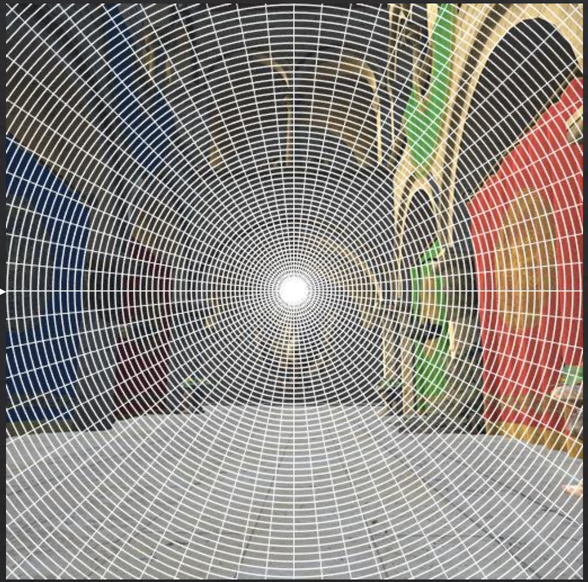
$$v = \frac{\left(\arctan\frac{y}{x} + \mathbf{1}\,[y - 0] \cdot 2\pi\right)}{2\pi} \cdot h \qquad y = e^{L \cdot \frac{u}{w}} \sin\left(v \cdot \frac{2\pi}{h}\right)$$

- $W{:}\, screen\ width \quad H{:}\, screen\ height \quad w{:}\, buffer\ width \quad h{:}\, buffer\ height$
- $\mathbf{1}\,[y < 0] = \begin{cases} 1 & y < 0 \\ 0 & y > 0 \end{cases}$
- $L = \log\sqrt{W^2 + H^2}$

Cartesian coordinates
(x, y)



Kernel log-polar coordinates
(u, v)



Cartesian coordinates
(x, y)

Kernel Log-polar Mapping

$$u = K^{-1}\left(\frac{log\sqrt{x^2 + y^2}}{L}\right) \cdot w$$

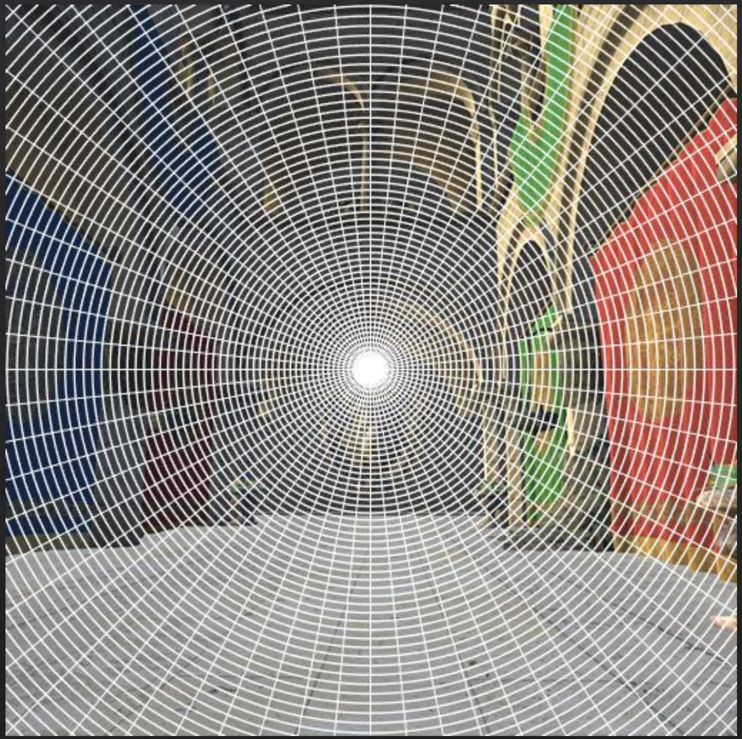
$$v = \frac{(arctan\frac{y}{x} + \mathbf{1}[y - 0] \cdot 2\pi)}{2\pi} \cdot h$$

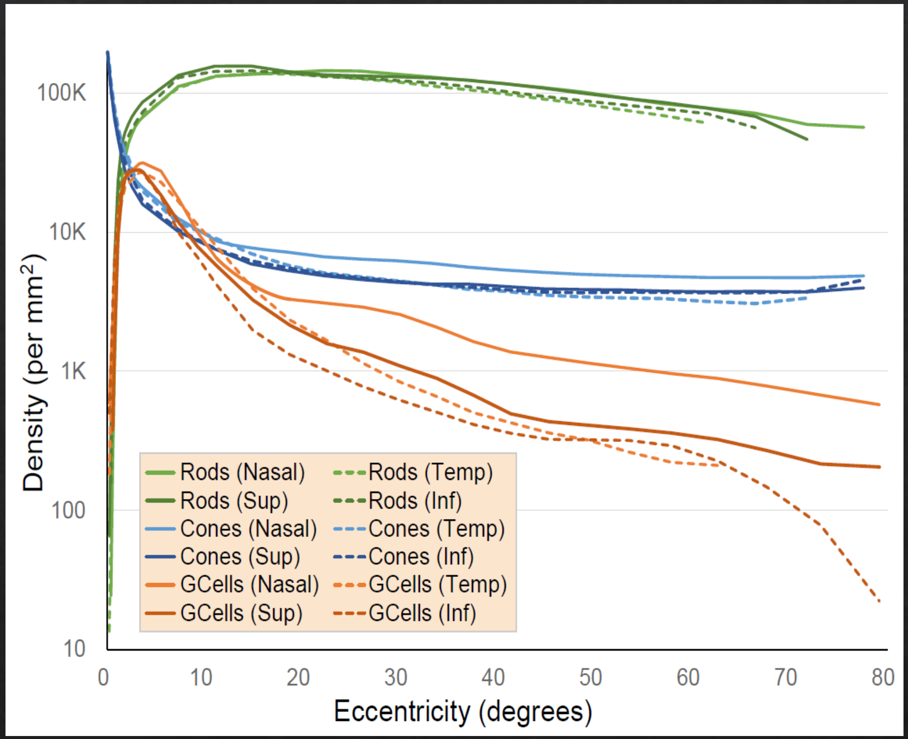
$$x = e^{L \cdot K(\frac{u}{w})} \cos\left(v \cdot \frac{2\pi}{h}\right)$$

$$y = e^{L \cdot K(\frac{u}{w})} \sin\left(v \cdot \frac{2\pi}{h}\right)$$

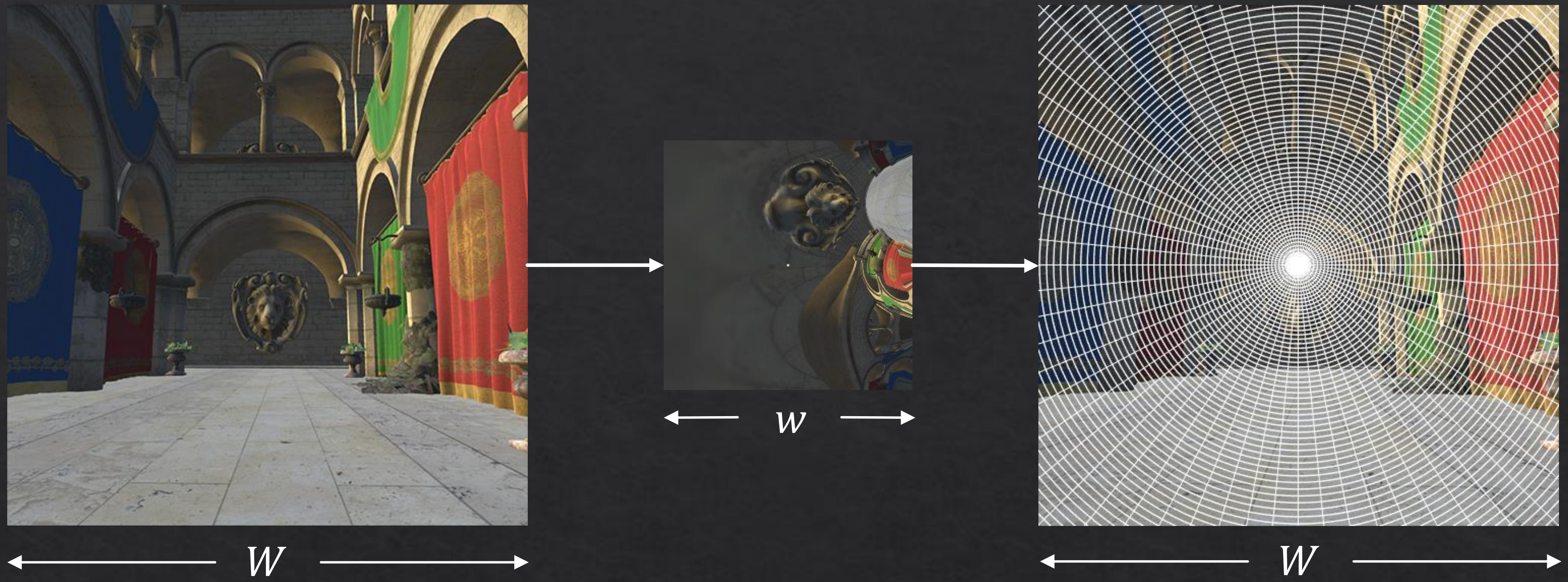
- W: *screen width* H: *screen height* w: *buffer width* h: *buffer height*
- $\mathbf{1}\,[y < 0] = \begin{cases} 1 & y < 0 \\ 0 & y > 0 \end{cases}$
- $L = \log\sqrt{W^2 + H^2}$
- $K(x) = \sum_{i=0}^{\infty} \beta_i x^i$, *where* $\sum_{i=0}^{\infty} \beta_i = 1$

Kernel Foveated Rendering





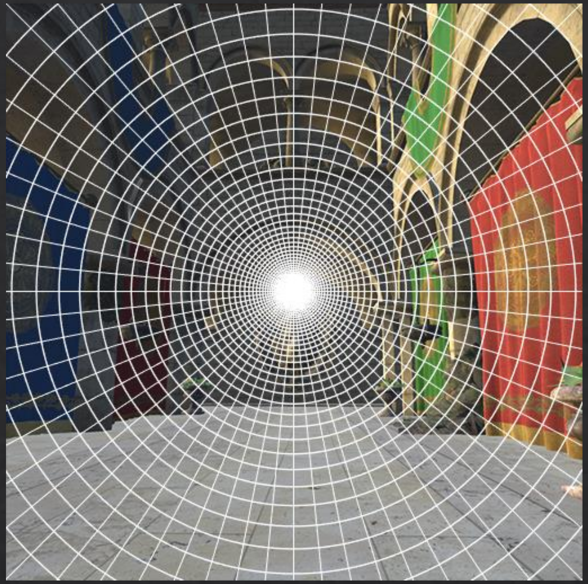
$$Distribution\ of\ pixels \xrightarrow{mimic} Distribution\ of\ photoreceptors\ in\ the\ human\ retina$$



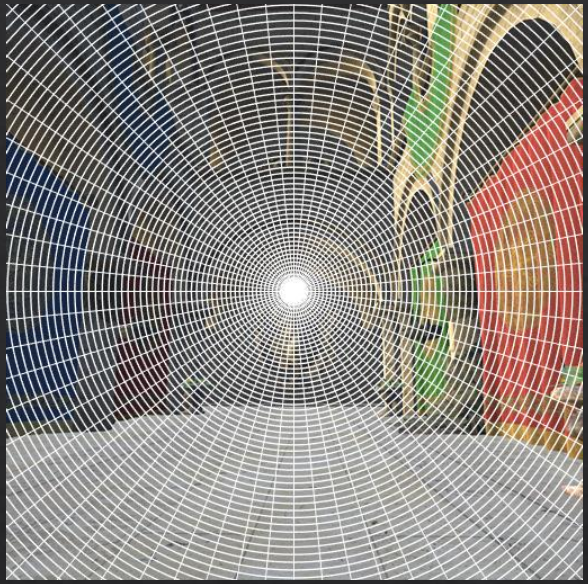
Kernel log-polar Mapping

- *Define buffer parameter σ*

$$\sigma = \frac{W}{w}$$



Result of log-polar
$(K(x) = x)$



Result of kernel log-polar
$(K(x) = x^4)$

Kernel log-polar Mapping

- *Define buffer parameter σ*

$$\sigma = \frac{W}{w}$$

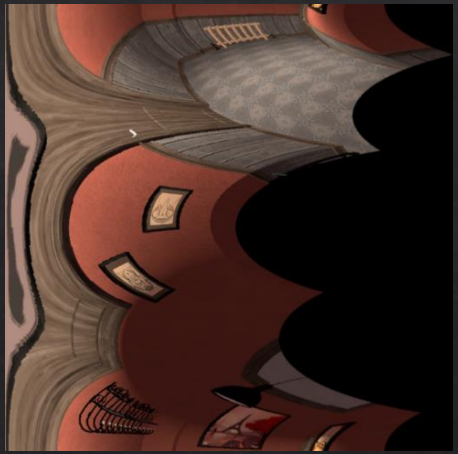
- *Define kernel function parameter α*

$$K(x) = x^{\alpha}$$

Buffer parameter
σ

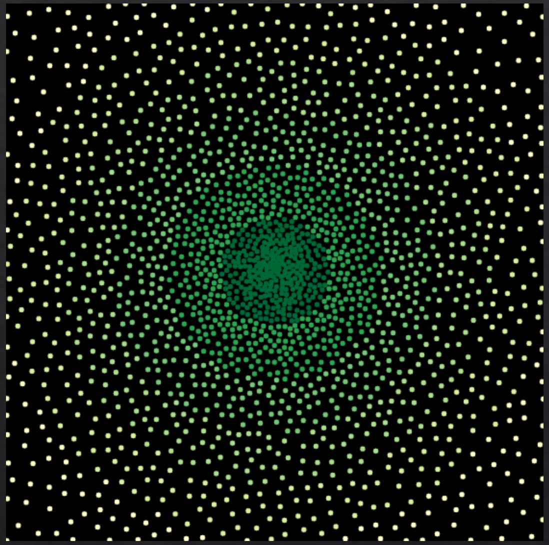
Original Frame

Buffer



Screen

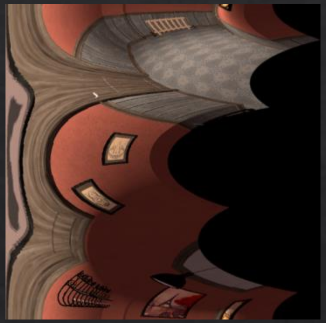
Sample Map



$$\sigma = 1.2$$

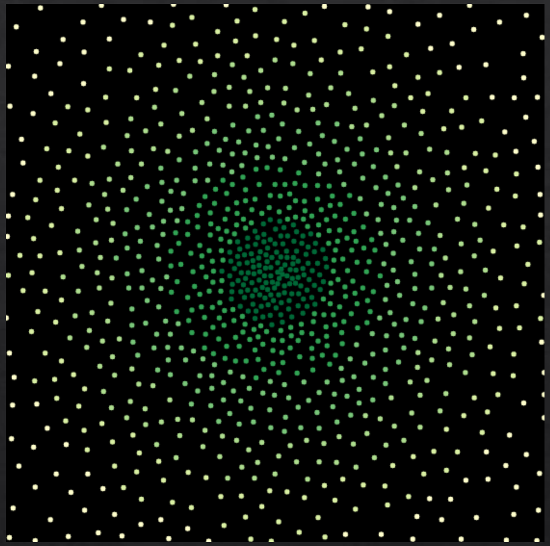
Original Frame

Buffer



Screen

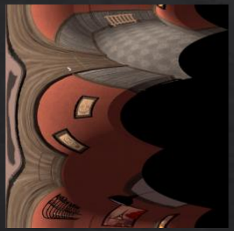
Sample Map



$$\sigma = 1.7$$

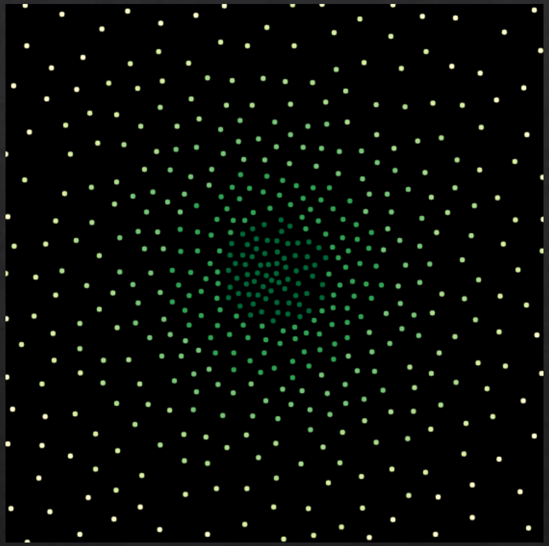
Original Frame

Buffer



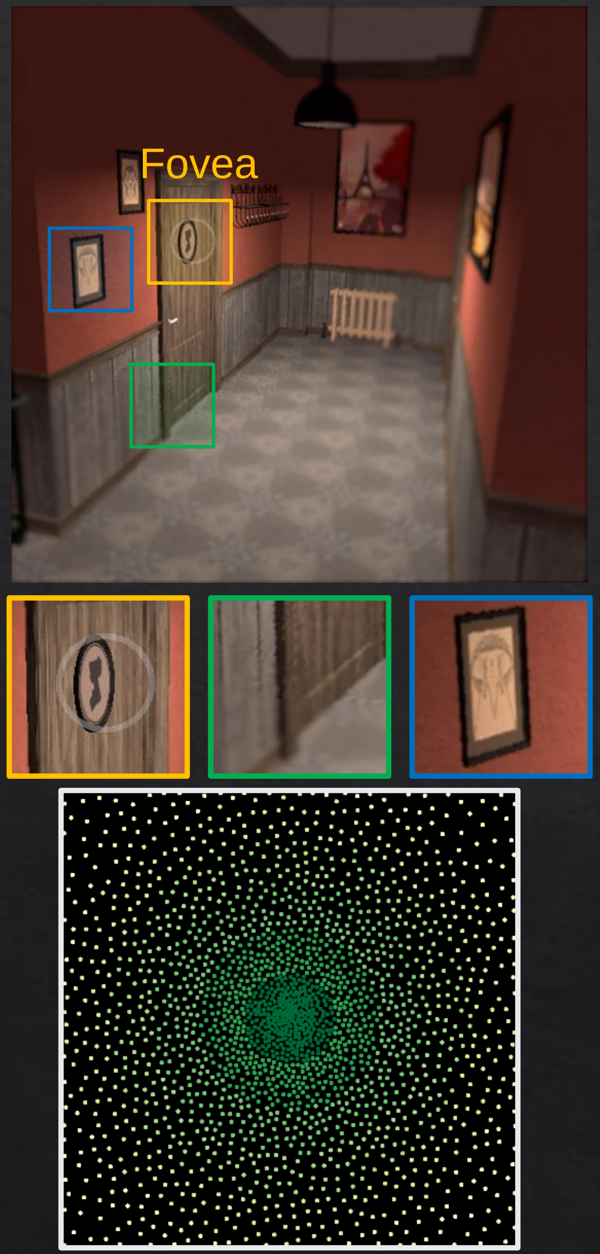
Screen

Sample Map



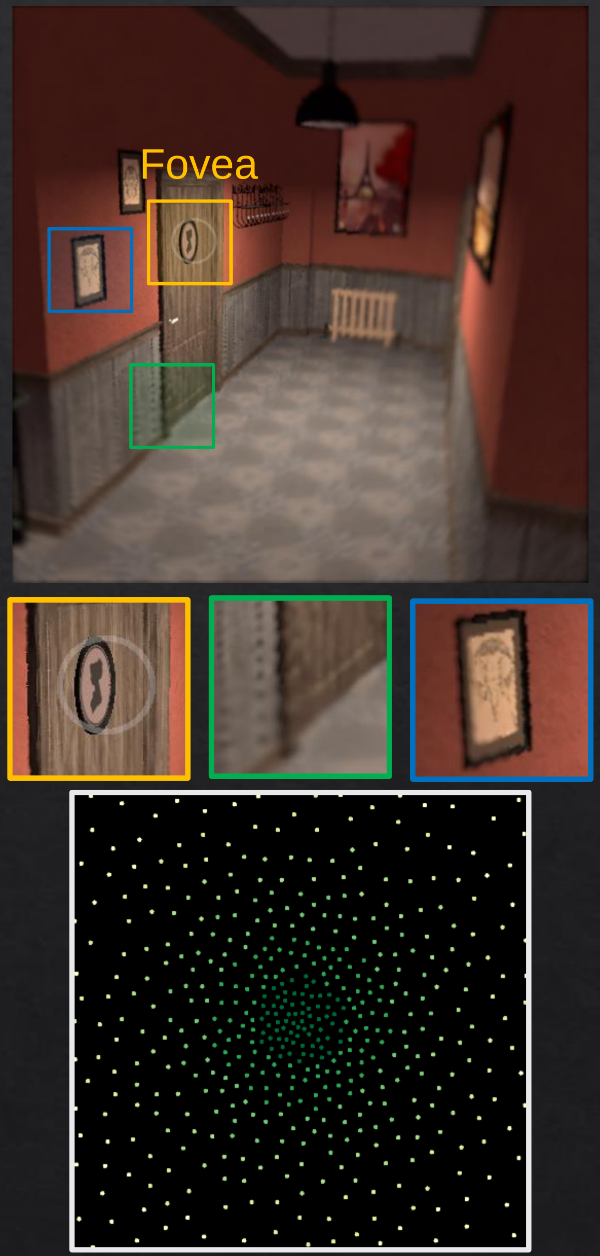
$$\sigma = 2.4$$

$\sigma = 1.2$



$\sigma = 1.7$

$\sigma = 2.4$



kernel function parameter

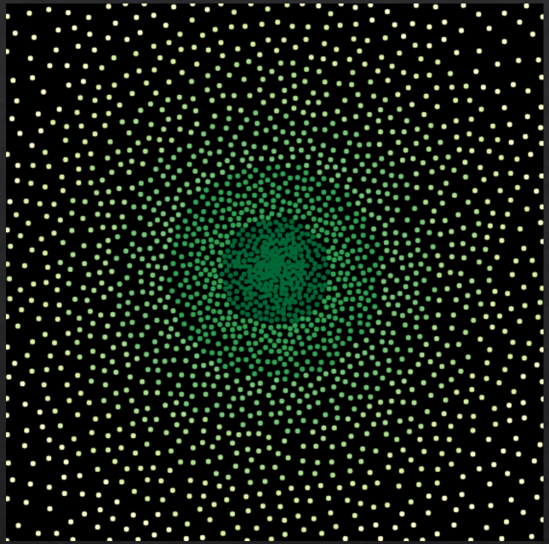
α

Original Frame

Buffer

Screen

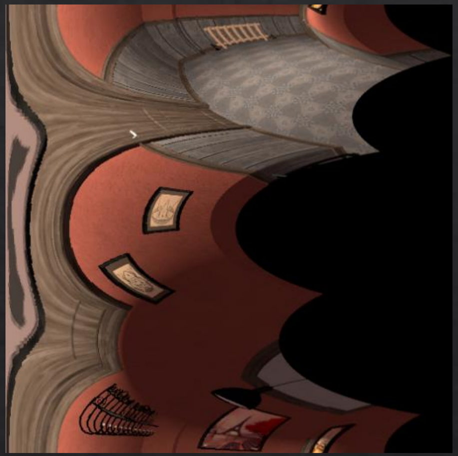
Sample Map



$$\alpha = 1$$

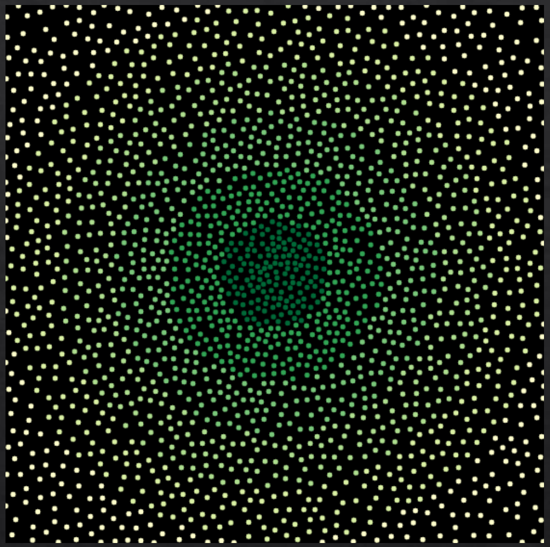
Original Frame

Buffer



Screen

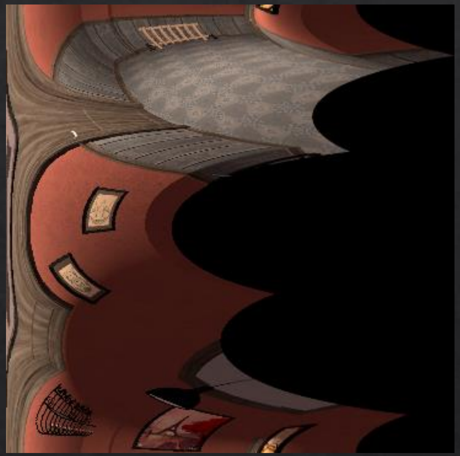
Sample Map



$$\alpha = 4$$

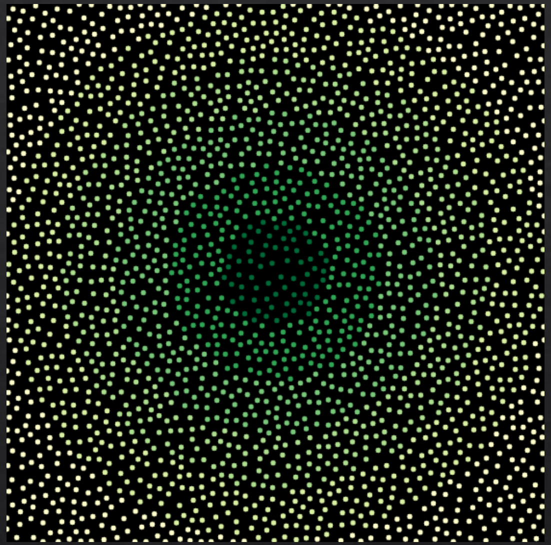
Original Frame

Buffer



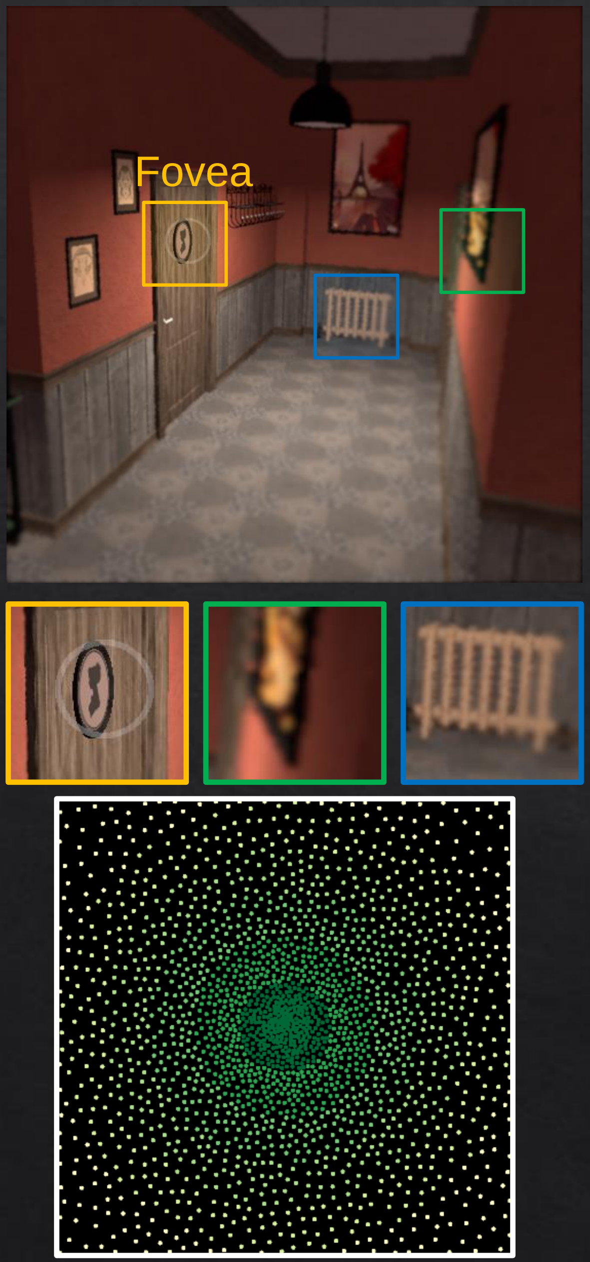
Screen

Sample Map

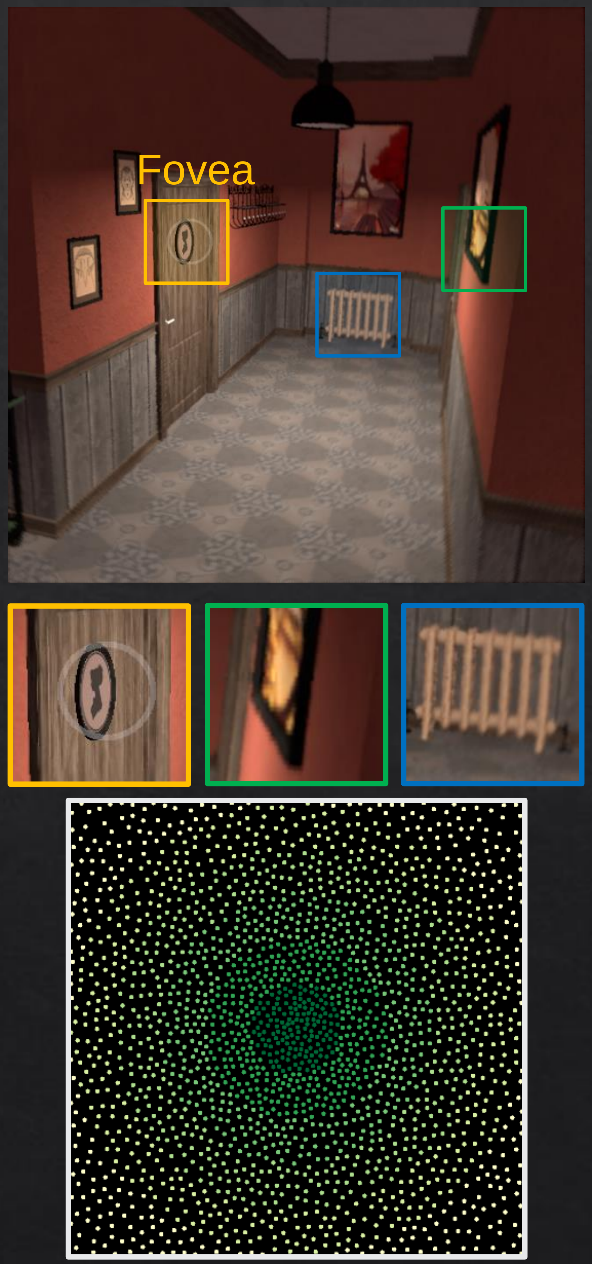


$$\alpha = 6$$

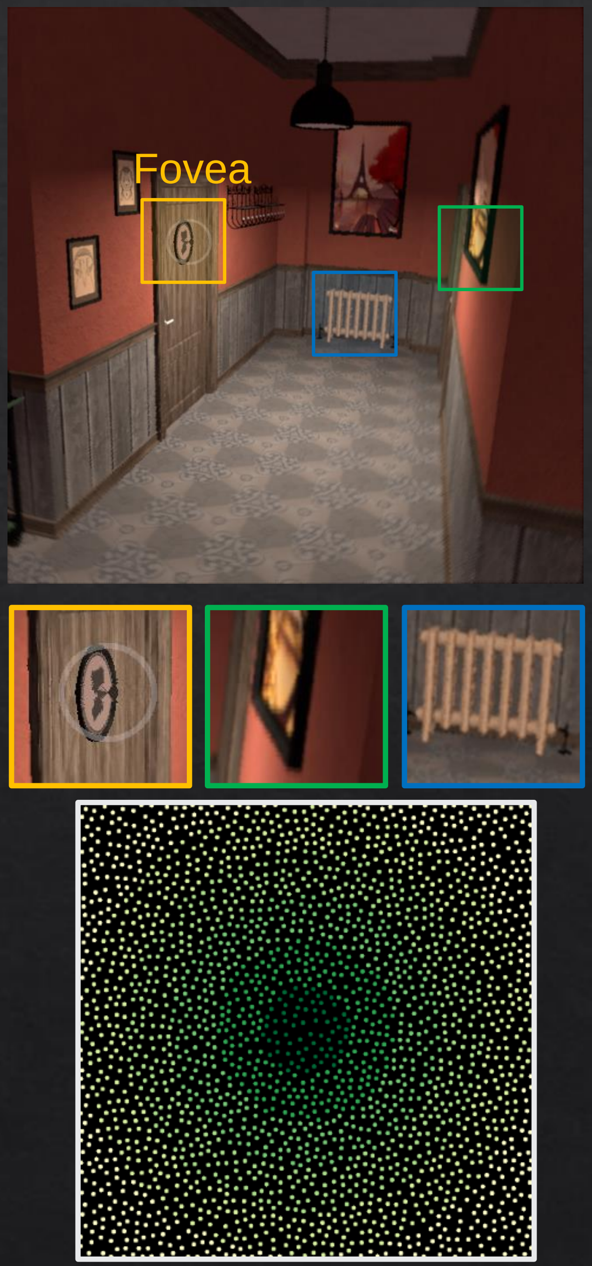
$\alpha = 1$



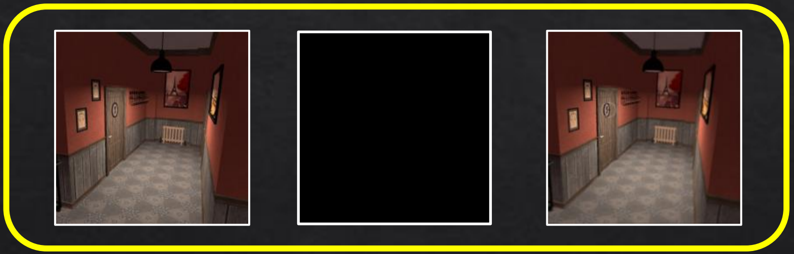
$\alpha = 4$

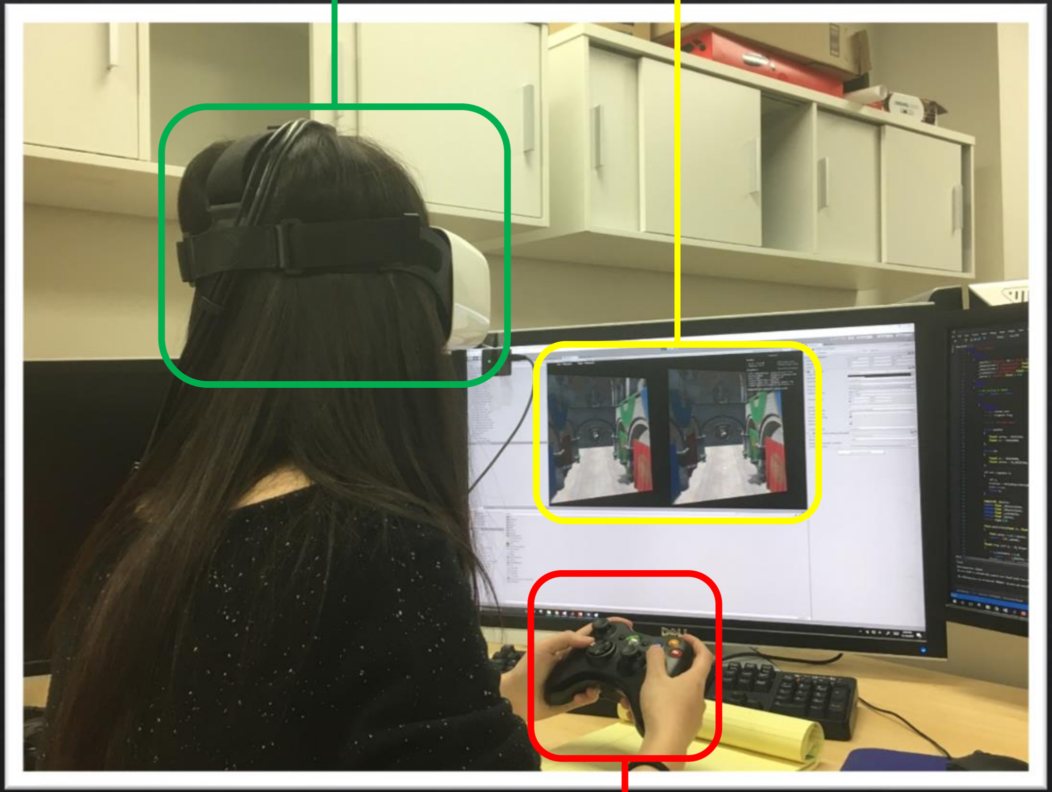


$\alpha = 6$



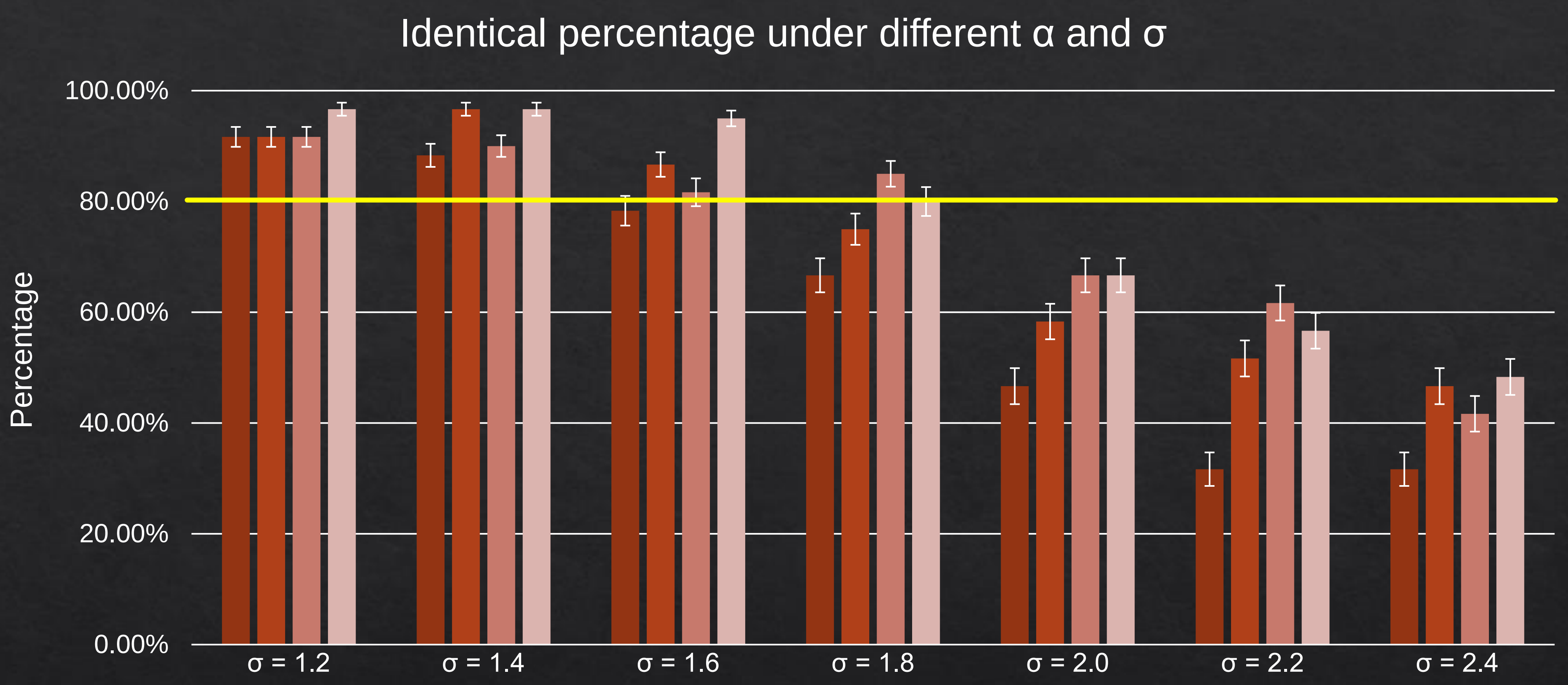
User Study





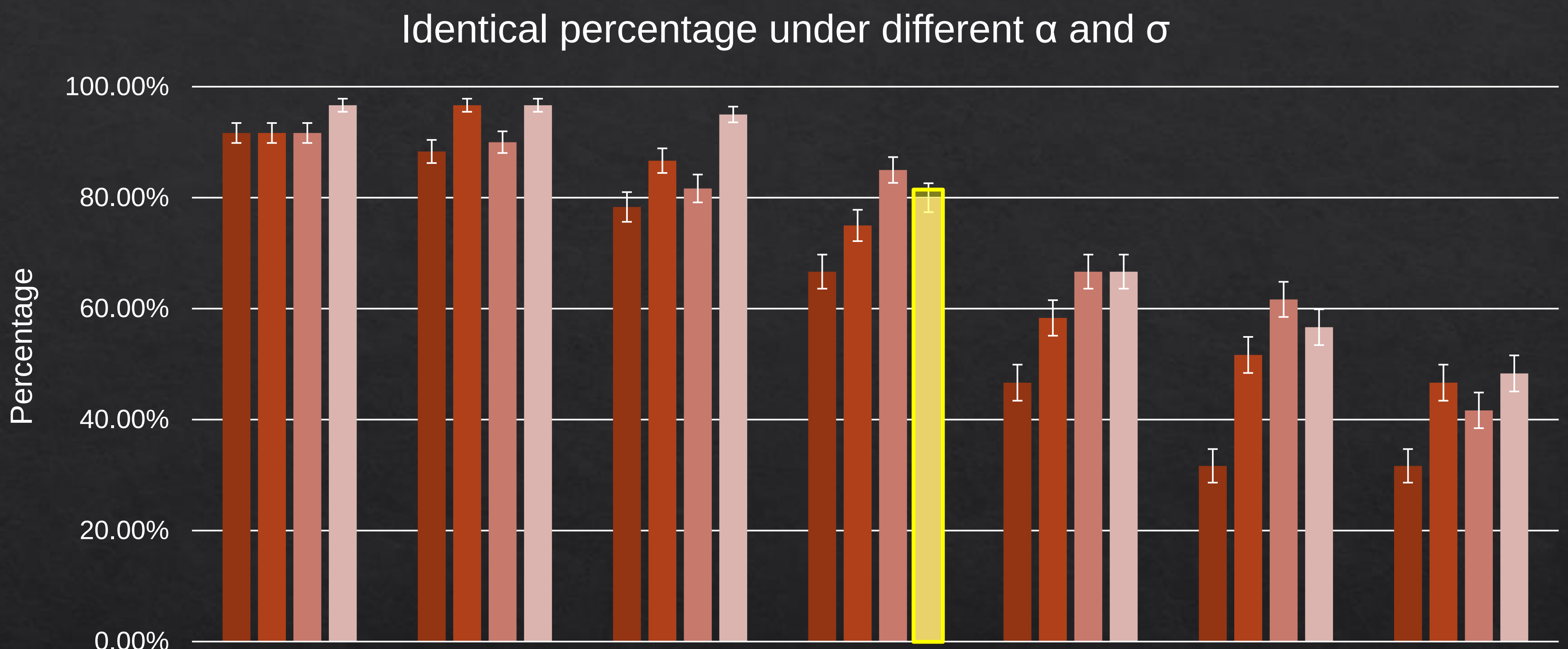
Resolution: 2560×1440
Field of view: up to 100 degrees

$\sigma \in [1.2, 2.4]$ step size: 0.2
$\alpha \in [1, 4]$ step size: 1.0



	σ = 1.2	σ = 1.4	σ = 1.6	σ = 1.8	σ = 2.0	σ = 2.2	σ = 2.4
α = 1	91.67%	88.33%	78.33%	66.67%	46.67%	31.67%	31.67%
α = 2	91.67%	96.67%	86.67%	75.00%	58.33%	51.67%	46.67%
α = 3	91.67%	90.00%	81.67%	85.00%	66.67%	61.67%	41.67%
α = 4	96.67%	96.67%	95.00%	80.00%	66.67%	56.67%	48.33%

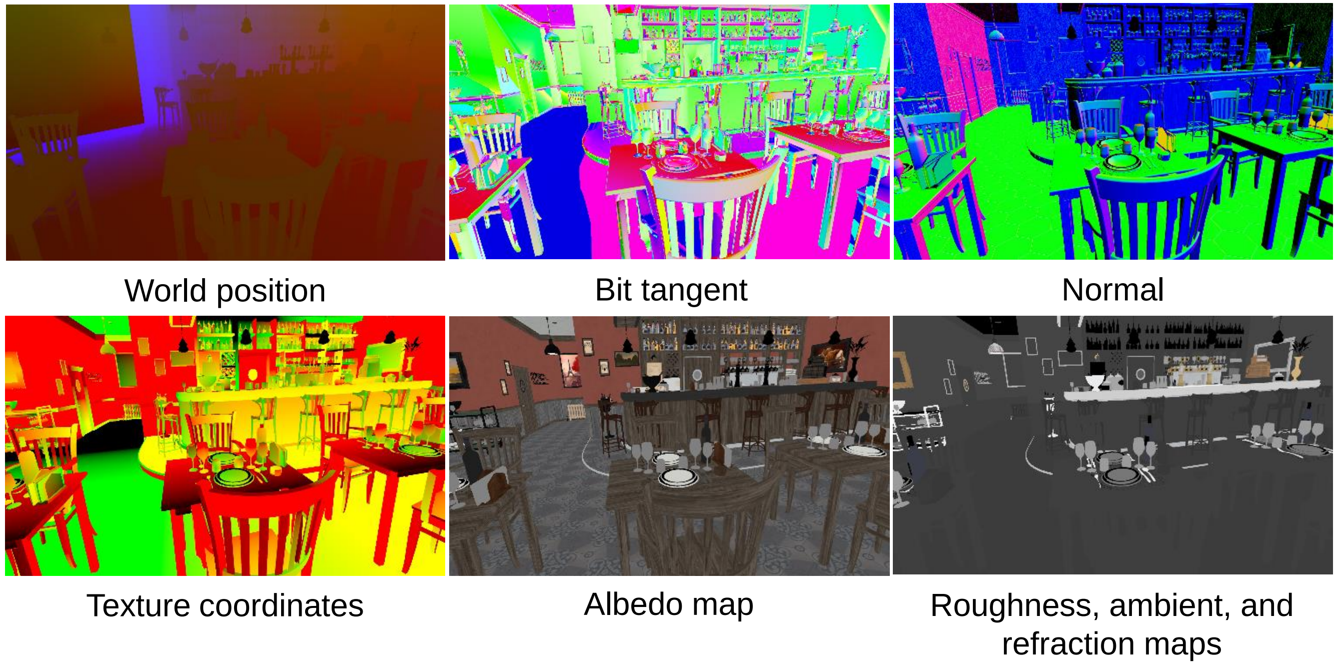
α = 1 α = 2 α = 3 α = 4

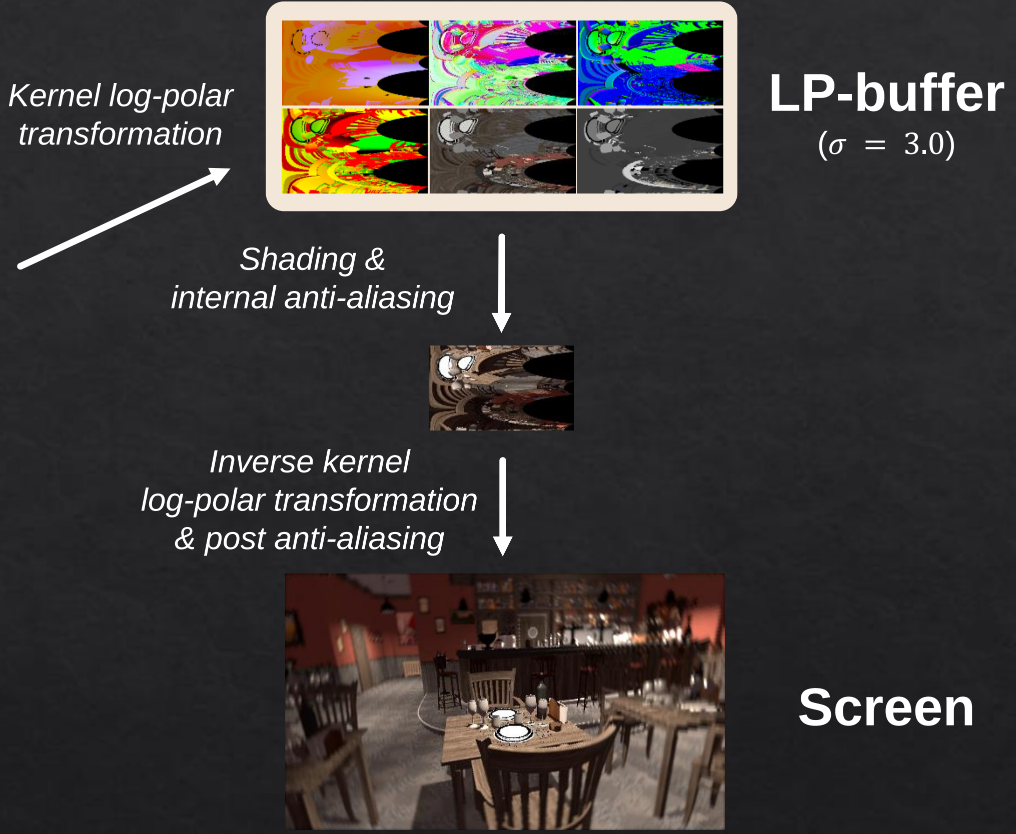


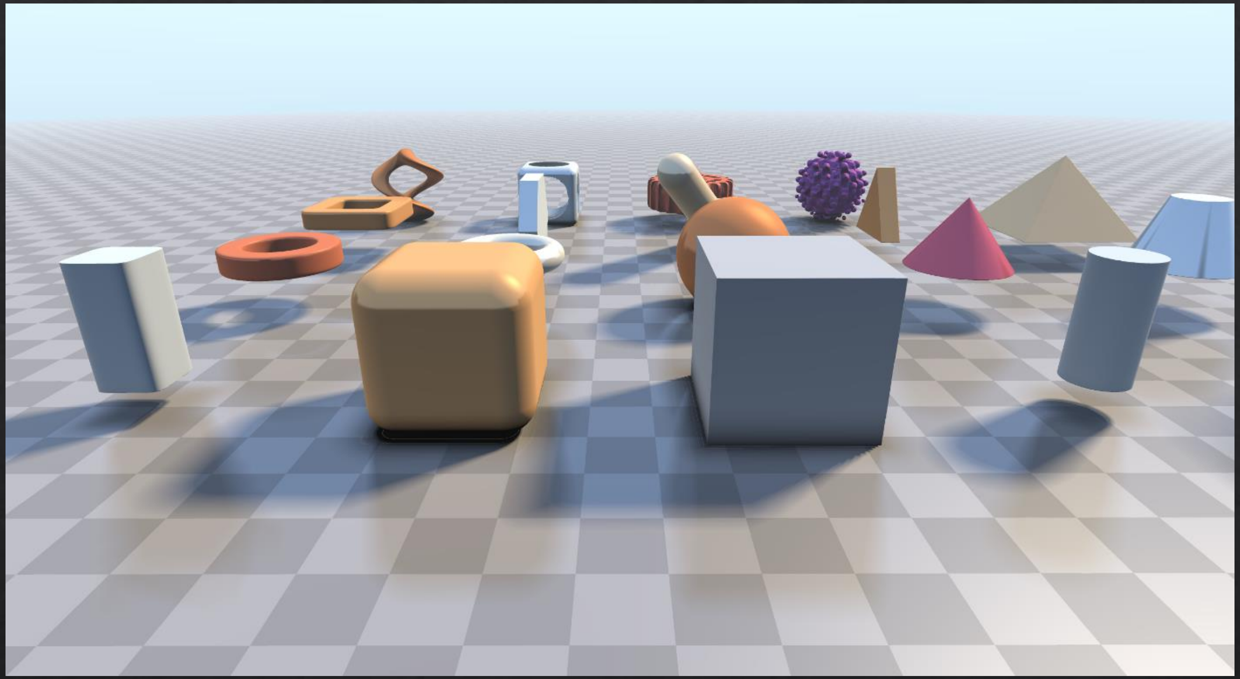
	σ = 1.2	σ = 1.4	σ = 1.6	σ = 1.8	σ = 2.0	σ = 2.2	σ = 2.4
α = 1	91.67%	88.33%	78.33%	66.67%	46.67%	31.67%	31.67%
α = 2	91.67%	96.67%	86.67%	75.00%	58.33%	51.67%	46.67%
α = 3	91.67%	90.00%	81.67%	85.00%	66.67%	61.67%	41.67%
α = 4	96.67%	96.67%	95.00%	80.00%	66.67%	56.67%	48.33%

α = 1 α = 2 α = 3 α = 4

G-buffer


World position
Bit tangent
Normal
Texture coordinates
Albedo map
Roughness, ambient, and refraction maps


Kernel log-polar transformation
LP-buffer
($\sigma = 3.0$)
Shading & internal anti-aliasing
Inverse kernel log-polar transformation & post anti-aliasing
Screen



original ray-marching scene
10 FPS

foveated ray-marching scene (σ = 1.8, α = 4)
30 FPS

* Scene created by Íñigo Quílez.

original 3D geometries
31 FPS

foveated 3D geometries ($\sigma = 1.8$, $\alpha = 4$)
67 FPS

Scene	3D Textured Meshes			Ray Casting		
Resolution	Ground Truth	Foveated	Speed up	Ground Truth	Foveated	Speed up
1920×1080	55 FPS	110 FPS	2.0X	20 FPS	57 FPS	2.9X
2560×1440	31 FPS	67 FPS	2.2X	10 FPS	30 FPS	3.0X
3840×2160	8 FPS	23 FPS	2.8X	5 FPS	16 FPS	3.2X

Summary

- Kernel log-polar transformation for 3D graphics
  - Deferred Shading
  - Parameterize with kernel parameter α and buffer parameter σ
- User study
  - Determine parameters to maximize perceptual realism and minimize computation
- Experiment
  - 2.8X – 3.2X speedup

Ground Truth

Kernel Foveated Rendering

Thanks!

video

paper

Kernel Foveated Rendering

Xiaoxu Meng, Ruofei Du, Matthias Zwicker and Amitabh Varshney

Augmentarium | UMIACS

University of Maryland, College Park

ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games 2018

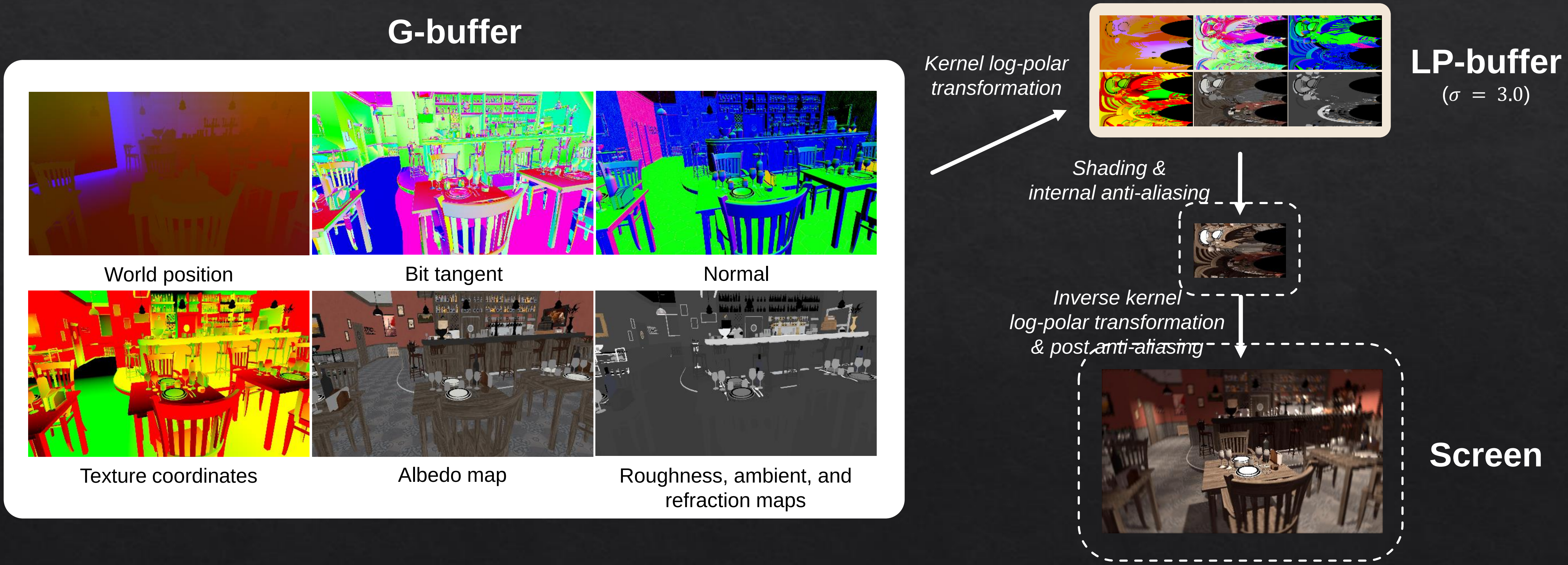
FOVE Headset

- DISPLAY
  - WQHD OLED (2560 X 1440)
  - Frame rate: 70fps
  - Field of view: Up to 100 degrees
- EYE TRACKING SENSORS
  - Infrared eye tracking system x 2
  - Tracking accuracy: less than 1 degree
  - Frame rate: 120fps

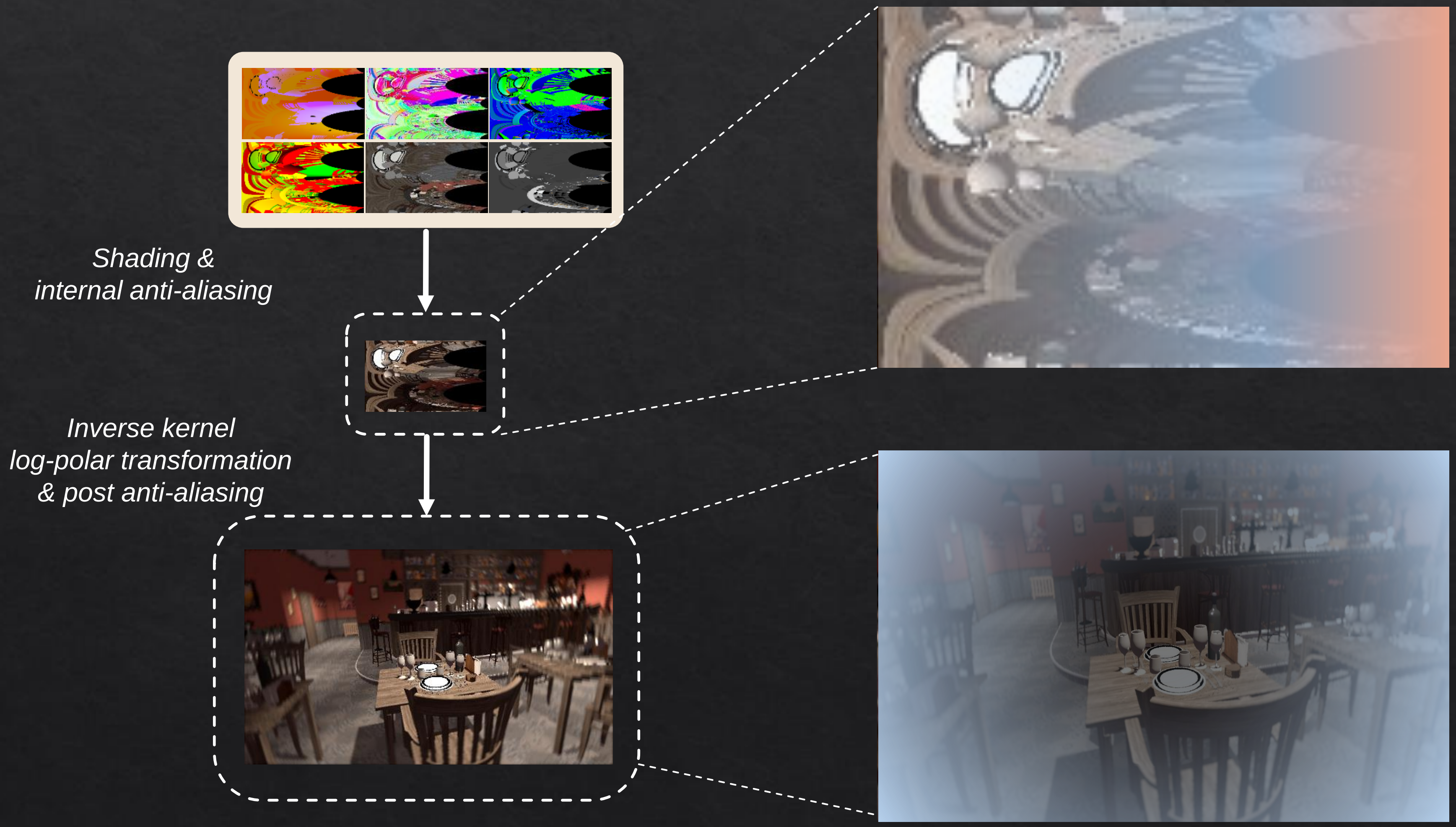
User Study: Significance

σ^2 –value	1.2	1.4	1.6	1.8	2.0	2.2	2.4
Cochran's Q value	1.72	5.76	8.20	8.25	7.49	14.27	5.48
p-value	0.631	0.122	0.042	0.041	0.058	0.002	0.139

Two-level Anti-aliasing



Two-level Anti-aliasing



Non-uniform Gaussian Blur

Kernel size increase from left (fovea) to right (periphery)

Non-uniform Gaussian Blur

Kernel size increase from fovea to periphery

Video & Paper

video

paper