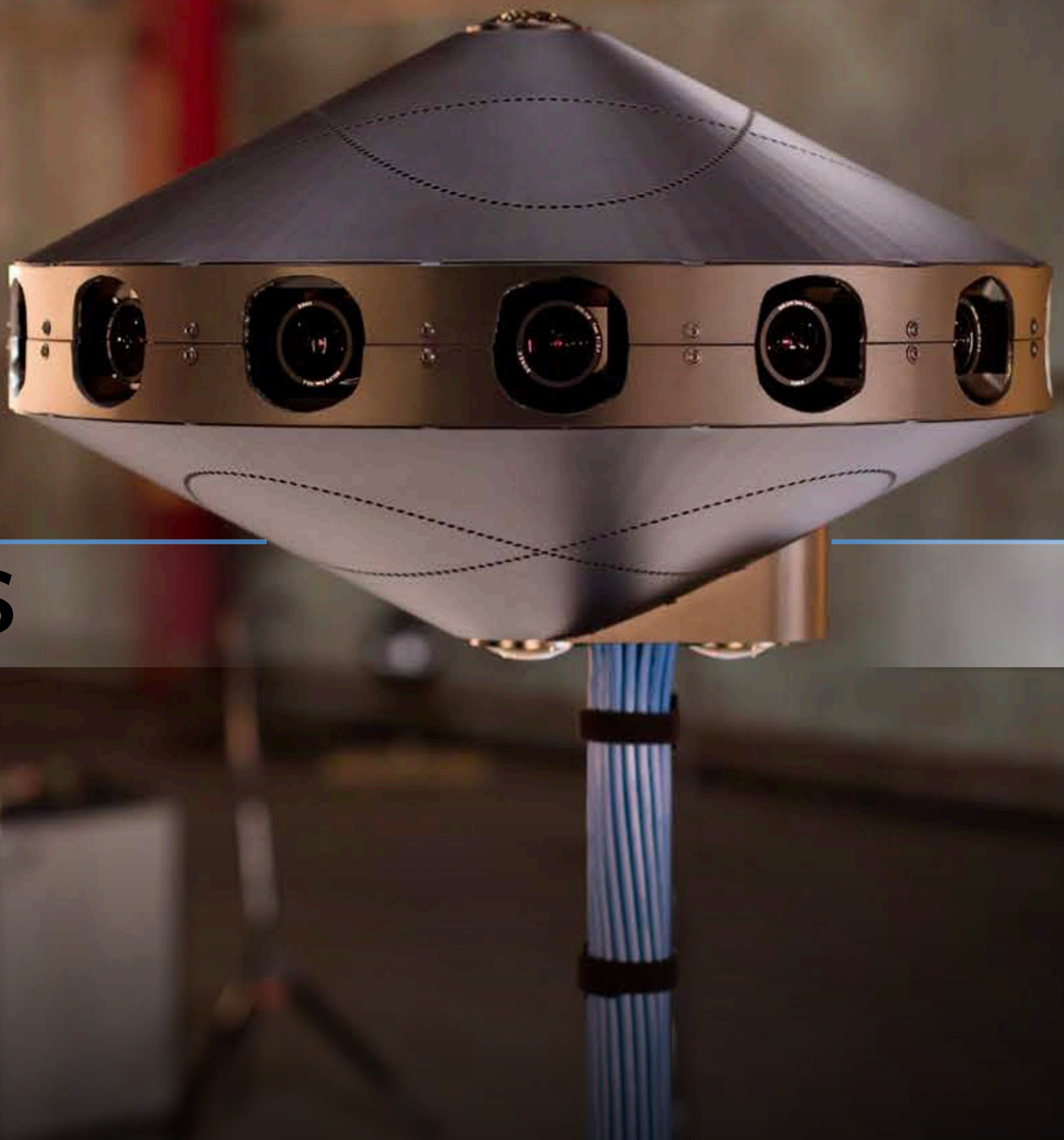
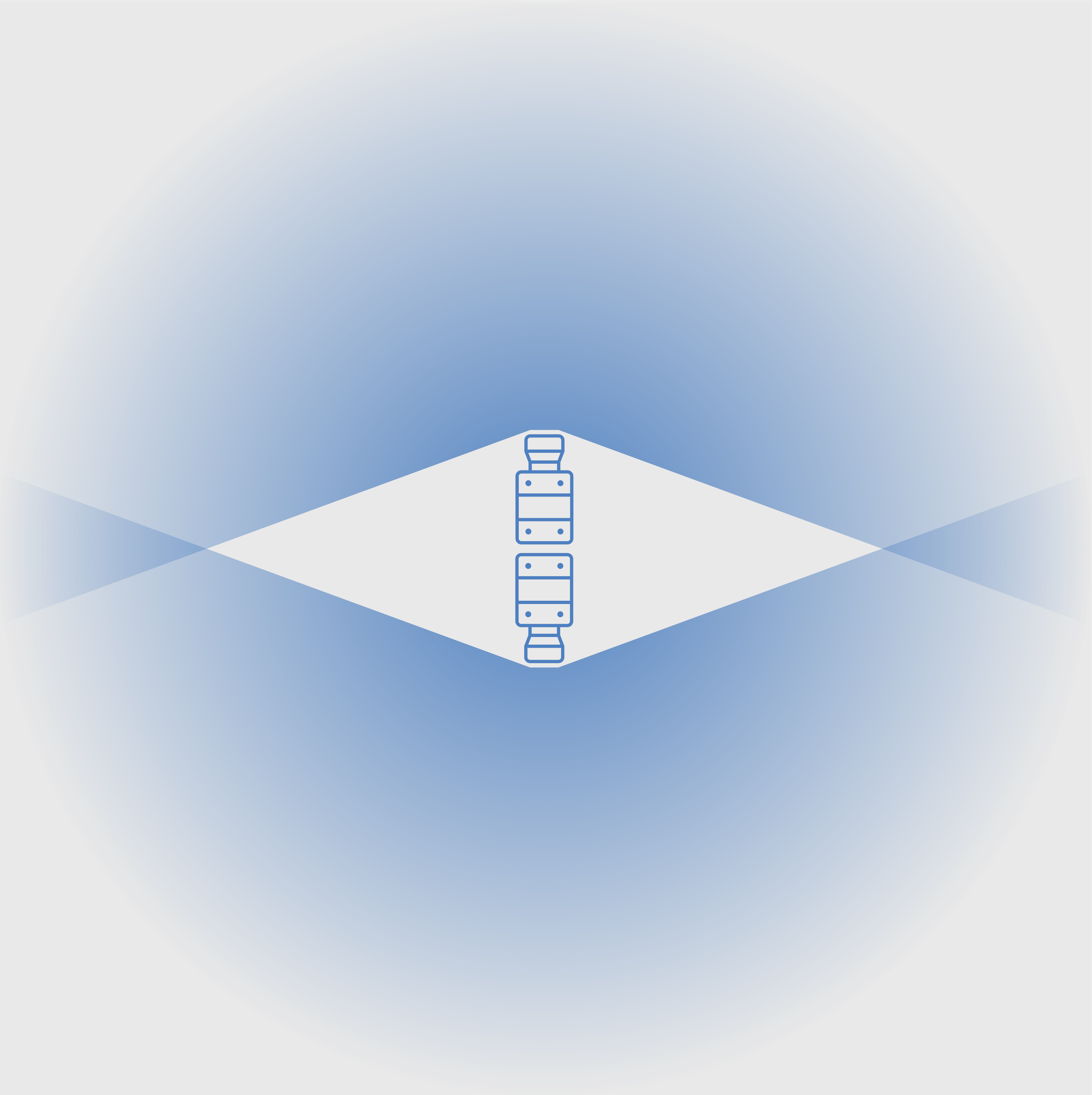
360 and ODS

VIDEO CAPTURE SYSTEMS

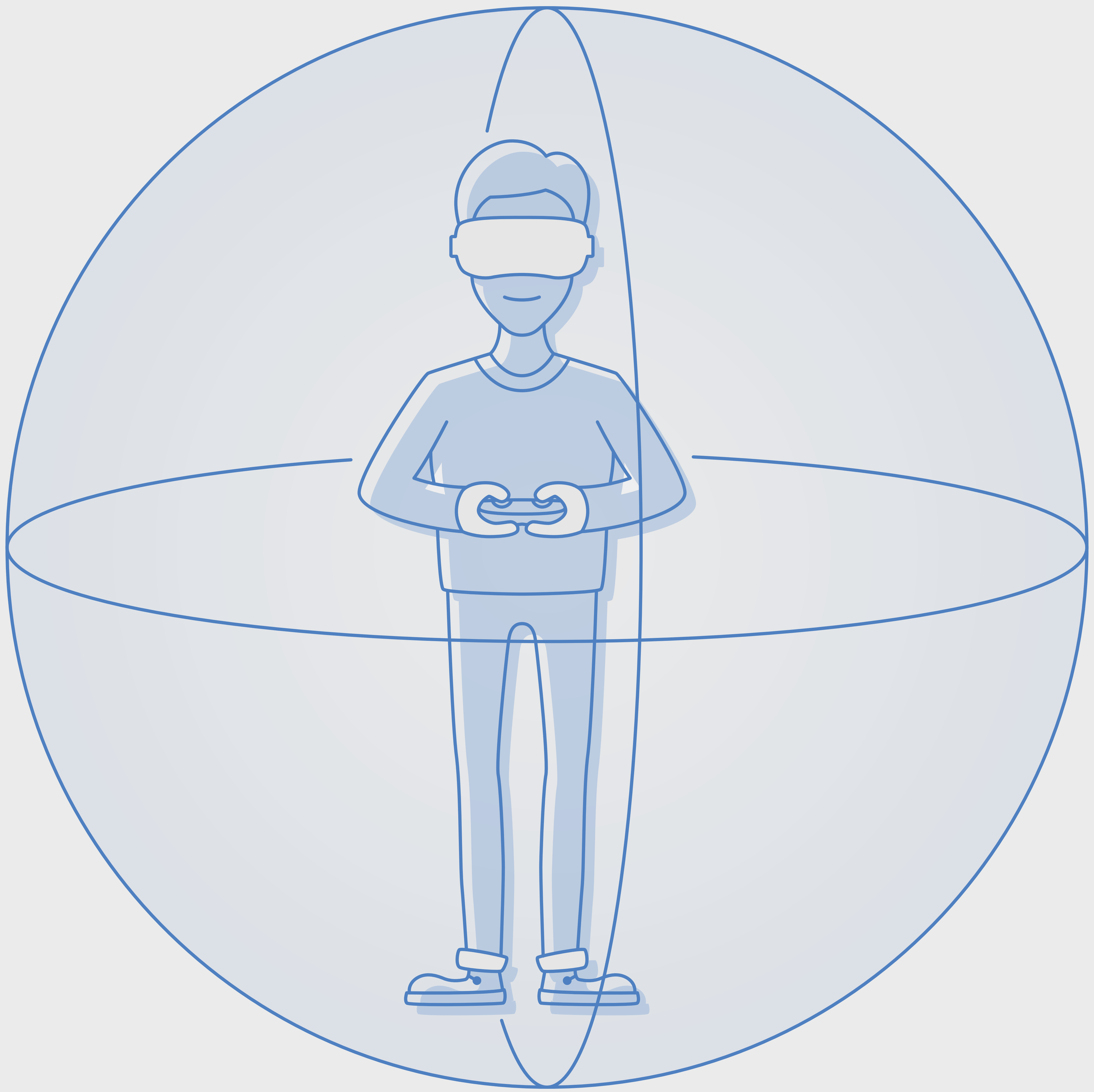
Brian Cabral facebook



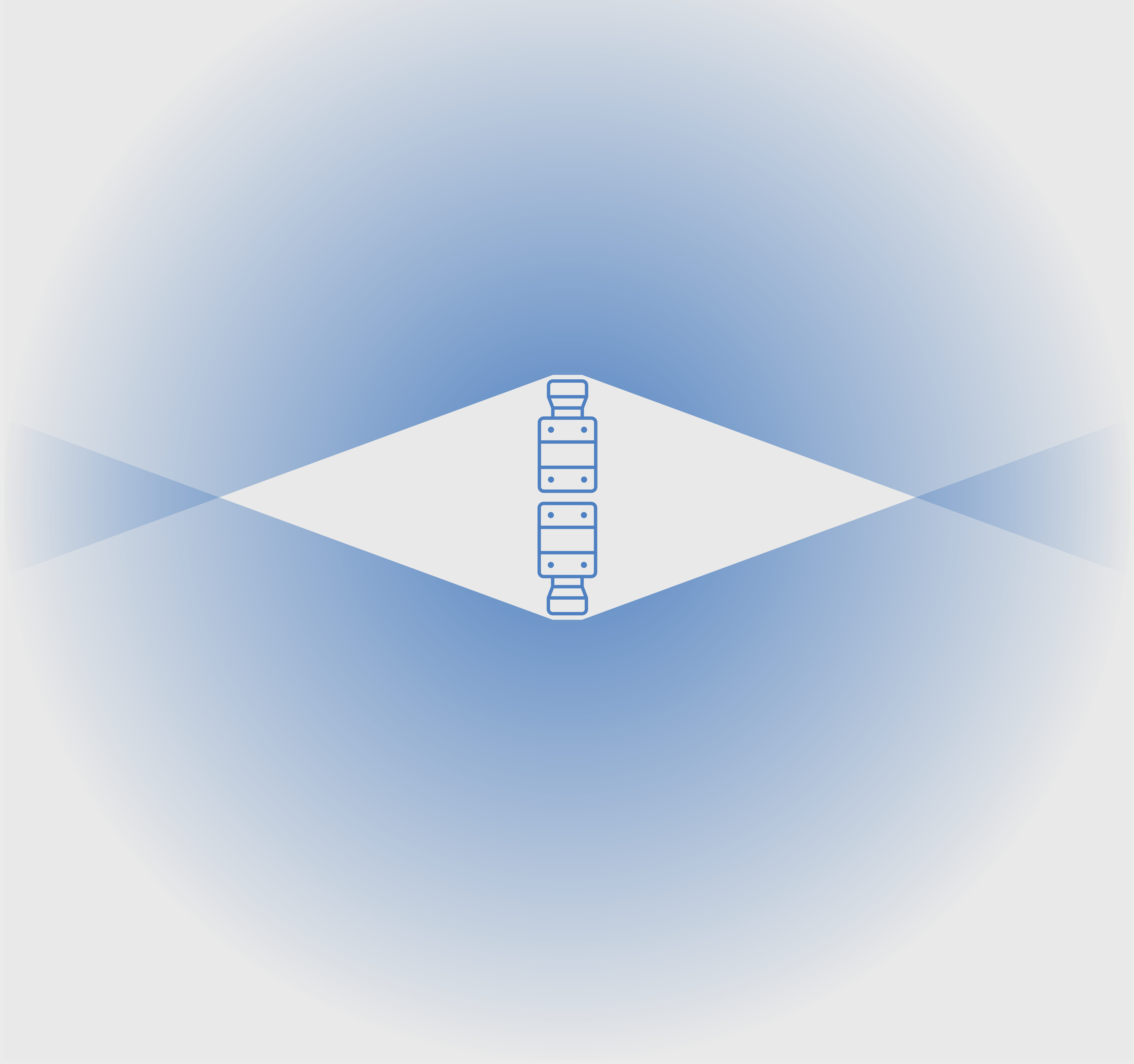
Monoscopic 360° capture



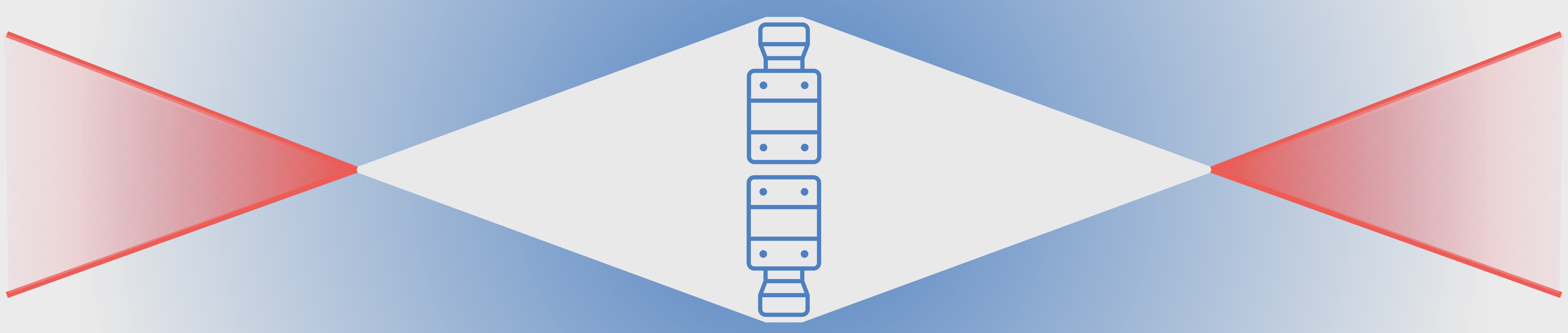
Monoscopic 360° viewing



Assumption of infinity

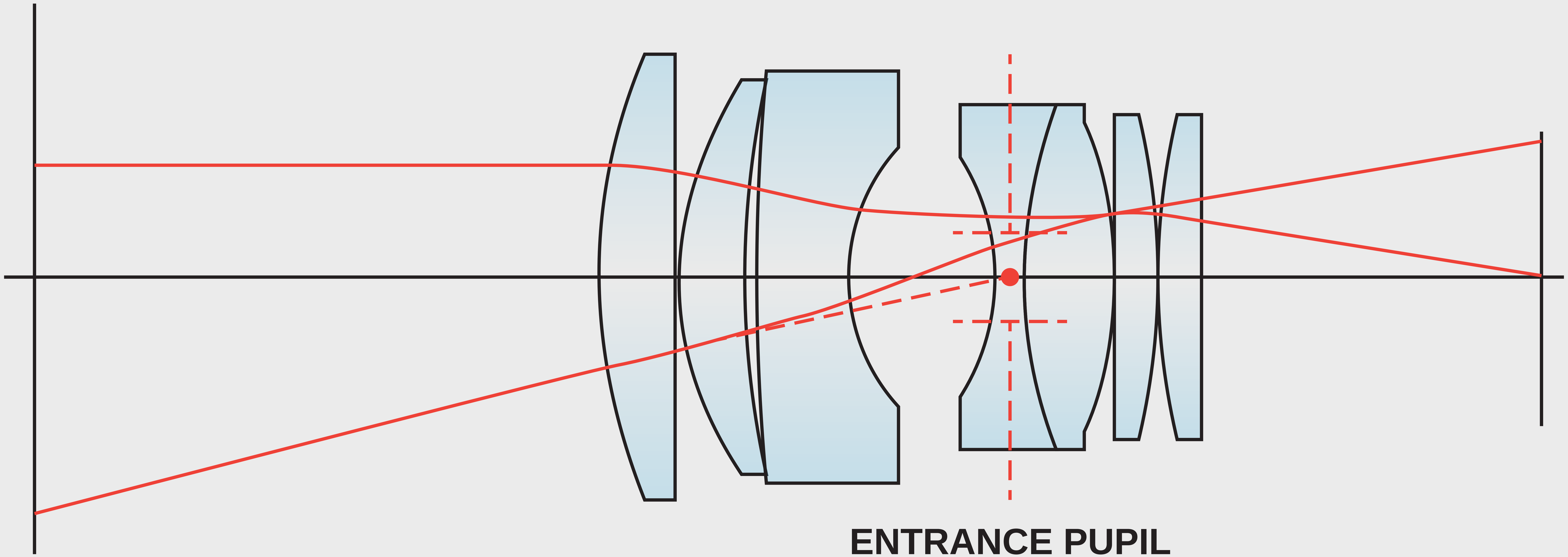


Assumption of infinity



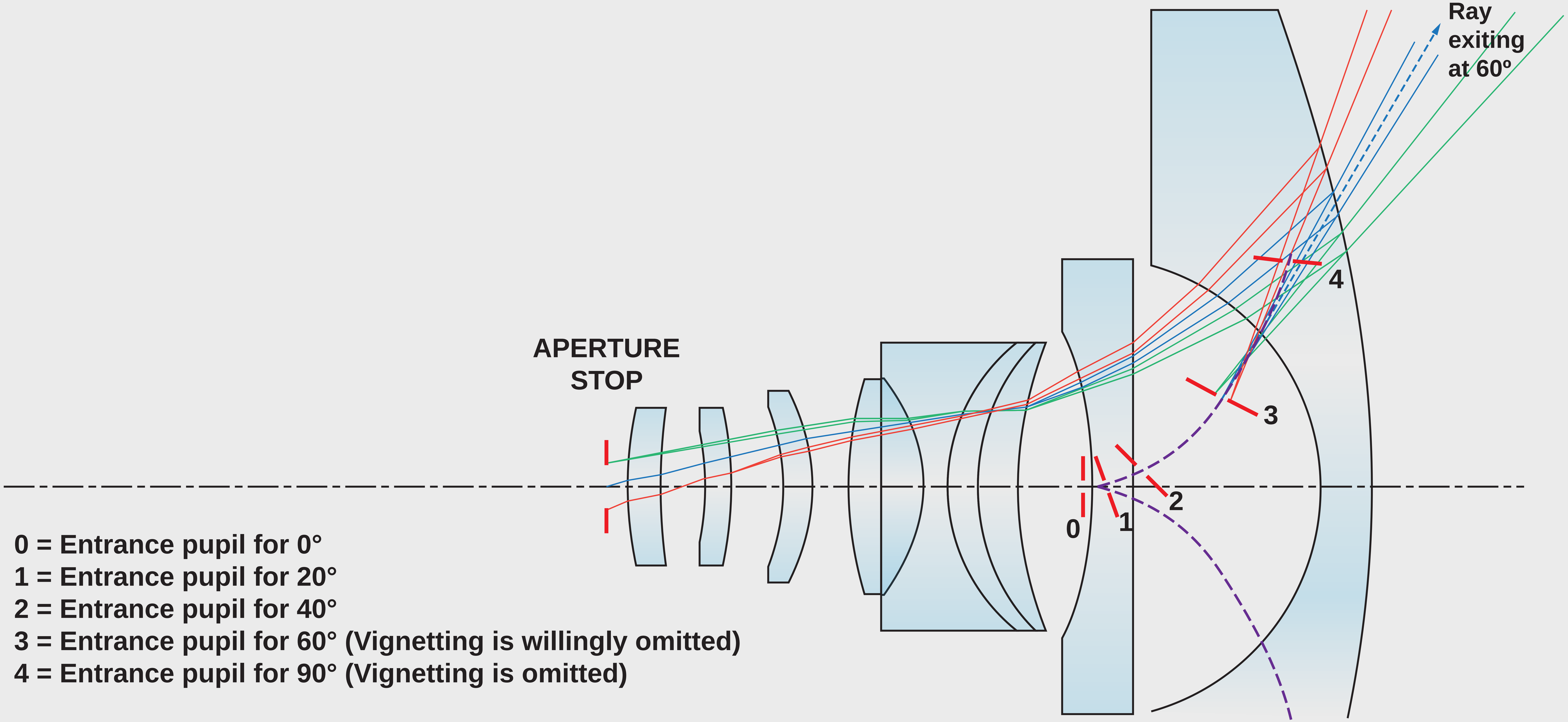
Regions of overlap have parallax
near and within hyperfocal distance

The entrance pupil the point of zero parallax



f $\tan(\theta)$ lenses have single entrance pupil

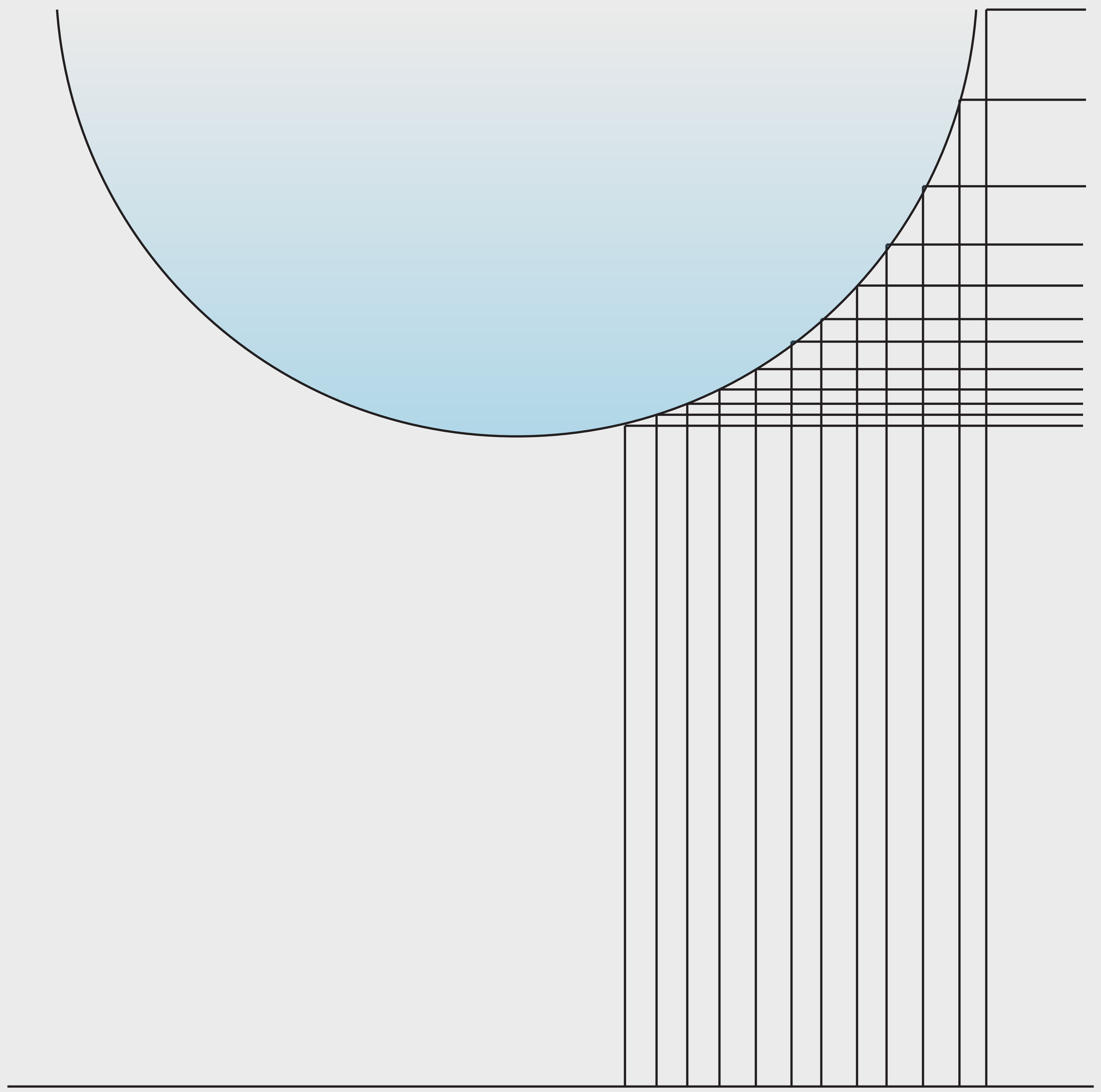
The entrance pupil Fisheye lenses



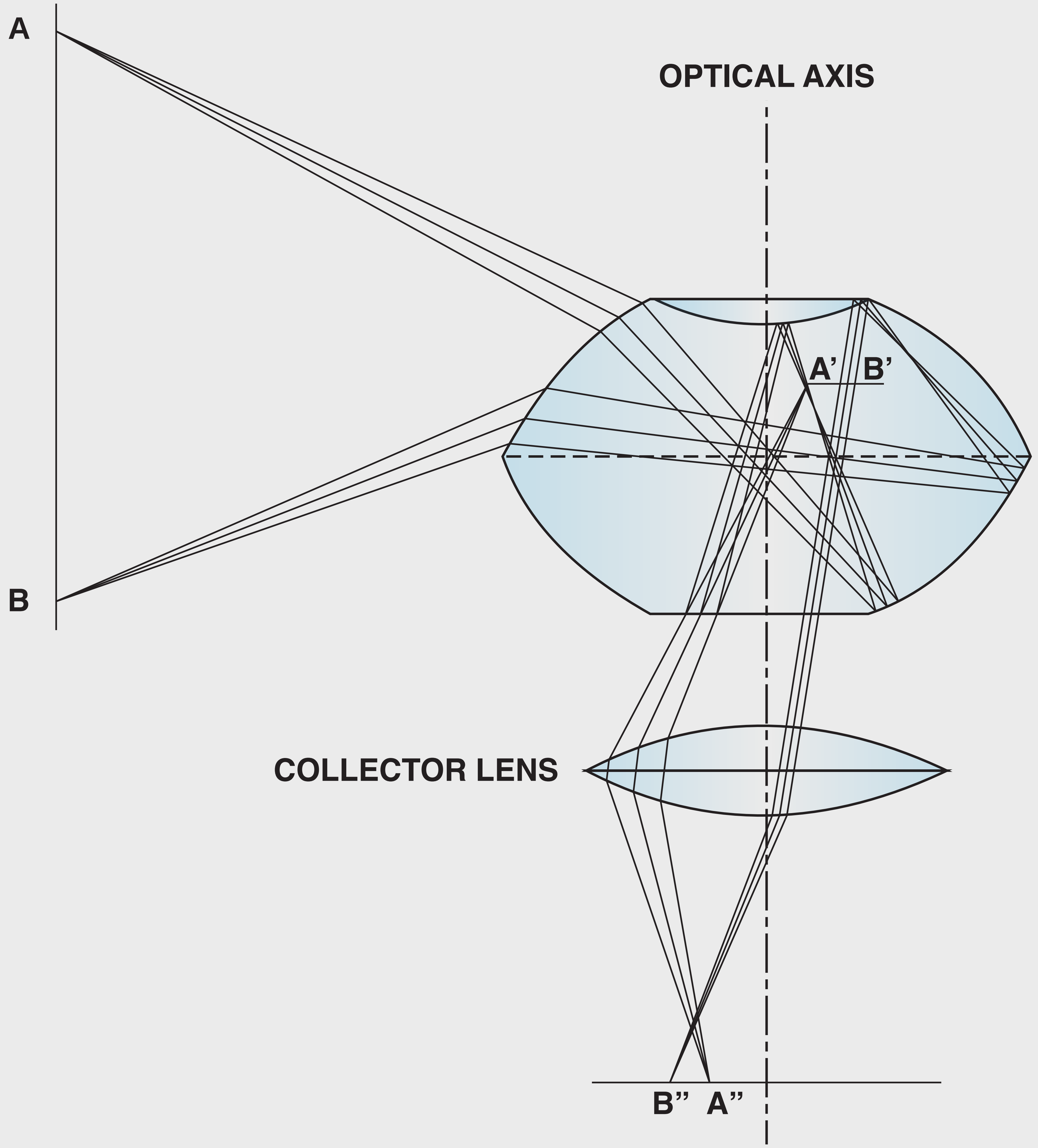
0 = Entrance pupil for 0°
1 = Entrance pupil for 20°
2 = Entrance pupil for 40°
3 = Entrance pupil for 60° (Vignetting is willingly omitted)
4 = Entrance pupil for 90° (Vignetting is omitted)

f θ lenses have <u>no</u> single entrance pupil

Catadioptric



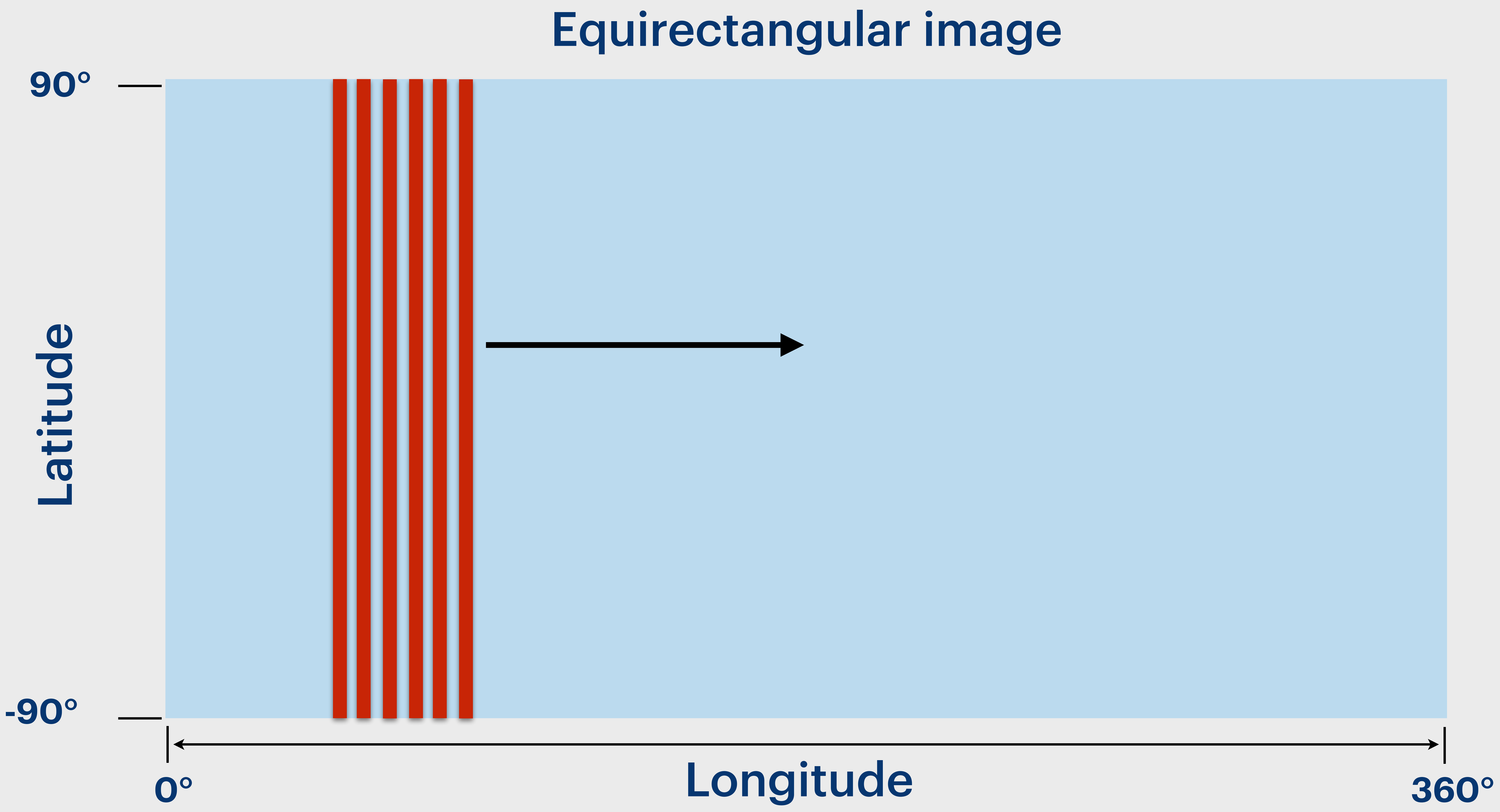
Panoramic Annular Optics



These are really cylindrical capture but no stitch lines!

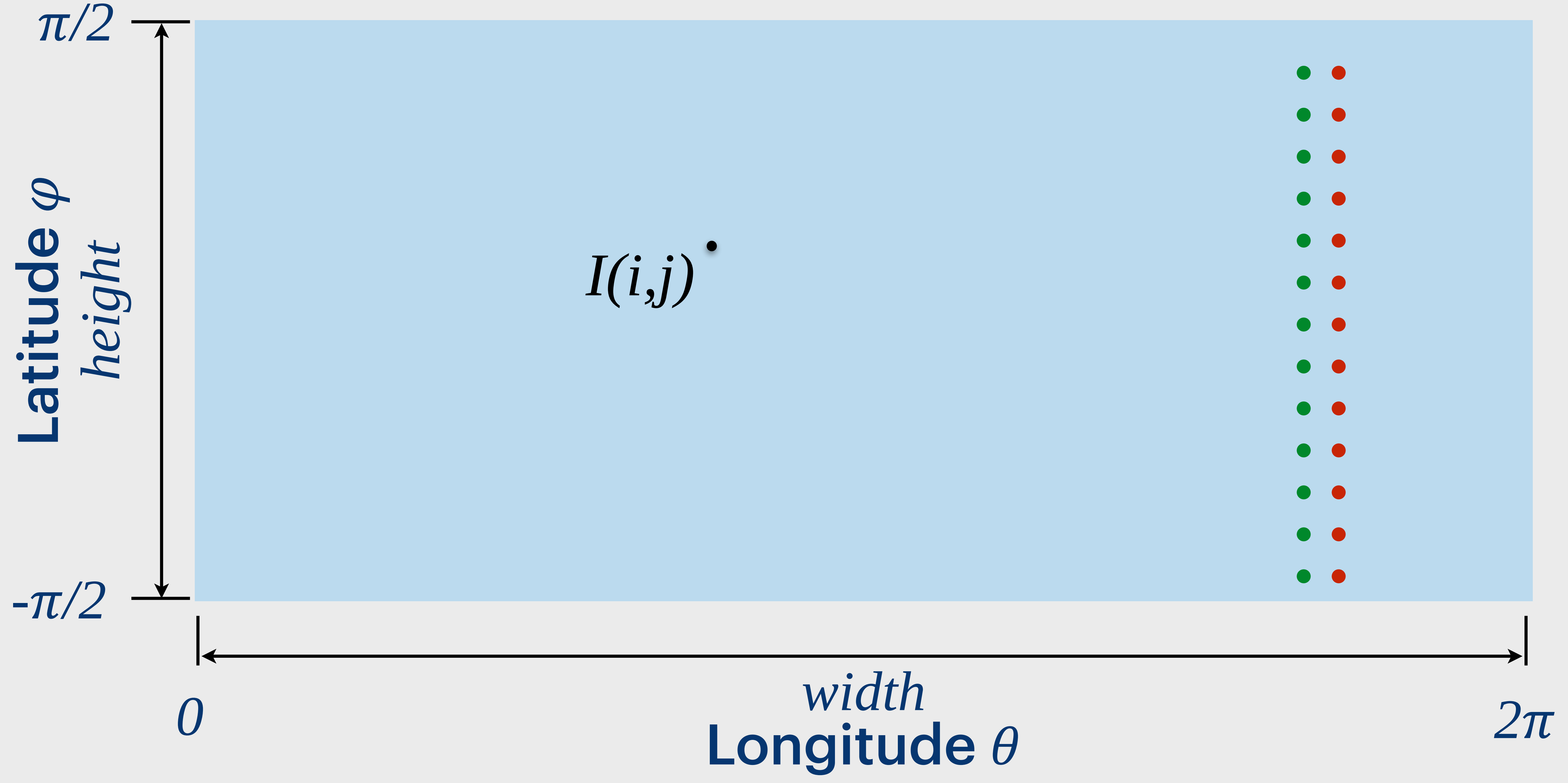
The slit-scan camera model

Another way to create a 360° image

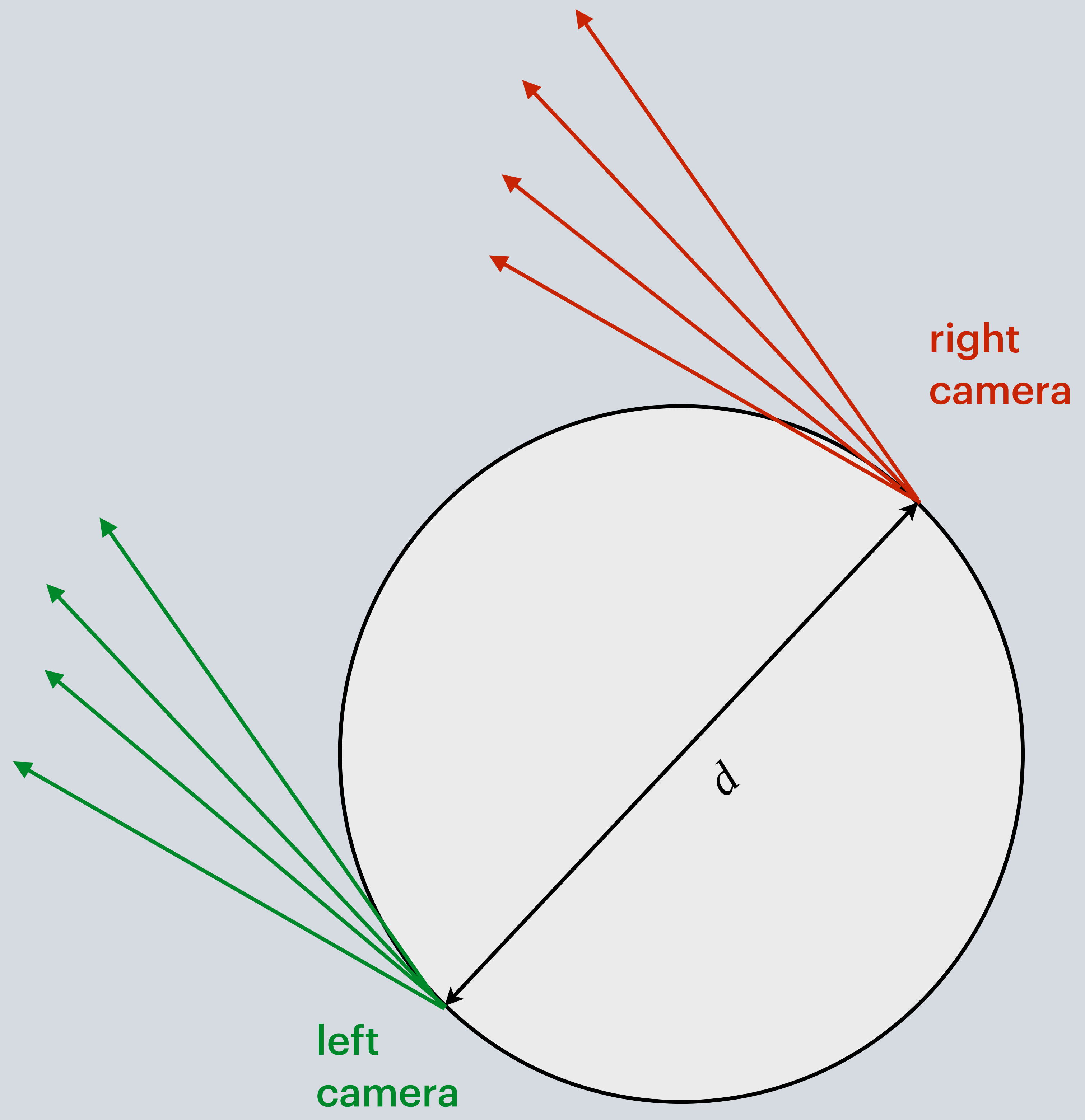


Omni-Directional Stereo (ODS)

Do slit photography for each eye



$$\varphi = \pi\ i\ /\ height - \pi/2$$
$$\theta = 2\pi\ \mathrm{j}\ /\ width$$



where p is the inter ocular distance, $r = d/2$

Left Eye Rays

$l.x = \sin(\varphi) * \cos(\theta) - r*\sin(\theta)$

$l.y = \sin(\varphi) * \sin(\theta) - r*\cos(\theta)$

$l.z = \cos(\varphi)$

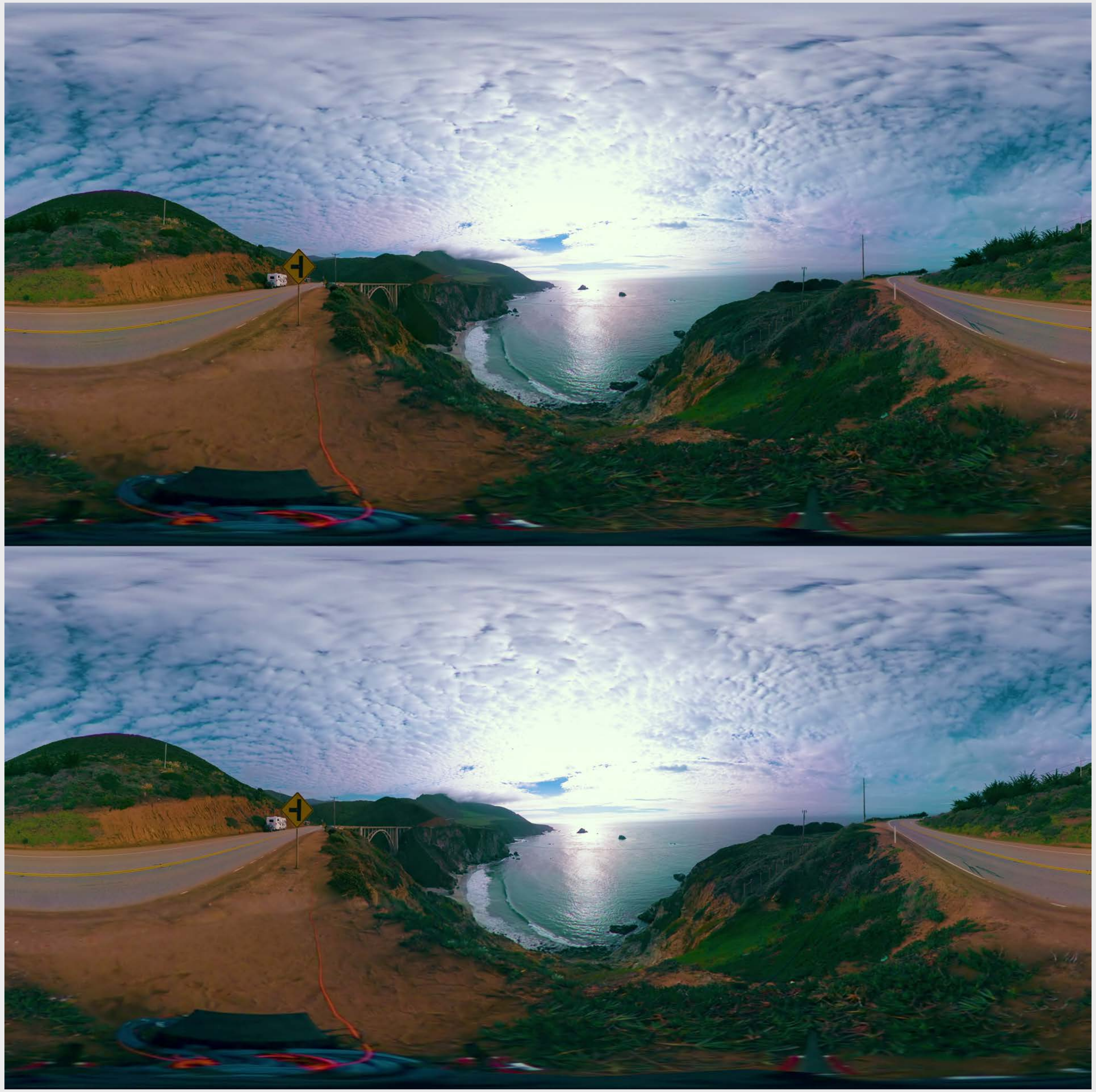
Right Eye Rays

$r.x = \sin(\varphi) * \cos(\theta) + r*\sin(\theta)$

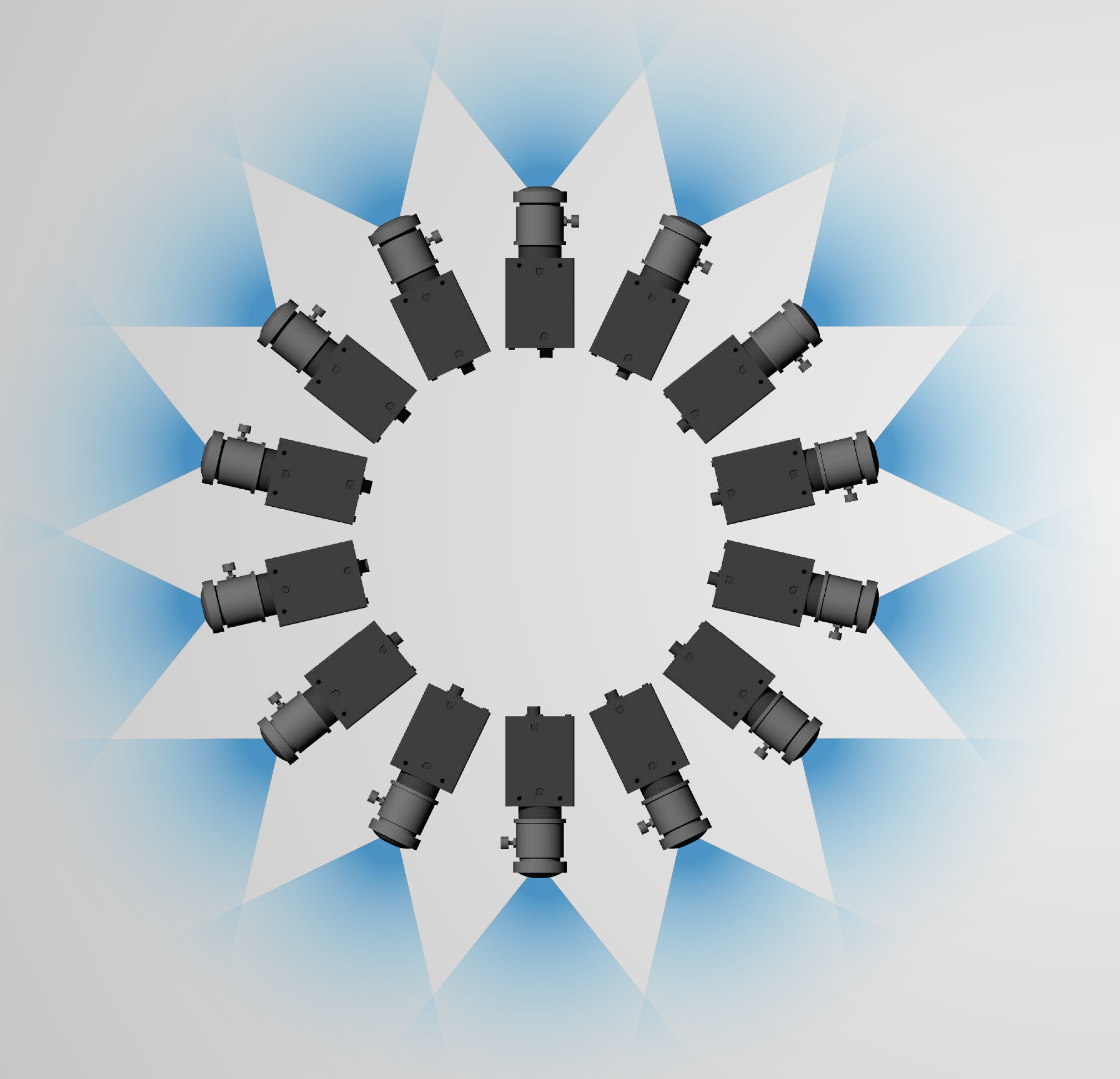
$r.y = \sin(\varphi) * \sin(\theta) + r*\cos(\theta)$

$r.z = \cos(\varphi)$

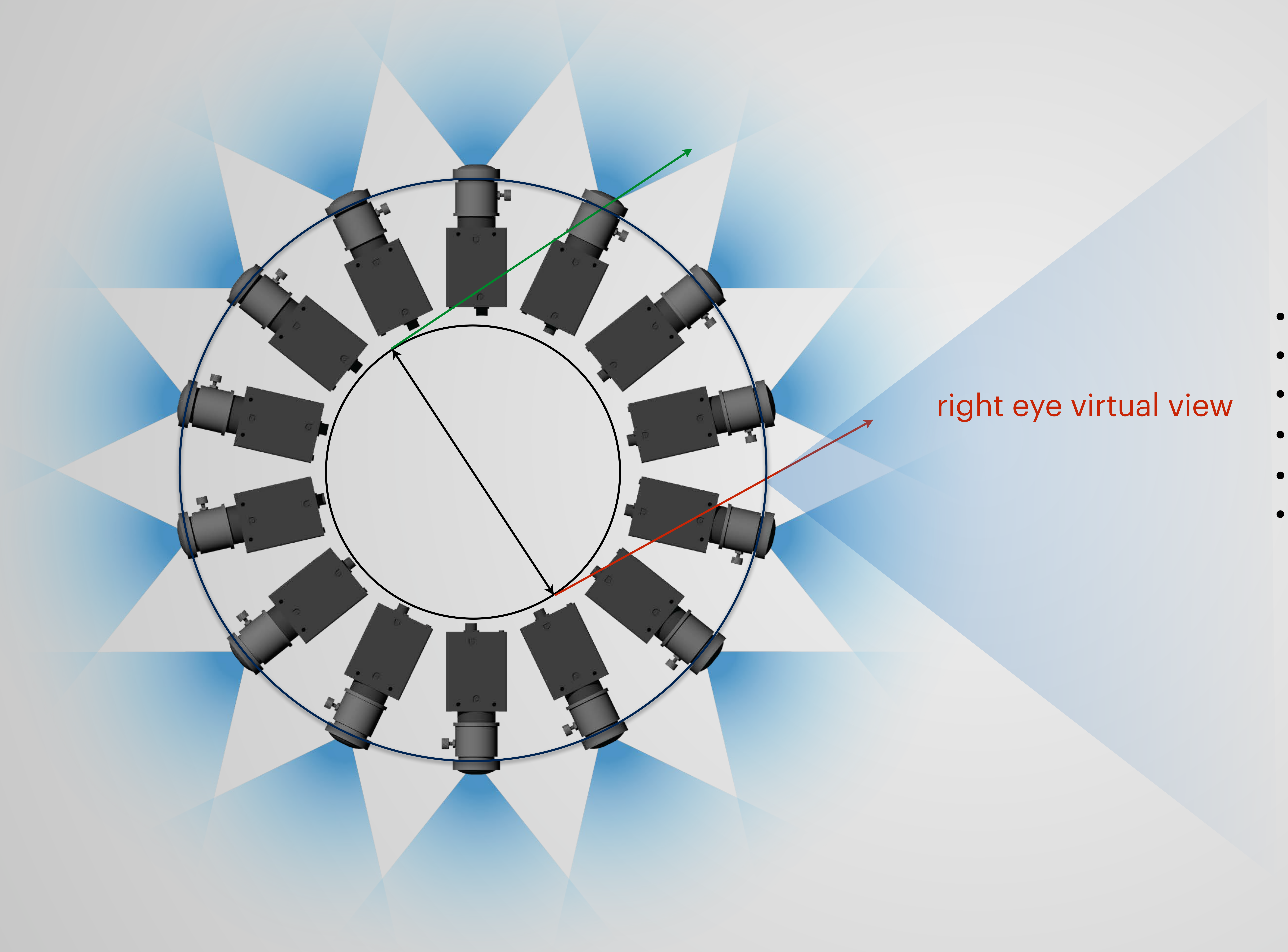
Left-right, top-bottom ODS stereo pair



Creating ODS with a fixed array of cameras

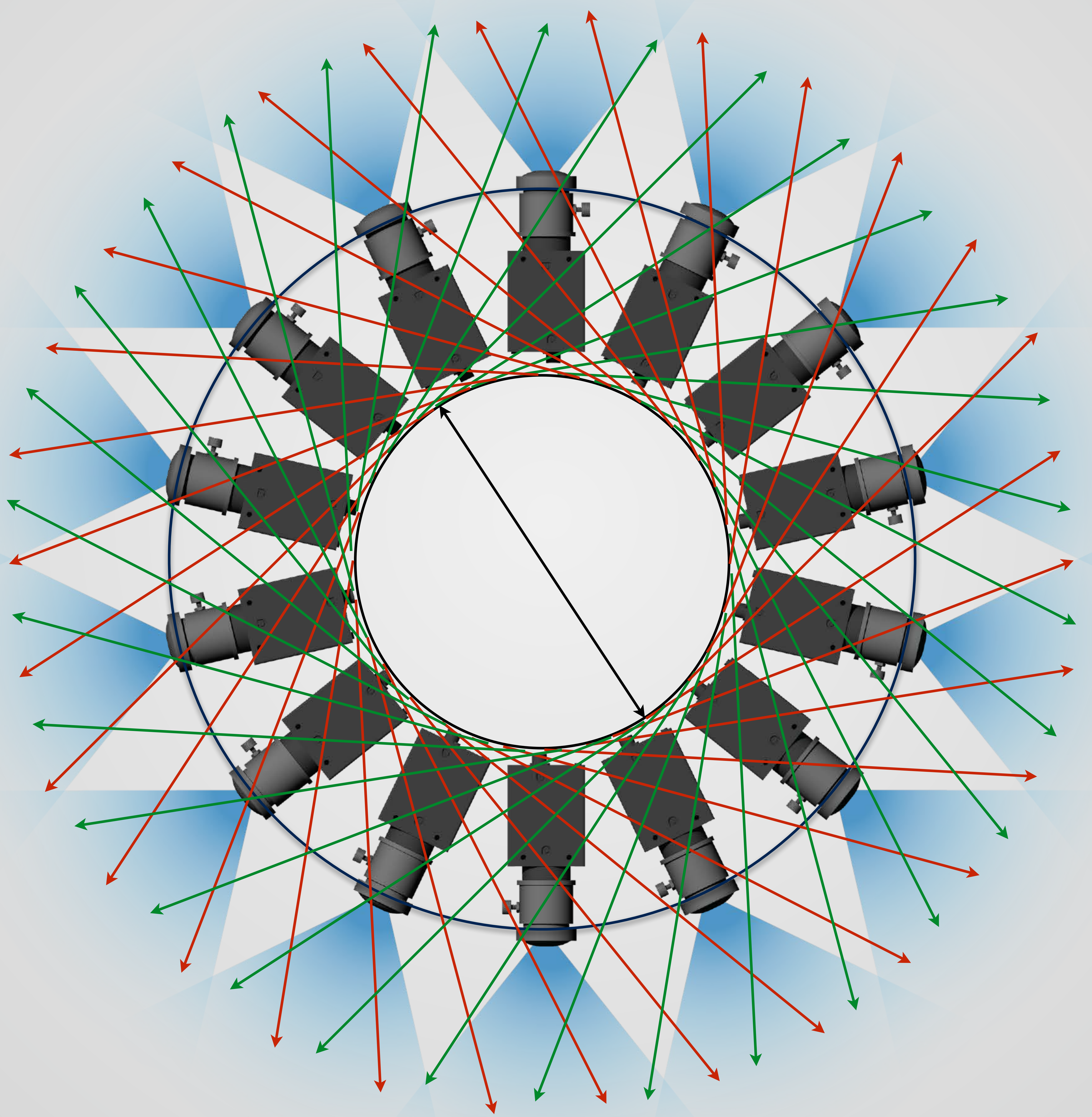


Creating ODS with a fixed array of cameras



- Warp/interpolate nearest 2 images
- Only need to do it for each specific slit
- The virtual camera is modeled as pinhole
- There are $2 * width$ slits
- Blend between cameras
- Handle ghosting via disparity clustering

Creating ODS with a fixed array of cameras



Optical flow between two images

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t)$$

$$I(x + \Delta x, y + \Delta y, t + \Delta t) = I(x, y, t) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t + \ldots$$

$$\frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t = 0$$

$$\frac{\partial I}{\partial x}\frac{\Delta x}{\Delta t} + \frac{\partial I}{\partial y}\frac{\Delta y}{\Delta t} + \frac{\partial I}{\partial t}\frac{\Delta t}{\Delta t} = 0$$

$$I_x V_x + I_y V_y = -I_t$$

Horn and Schunck

$$E = \int\int [(I_x V_x + I_y V_y + I_t)^2 + \alpha^2(||\nabla V_x||^2 + ||\nabla V_y||^2)]dxdy$$

Solving the 3-D Euler-Lagrange equations

$$I_x(I_x^{-k} V_x + I_y^{-k} V_y + I_t) - \alpha^2 \Delta V_x = 0$$
$$I_y(I_x^{-k} V_x + I_y^{-k} V_y + I_t) - \alpha^2 \Delta V_y = 0$$

Using finite difference approximations and rearranging

$$(I_x^2 + 4\alpha^2)V_x + I_x I_y V_y = 4\alpha^2 \overline{V_x} - I_x I_t$$
$$(I_y^2 + 4\alpha^2)V_y + I_x I_y V_x = 4\alpha^2 \overline{V_y} - I_y I_y$$

Solving for the next flow time step

$$V_x^{k+1} = V_x^{-k} - \frac{I_x(I_x^{-k} V_x + I_y^{-k} V_y + I_t)}{4\alpha^4 + I_x^2 + I_y^2}$$

$$V_y^{k+1} = V_y^{-k} - \frac{I_y(I_x^{-k} V_x + I_y^{-k} V_y + I_t)}{4\alpha^4 + I_x^2 + I_y^2}$$

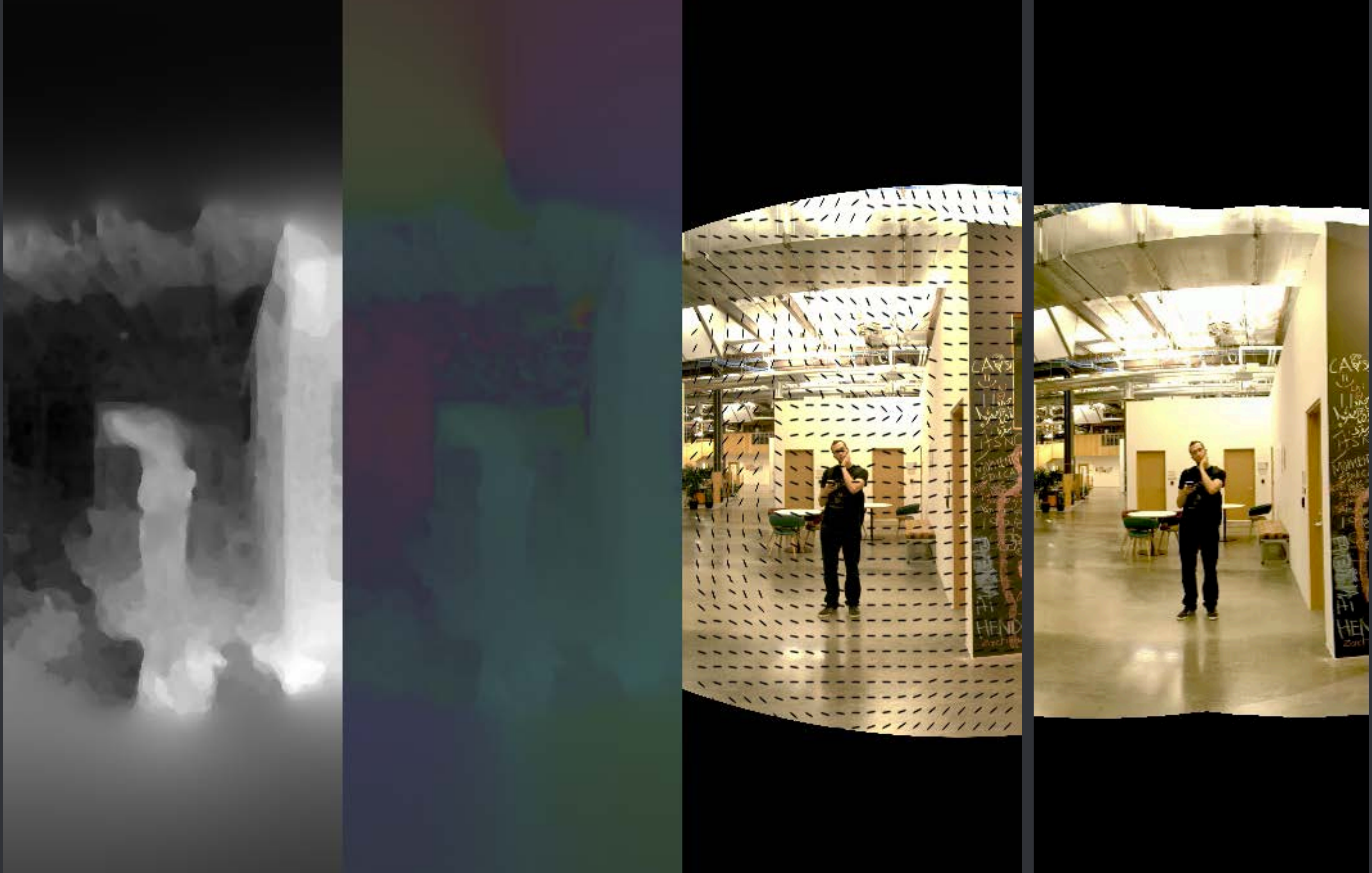

HENDRIK!


Riya
HENDRIK!

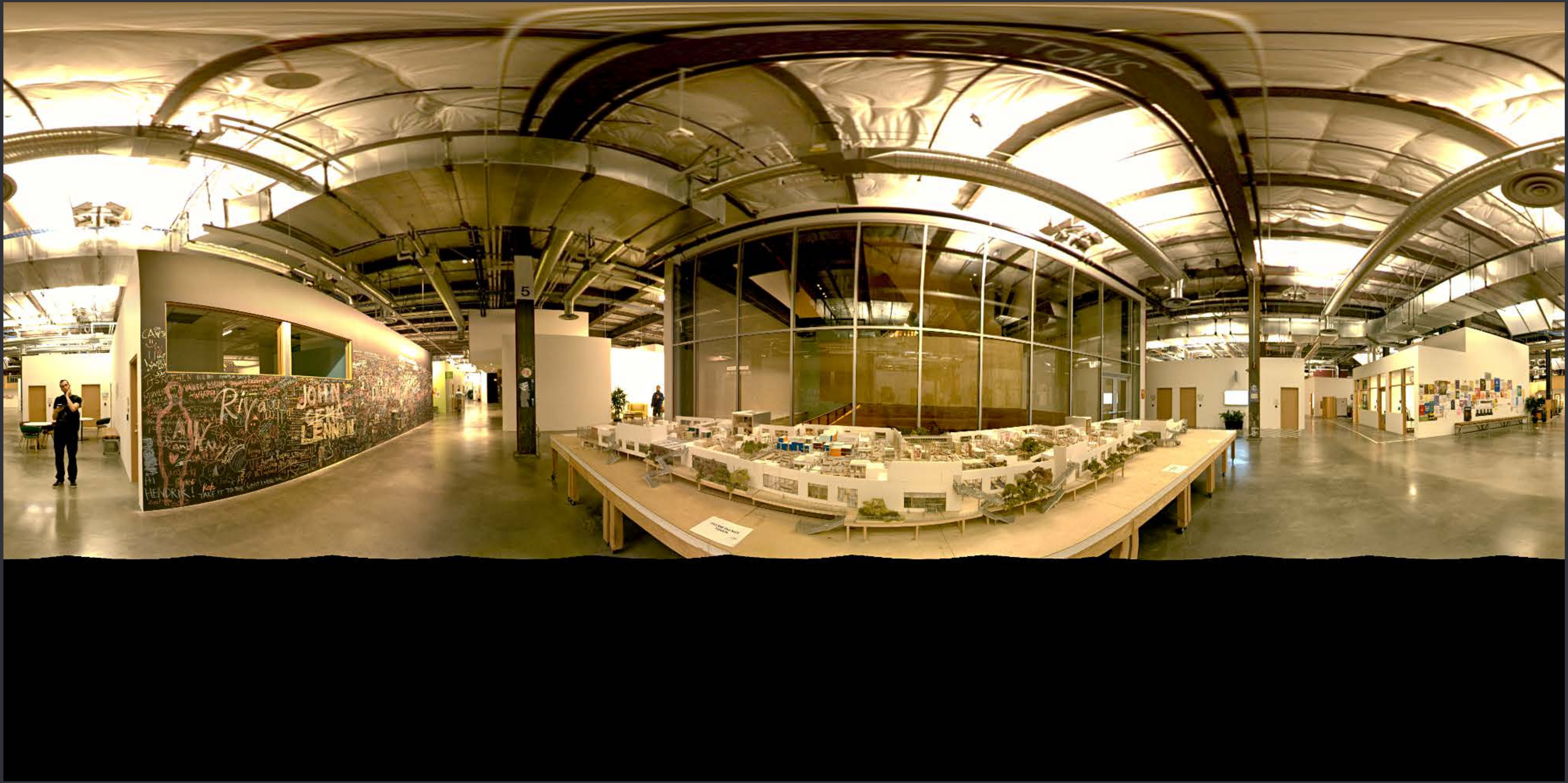

JOHN
LENNON
HENDRIK!


JOHN
LENNON
HENDRIK!
5

Spherical projections align s.t. parallax = 0 @ infinity



Sharpen (Periodic Boundary Aware)




NEW HAVEN LINE DEPARTURES
INFORMATION

Thank you