Lecture 27:

Rendering Challenges of VR

Computer Graphics
CMU 15-462/15-662, Fall 2015

Virtual reality (VR) vs augmented reality (AR)

VR = virtual reality

User is completely immersed in virtual world (sees only light emitted by display

AR = augmented reality

Display is an overlay that augments user's normal view of the real world (e.g., terminator)

Image credit: Terminator 2 (naturally)

VR headsets

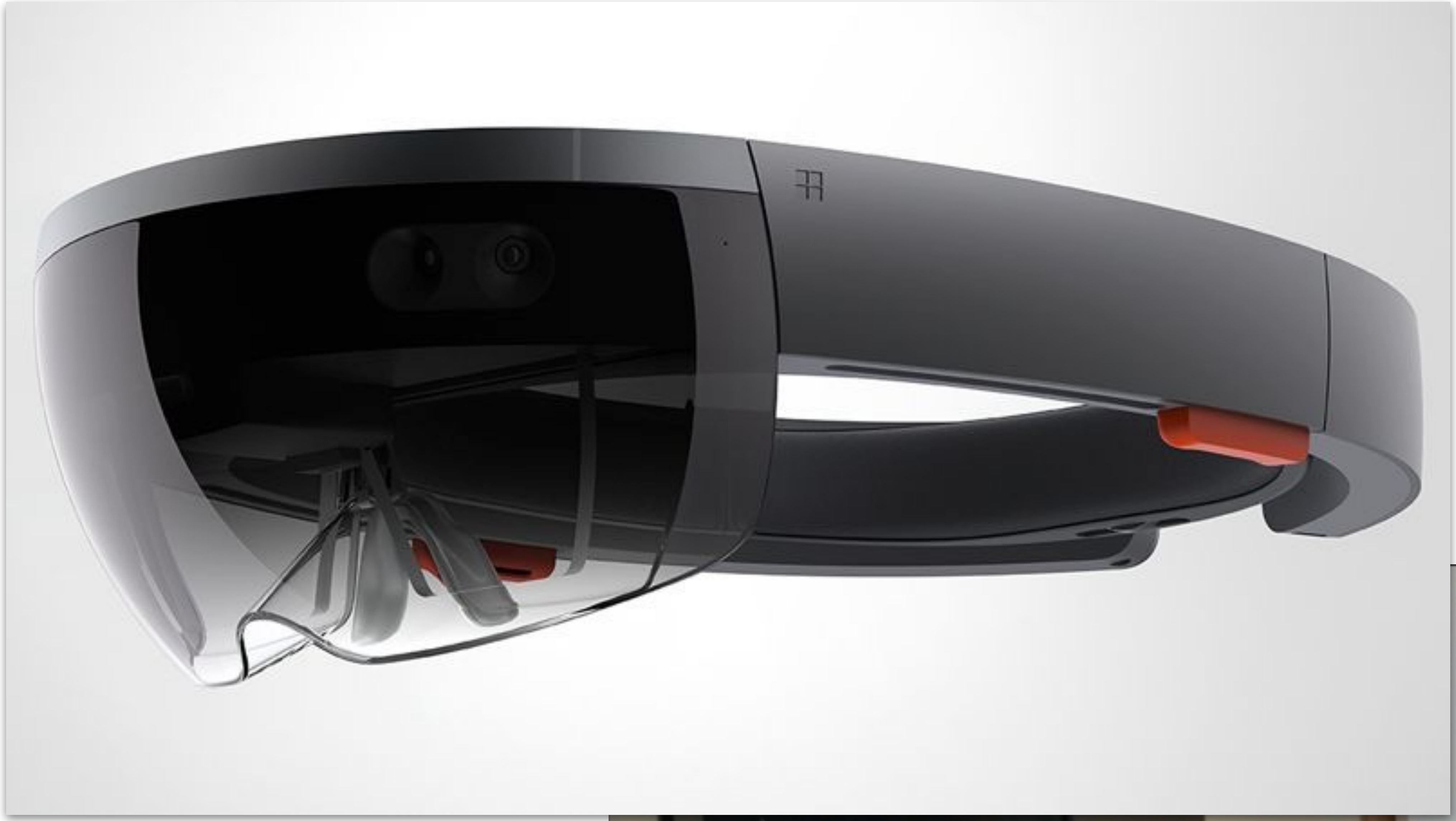
Oculus Rift (Crescent Bay Prototype)

Sony Morpheus

HTV Vive

Google Cardboard

AR headset: Microsoft Hololens



Today: rendering challenges of VR

- Since you are now all experts in renderings, today we will talk about the unique challenges of rendering in the context of modern VR headsets

- VR presents many other difficult technical challenges
  - display technologies
  - accurate tracking of face, head, and body position
  - haptics (simulation of touch)
  - sound synthesis
  - user interface challenges (inability of user to walk around environment, how to manipulate objects in virtual world)
  - content creation challenges
  - and on and on…

VR gaming

Bullet Train Demo (Epic)

VR video

VR video

VR teleconference / video chat

http://vrchat.com/

Oculus Rift DK2

Rift DK2 is best documented of modern prototypes, so I'll use it for discussion here

Oculus Rift DK2

Oculus Rift DK2 headset

Image credit: ifixit.com

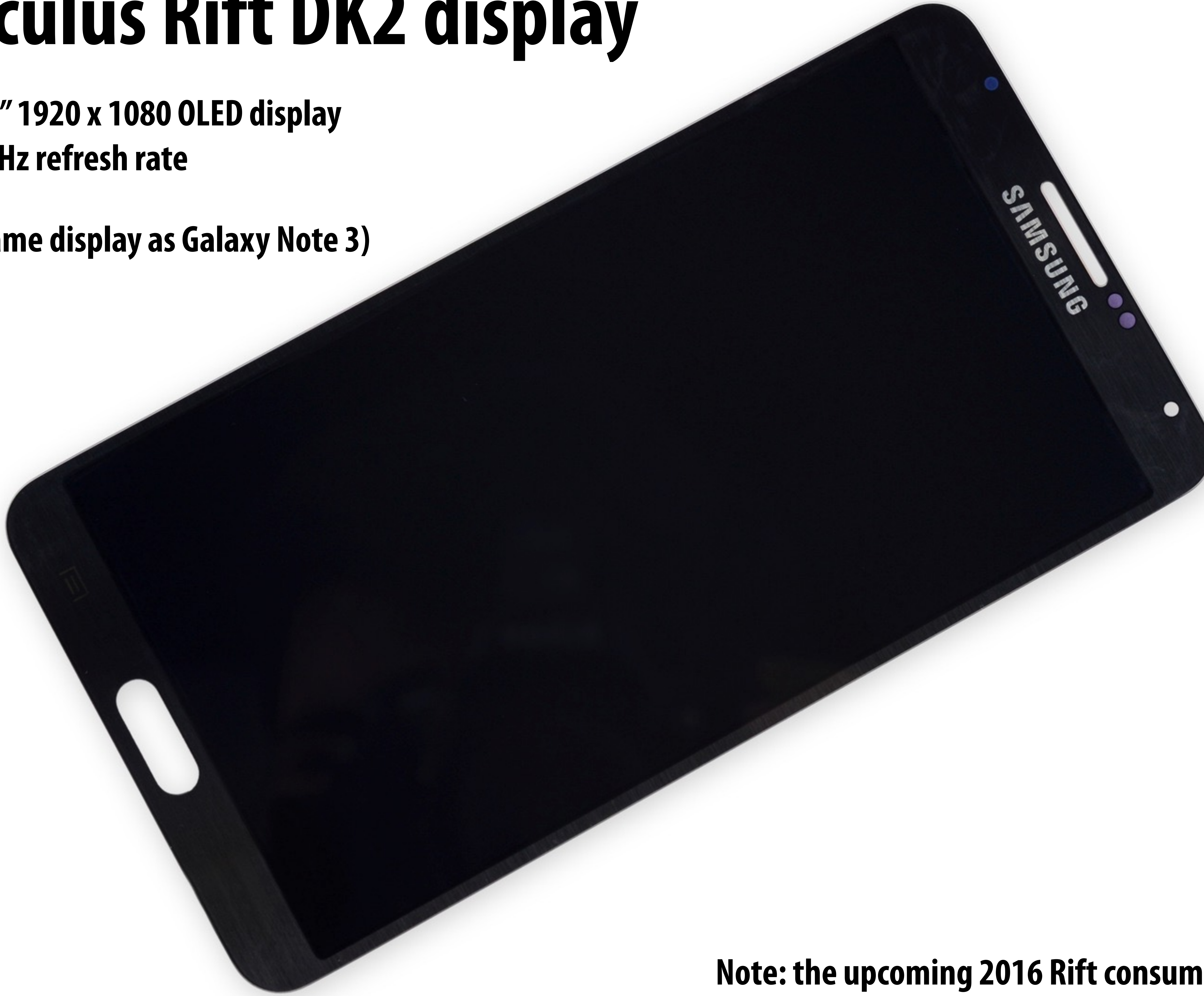
Oculus Rift DK2 headset

Image credit: ifixit.com

Oculus Rift DK2 display

5.7" 1920 x 1080 OLED display
75 Hz refresh rate

(Same display as Galaxy Note 3)



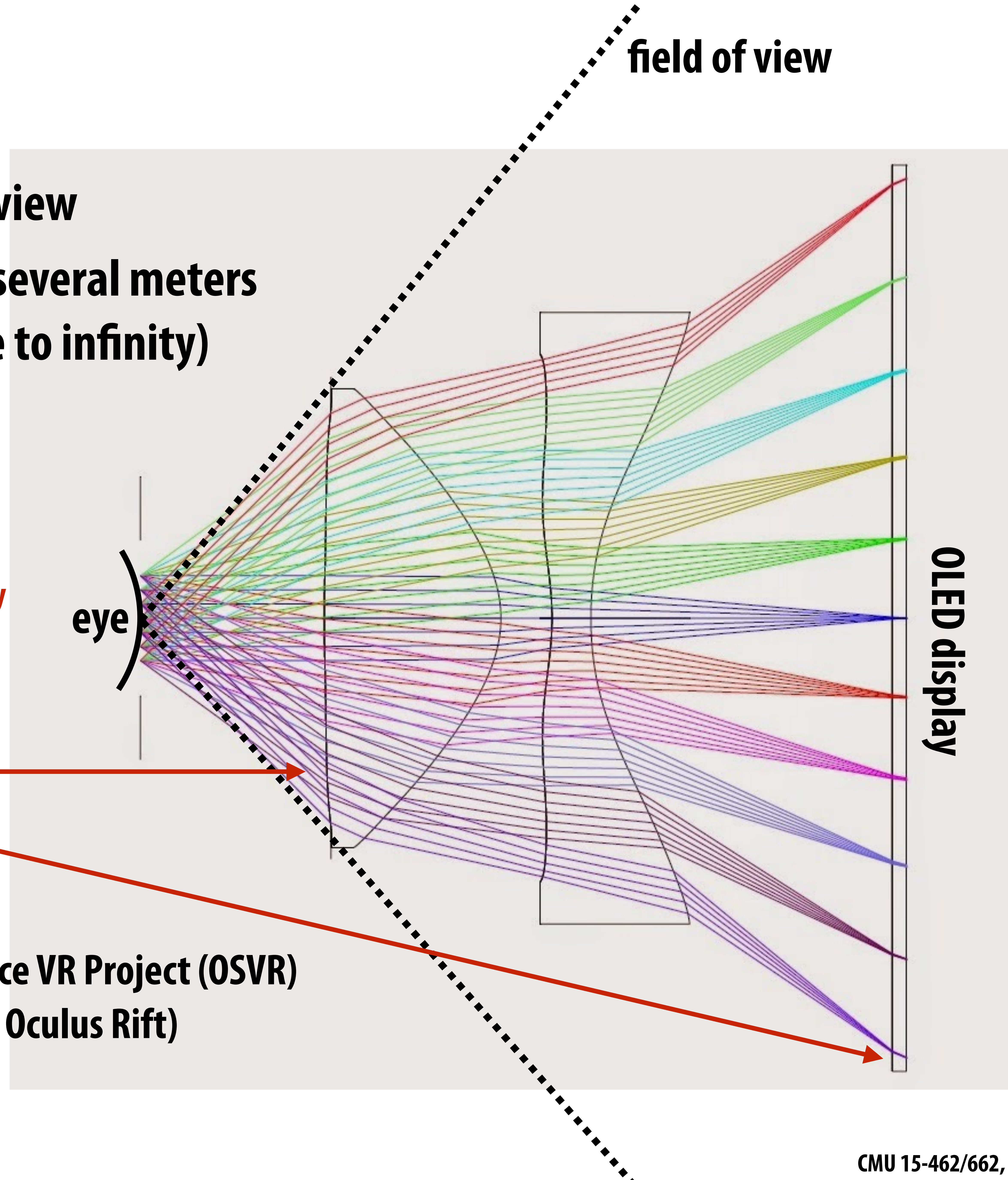
Note: the upcoming 2016 Rift consumer product features two 1080×1200 displays at 90Hz.

Image credit: ifixit.com

Role of optics

1. Create wide field of view
2. Place focal plane at several meters away from eye (close to infinity)

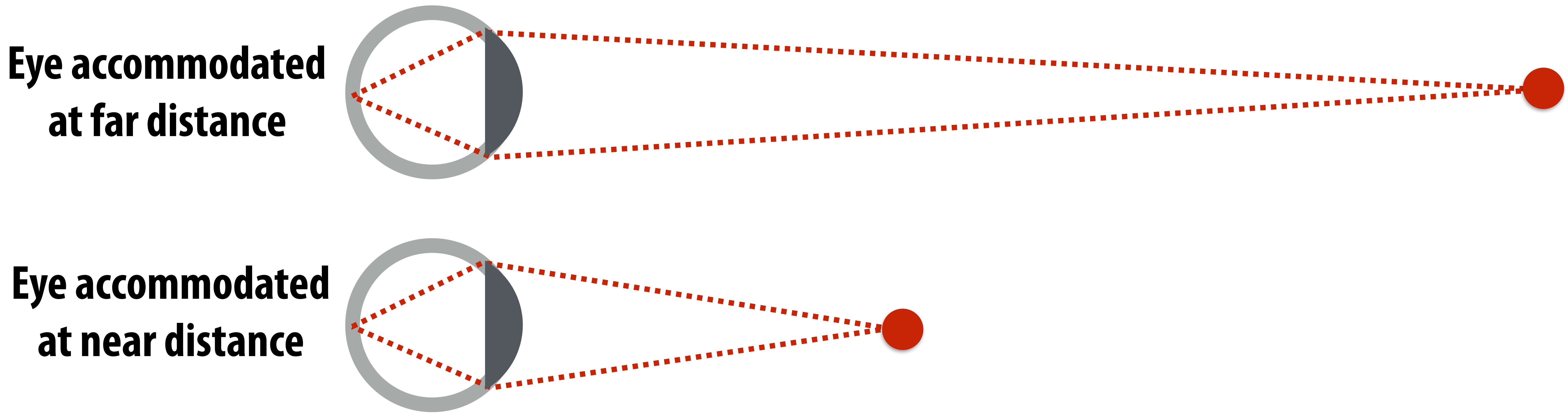
Note: parallel lines reaching eye converge to a single point on display (eye accommodates to plane near infinity)



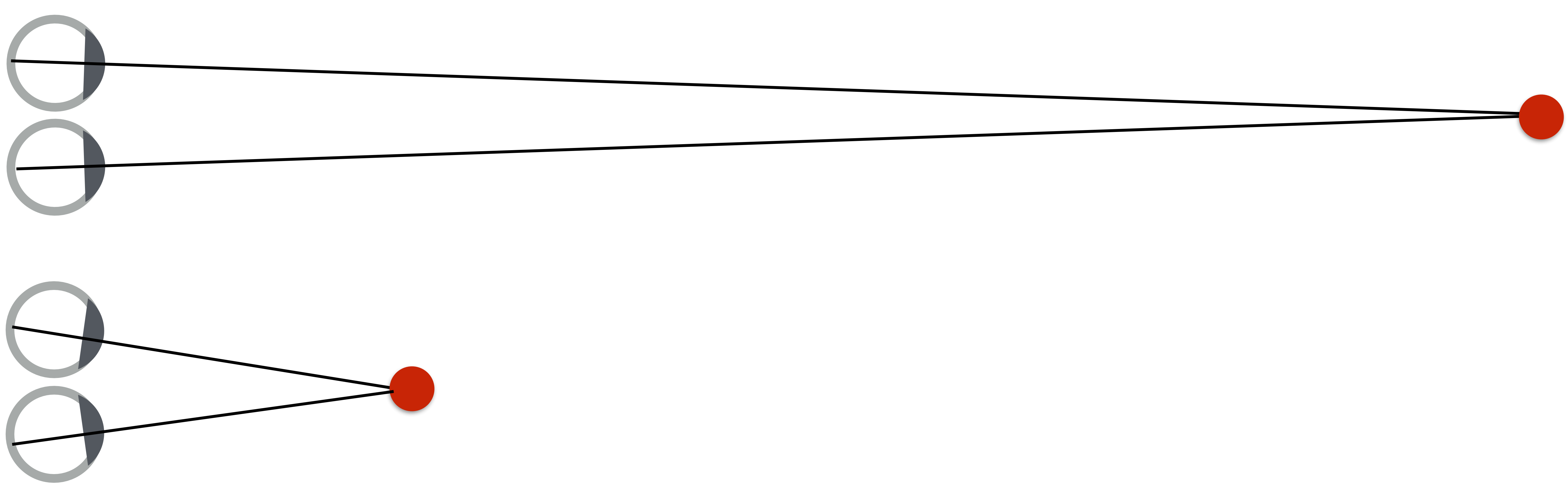
Lens diagram from Open Source VR Project (OSVR)
(Not the lens system from the Oculus Rift)
http://www.osvr.org/

Accommodation and vergence

Accommodation: changing the optical power of the eye to focus at different distances

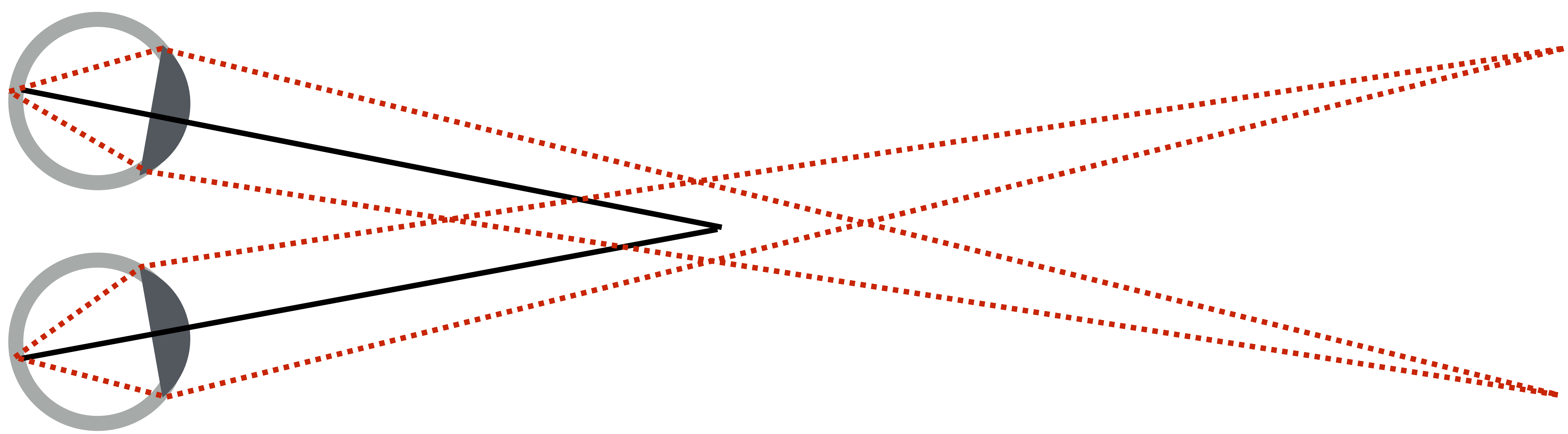


Vergence: rotation of eye to ensure projection of object falls in center of retina



Accommodation - vergence conflict

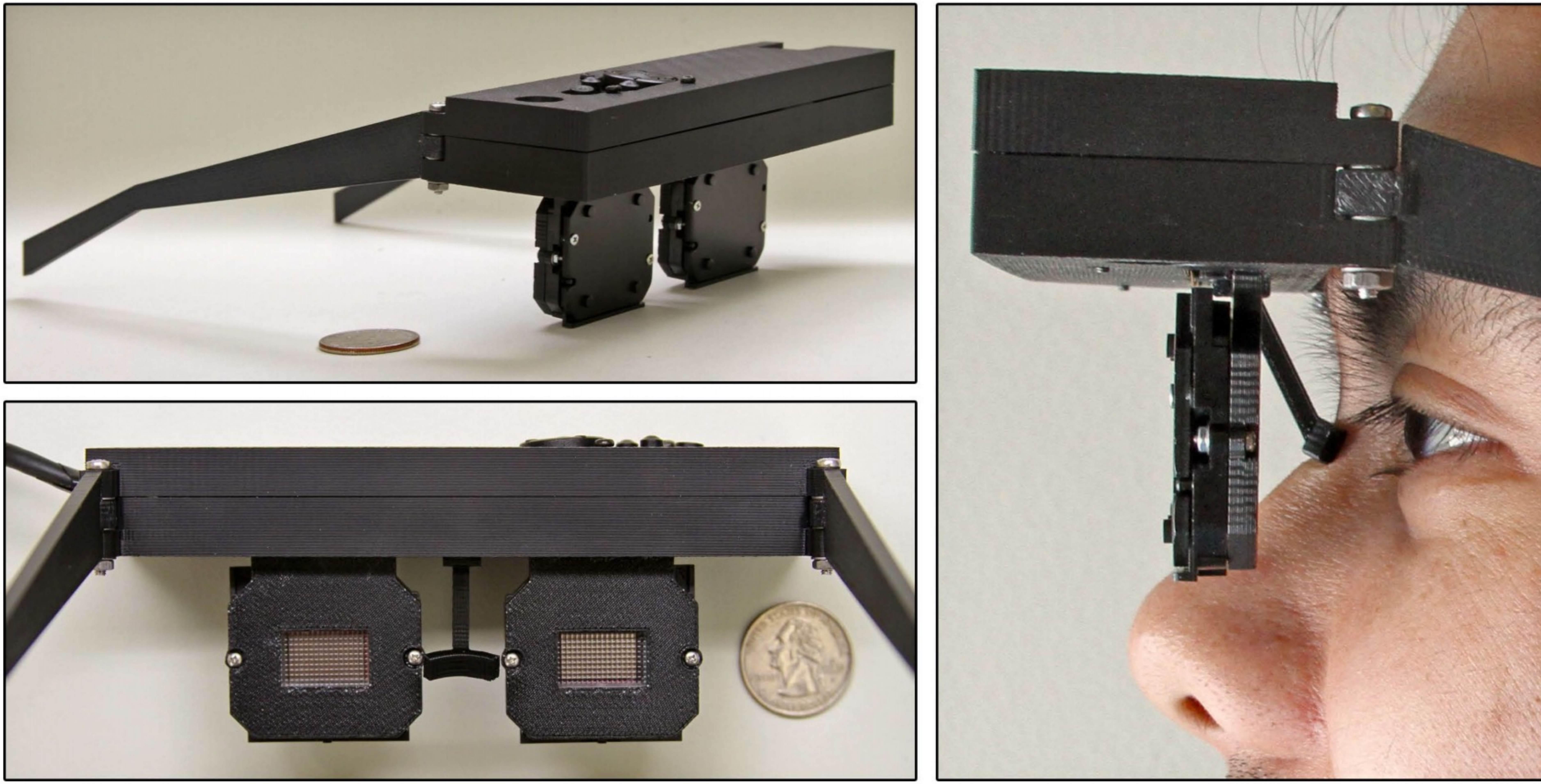
- Given design of current VR displays, consider what happens when objects are up-close to eye in virtual scene
  - Eyes must remain accommodated to near infinity (otherwise image on screen won't be in focus)
  - But eyes must converge in attempt to fuse stereoscopic images of object up close
  - Brain receives conflicting depth clues… (discomfort, fatigue, nausea)



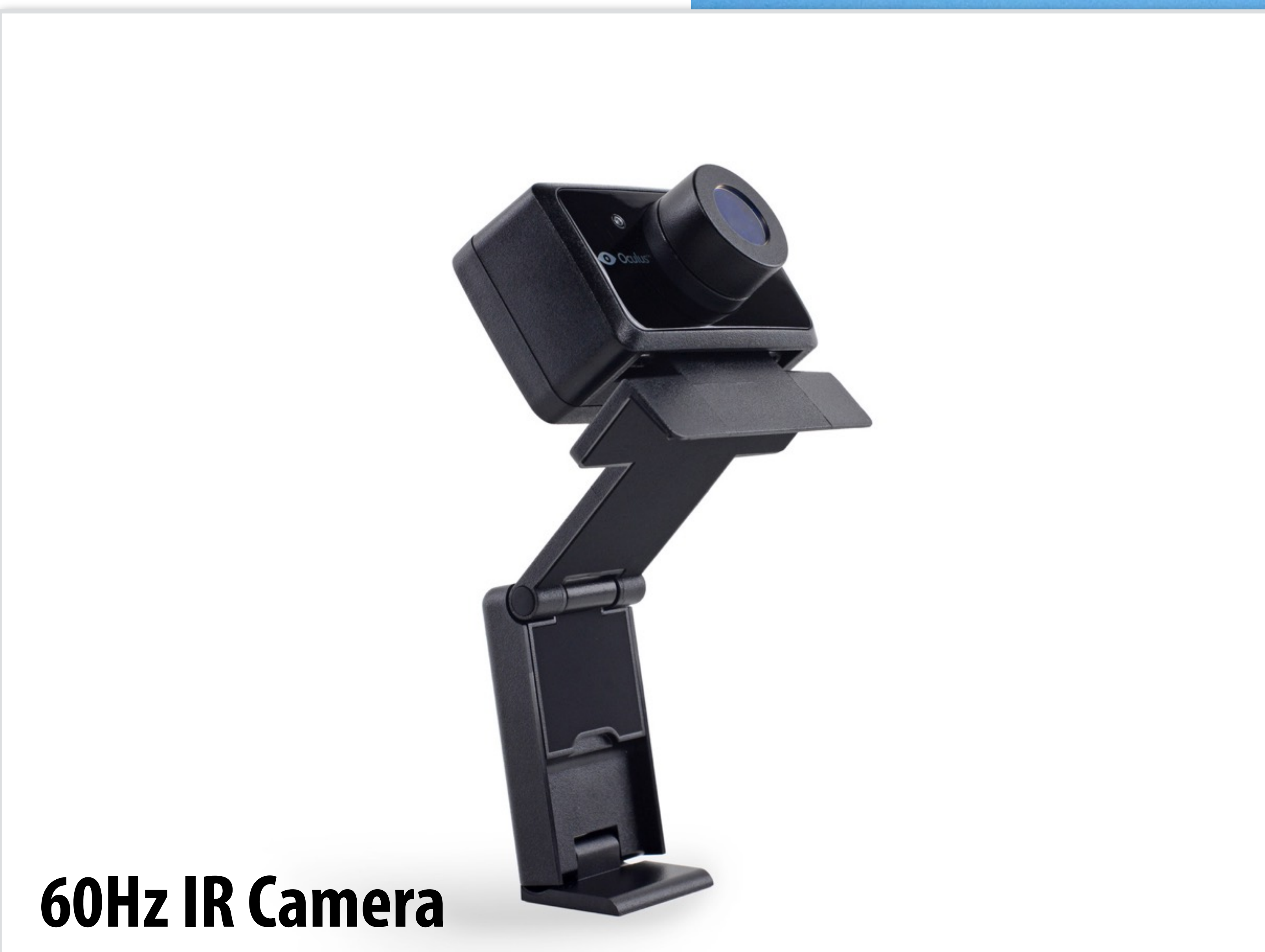
This problem stems from nature of display design. If you could just make a display that emits the light field that would be produced by a virtual scene, then you could avoid the accommodation - vergence conflict…

Aside: near-eye light field displays

Recreate light field in front of eye



Oculus DK2 IR camera and IR LEDs



60Hz IR Camera

Headset contains:

40 IR LEDs
Gyro + accelerometer (1000Hz)

Name of the game, part 1: low latency

- The goal of a VR graphics system is to achieve "presence", tricking the brain into thinking what it is seeing is real

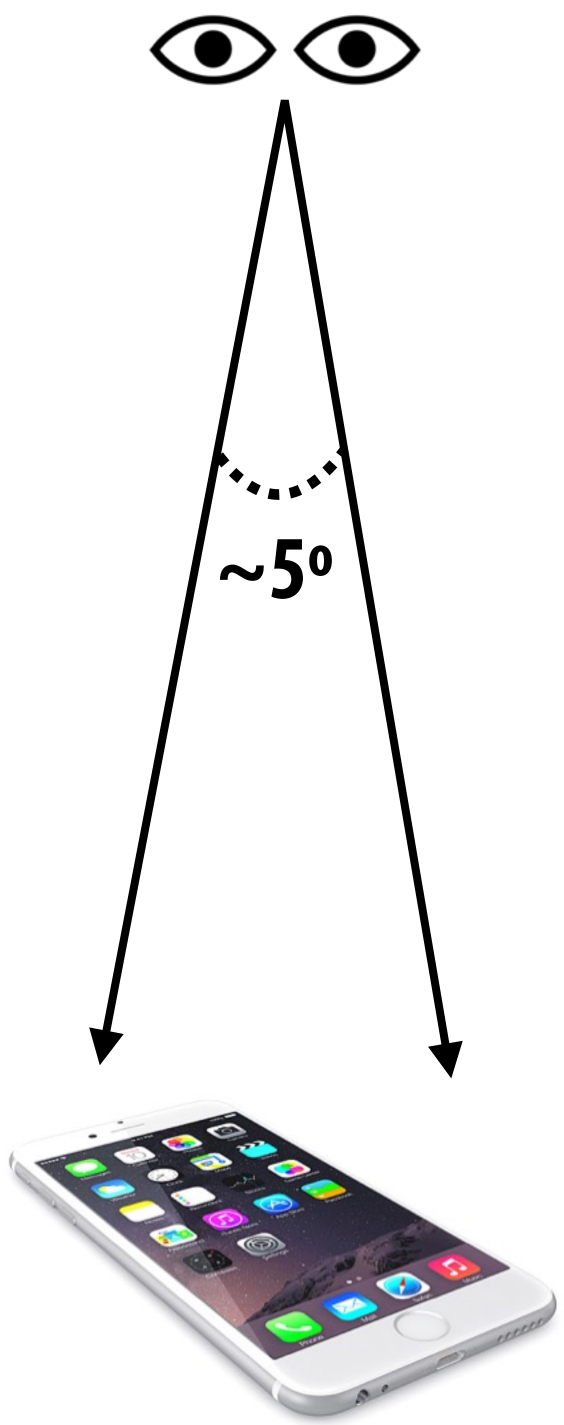
- Achieving presence requires an exceptional low-latency system
  - What you see must change when you move your head!
  - End-to-end latency: time from moving your head to the time new photons hit your eyes
    - Measure user's head movement
    - Update scene/camera position
    - Render new image
    - Transfer image to headset, then to transfer to display in headset
    - Actually emit light from display (photons hit user's eyes)
  - Latency goal of VR: 10-25 ms
    - Requires exceptionally low-latency head tracking
    - Requires exceptionally low-latency rendering and display

Thought experiment: effect of latency

- Consider a 1,000 x 1,000 display spanning 100° field of view
  - 10 pixels per degree
- Assume:
  - You move your head 90° in 1 second (only modest speed)
  - End-to-end latency of system is 50 ms (1/20 sec)
- Therefore:
  - Displayed pixels are off by 4.5° ~ 45 pixels from where they would be in an ideal system with 0 latency

Example credit: Michael Abrash

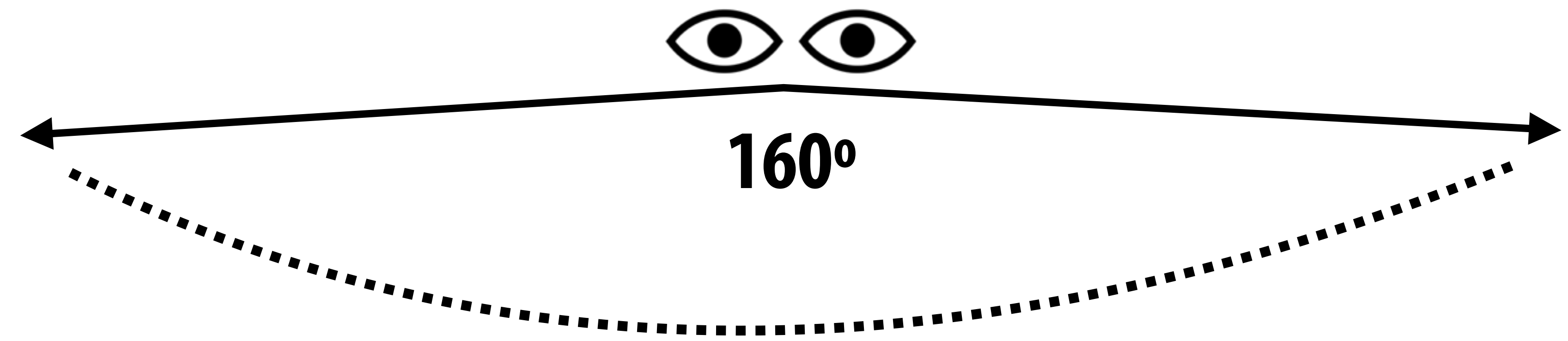
Name of the game, part 2: high resolution



iPhone 6: 4.7 in "retina" display:

1.3 MPixel

326 ppi → 57 ppd



Human: ~160° view of field per eye (~200° overall)

(Note: this does not account for eye's ability to rotate in socket)

Future "retina" VR display:

57 ppd covering 200°

= 11K x 11K display per eye

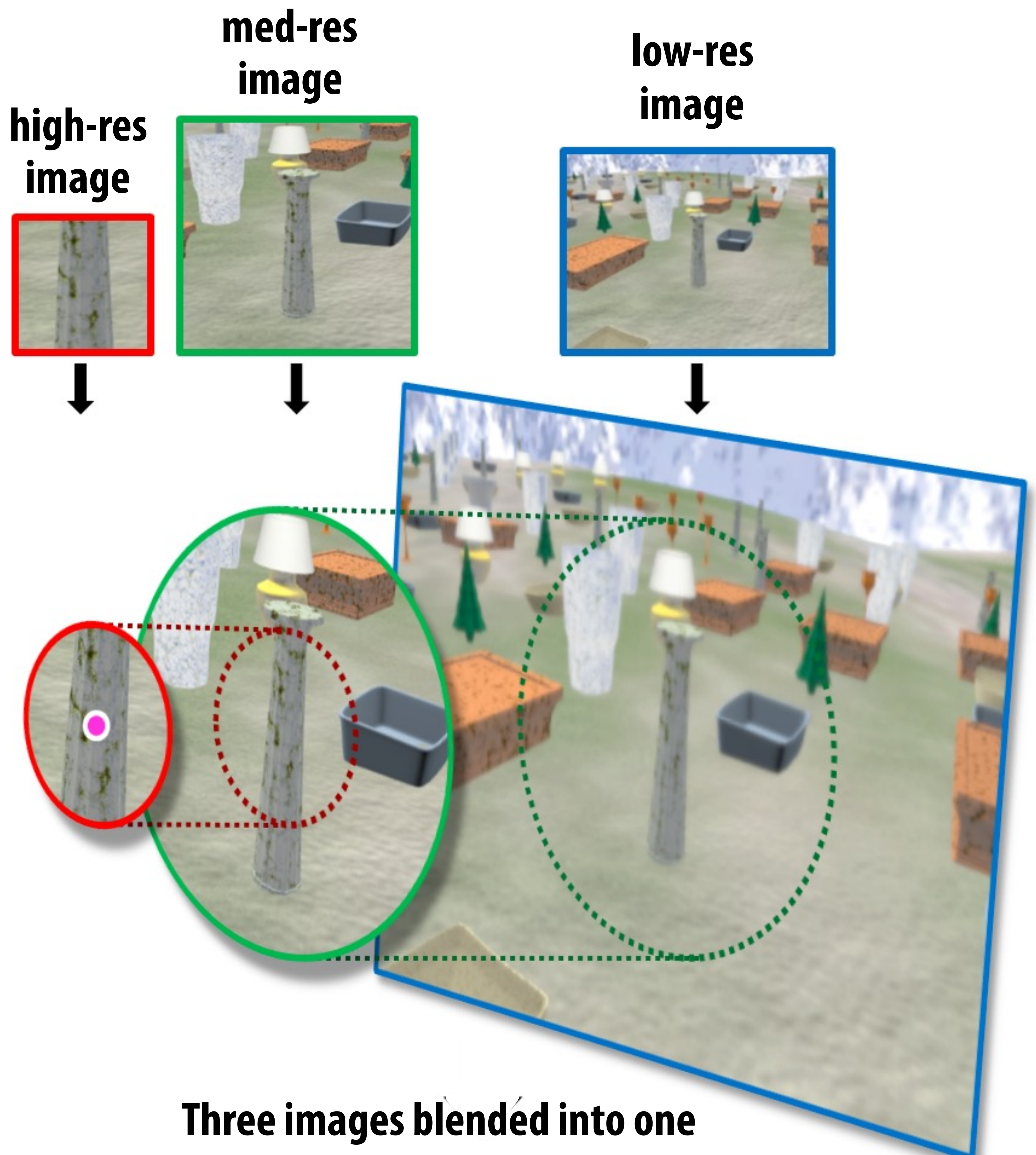
= 220 MPixel

Strongly suggests need for eye tracking and foveated rendering (eye can only perceive detail in 5° region about gaze point

Eyes designed by SuperAtic LABS from the thenounproject.com

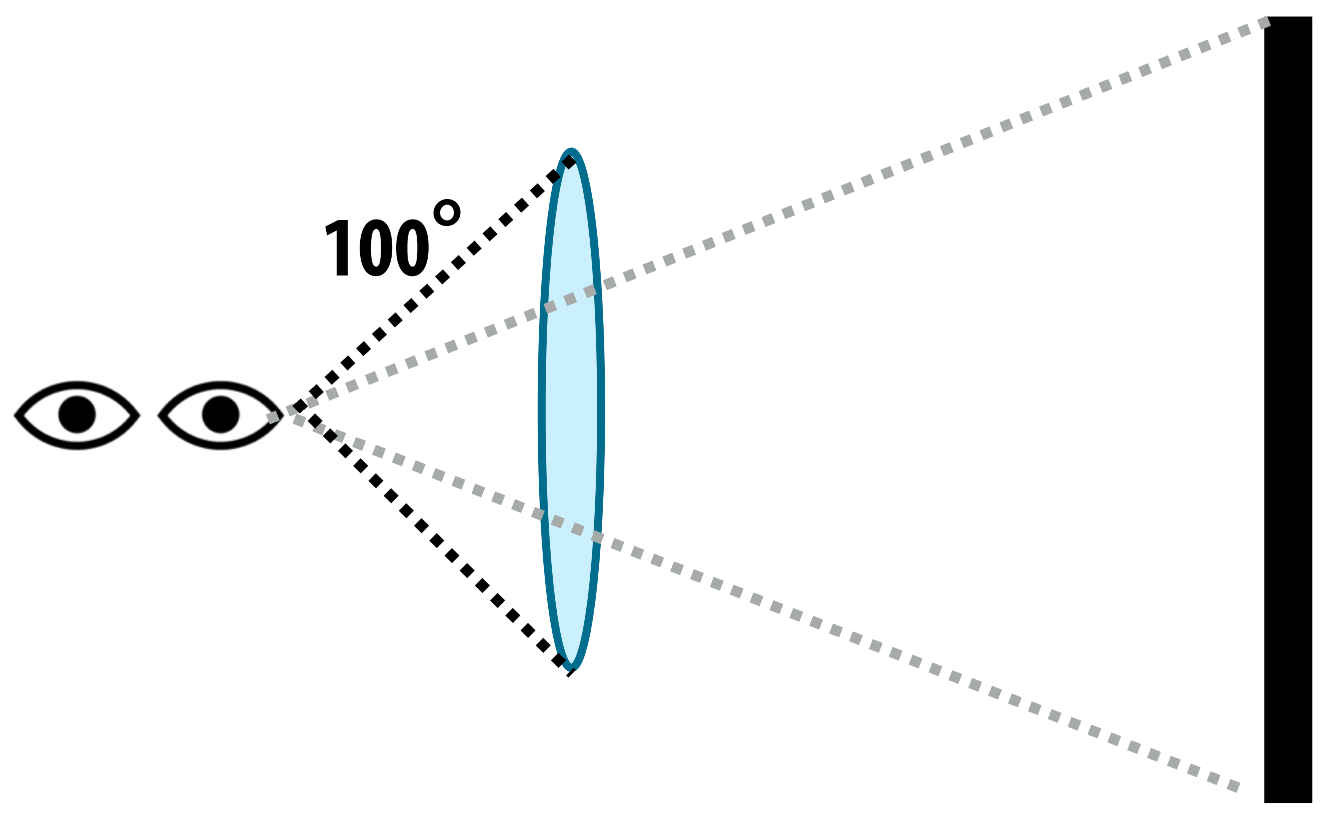
Foveated rendering

Idea: track user's gaze, render with increasingly lower resolution farther away from gaze point

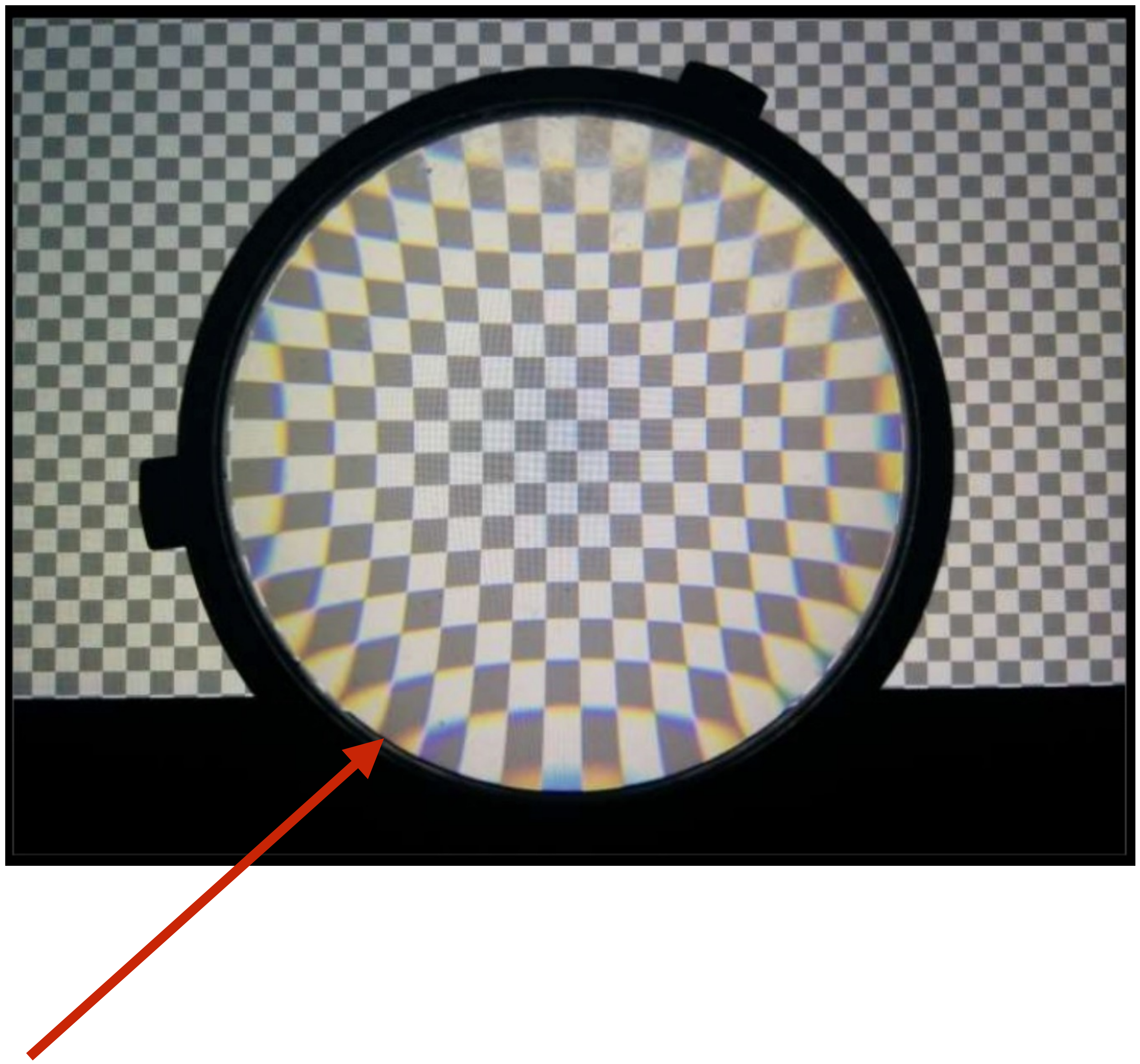


Three images blended into one for display

Requirement: wide field of view



View of checkerboard through Oculus Rift lens

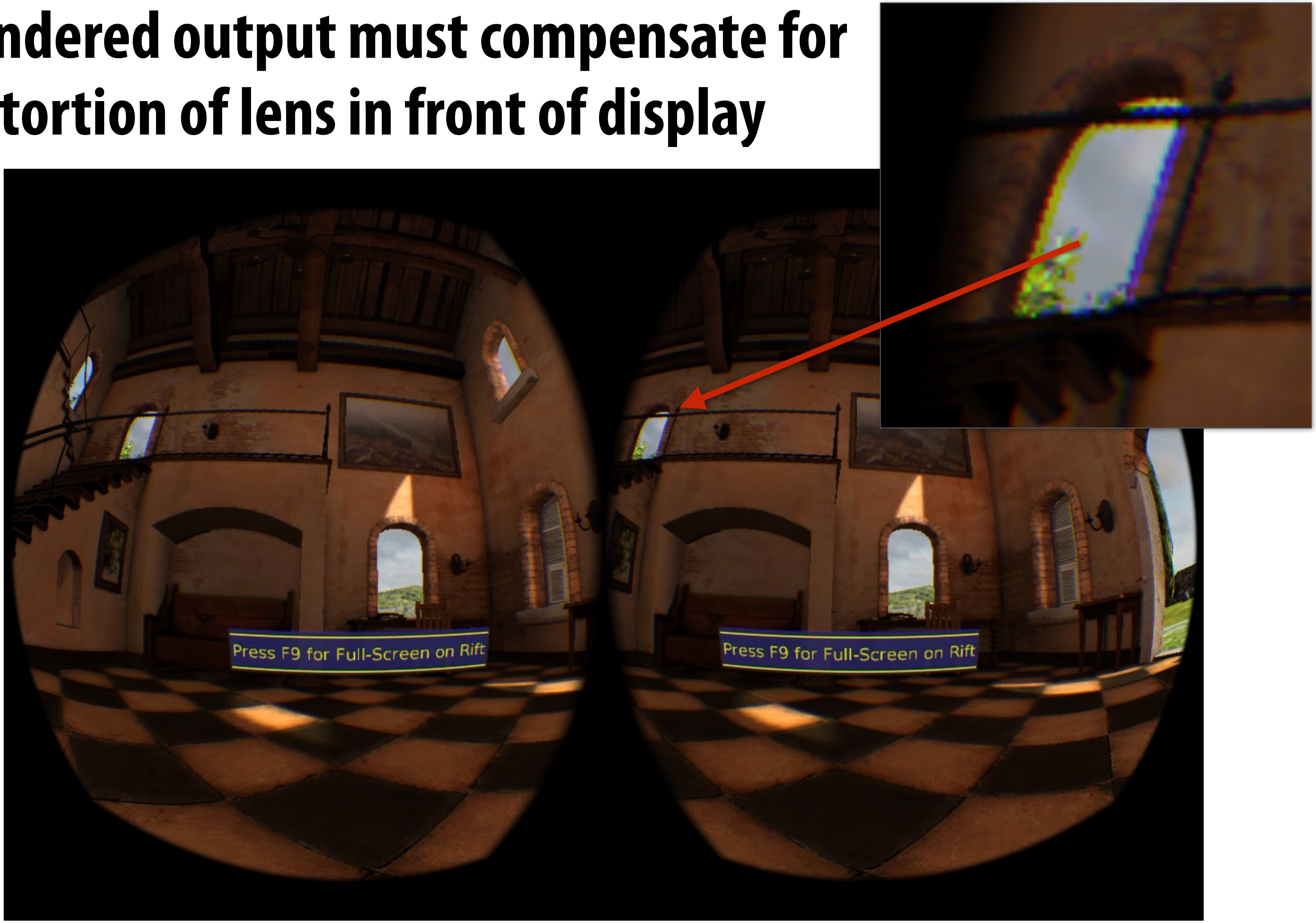


Lens introduces distortion

- Pincushion distortion
- Chromatic aberration (different wavelengths of light refract by different amount)

Icon credit: Eyes designed by SuperAtic LABS from the thenounproject.com
Image credit: Cass Everitt

Rendered output must compensate for distortion of lens in front of display



Step 1: render scene using traditional graphics pipeline at full resolution for each eye

Step 2: warp images and composite into frame rendering is viewed correctly after lens distortion

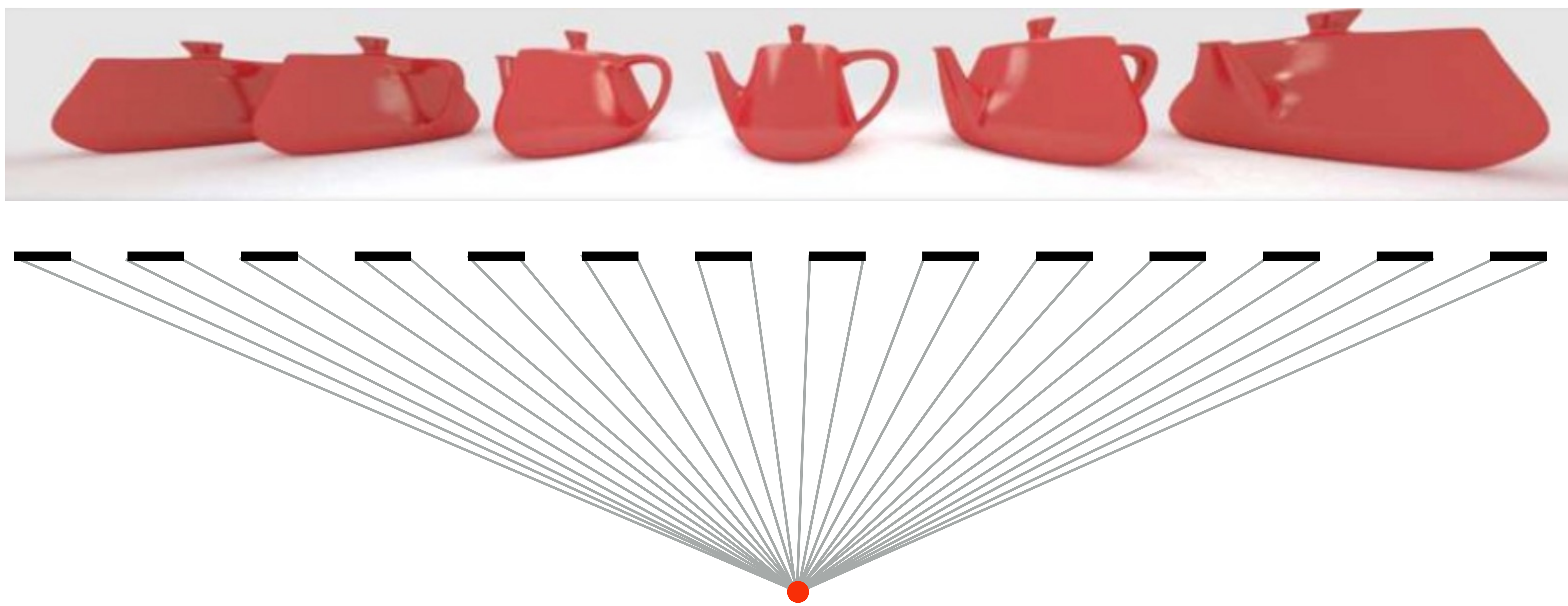
(Can apply unique distortion to R, G, B to approximate correction for chromatic aberration)

Image credit: Oculus VR developer guide

Challenge: rendering via planar projection

Recall: rasterization-based graphics is based on perspective projection to plane

- Reasonable for modest FOV, but distorts image under high FOV
- Recall: VR rendering spans wide FOV

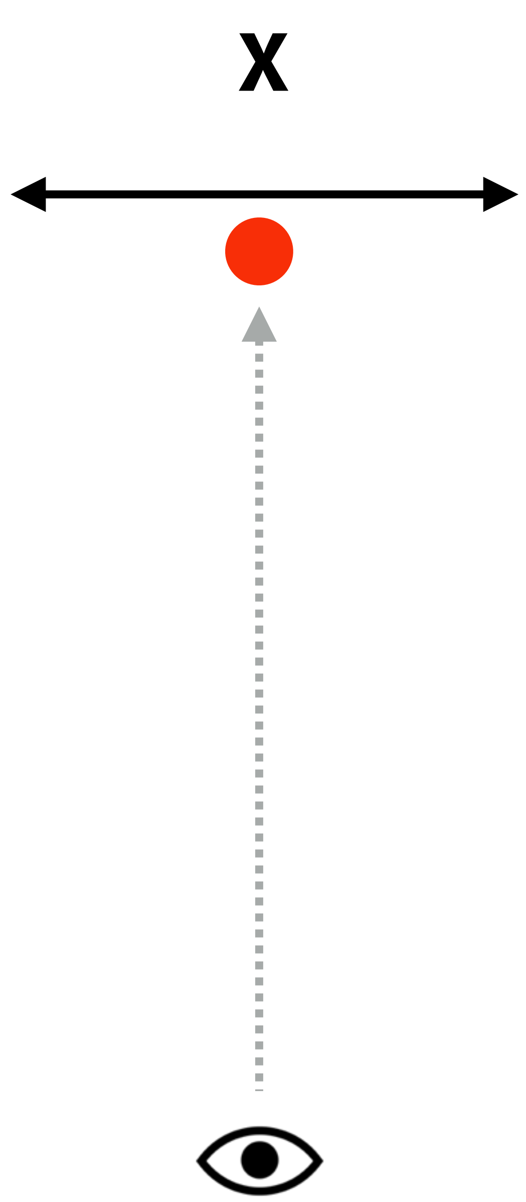


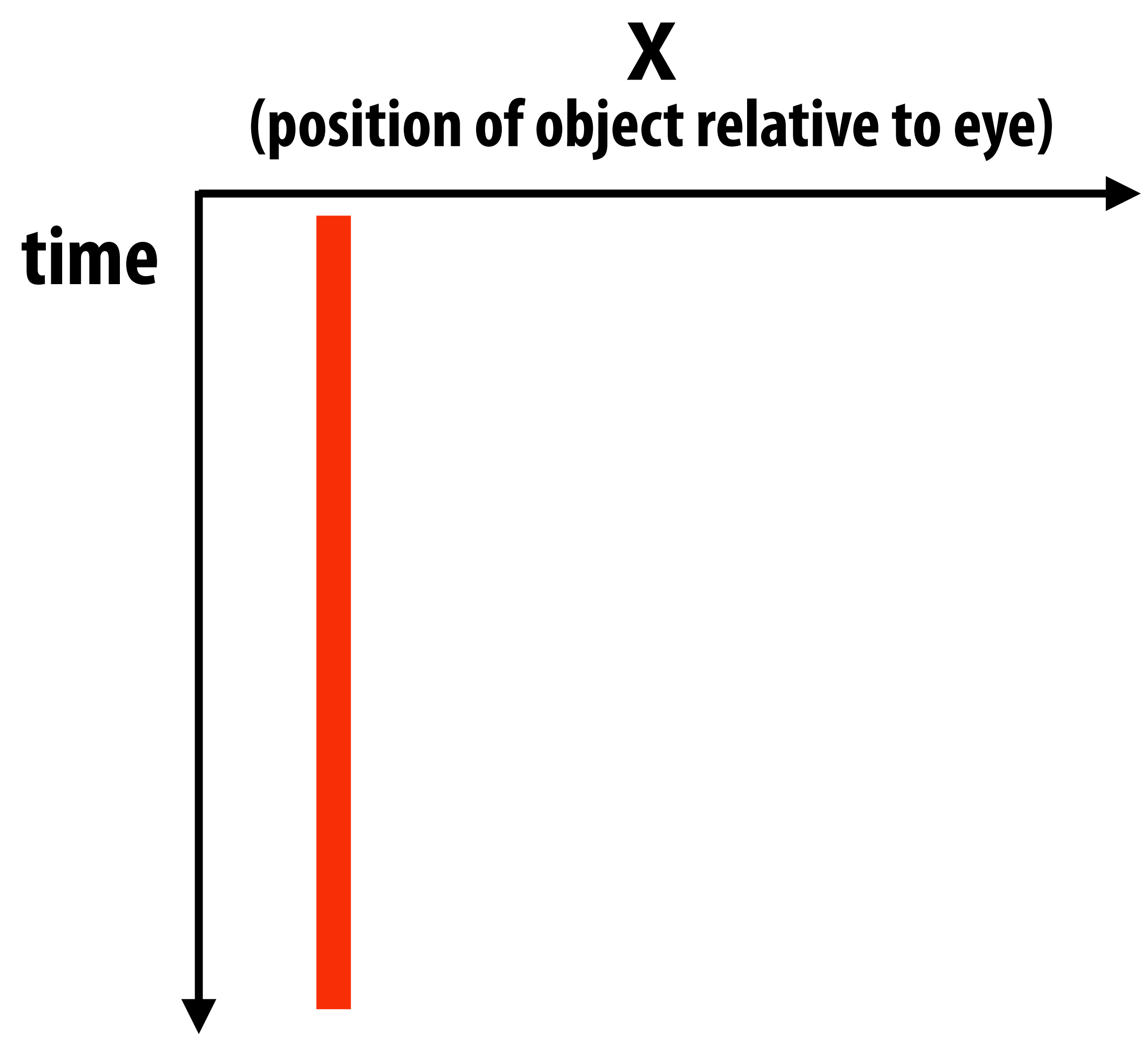
Pixels span larger angle in center of image
(lowest angular resolution in center)

Future investigations may consider: curved displays, ray casting to achieve uniform angular resolution, rendering with piecewise linear projection plane (different plane per tile of screen)

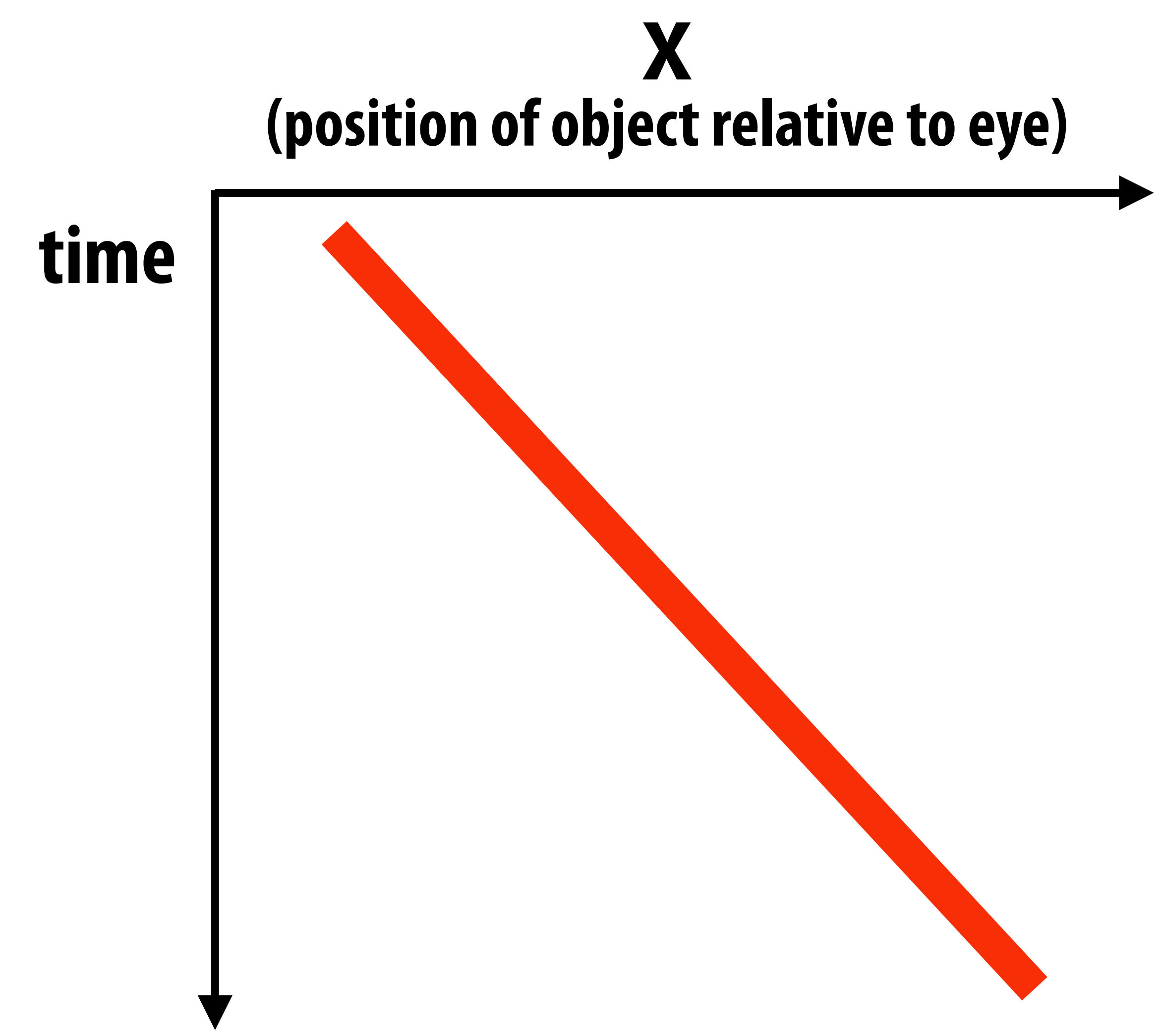
Image credit: Cass Everitt

Consider object position relative to eye





Case 1: object stationary relative to eye:
(eye still and red object still
OR
red object moving left-to-right and
eye moving to track object
OR
red object stationary in world but head moving
and eye moving to track object)



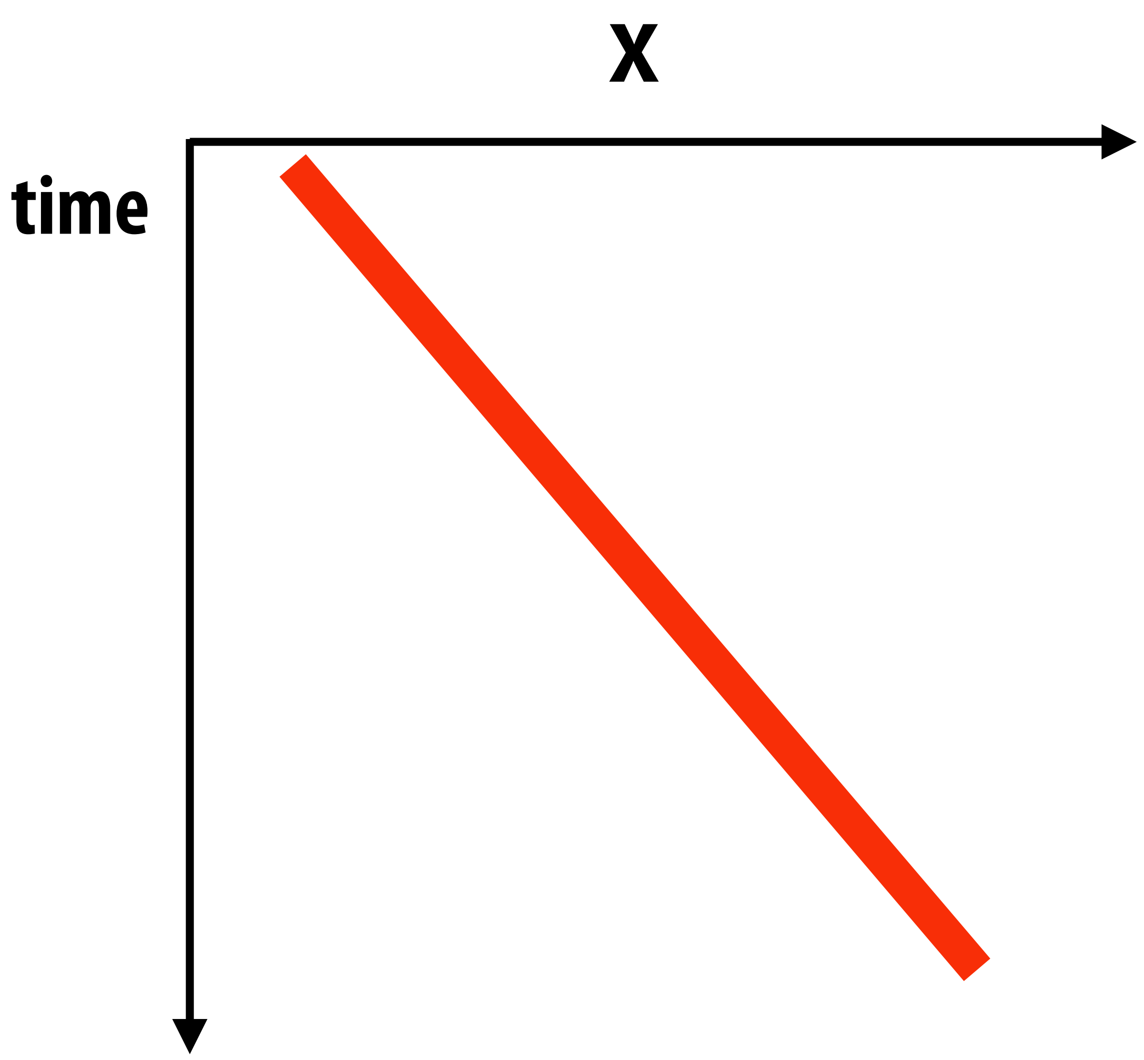
Case 2: object moving relative to eye:
(red object moving from left to right but
eye stationary, i.e., it's focused on a different
stationary point in world)

NOTE: THESE GRAPHS PLOT <u>OBJECT POSITION</u> RELATIVE TO EYE

RAPID HEAD MOTION WITH EYES TRACK A MOVING OBJECT IS A FORM OF CASE 1!!!

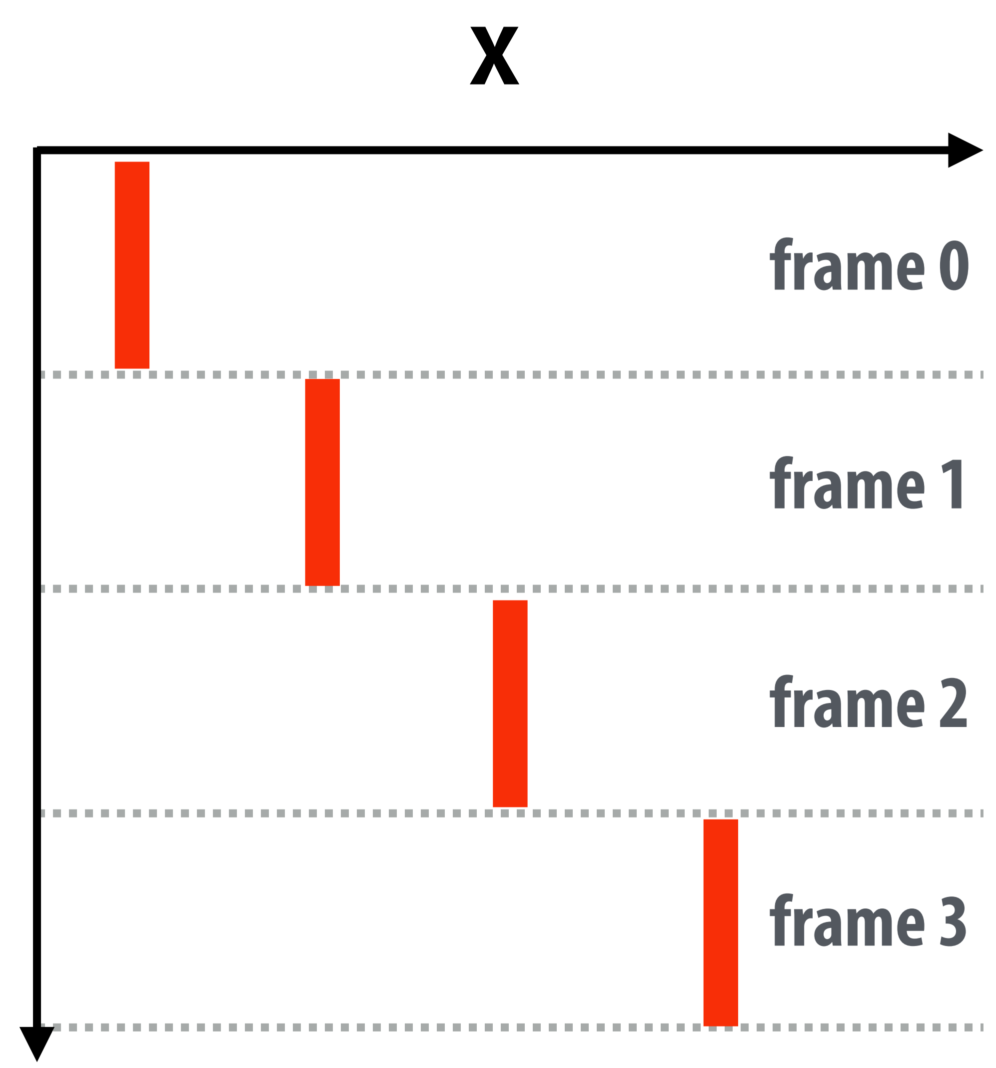
Spacetime diagrams adopted from presentations by Michael Abrash
Eyes designed by SuperAtic LABS from the thenounproject.com

Effect of latency: judder



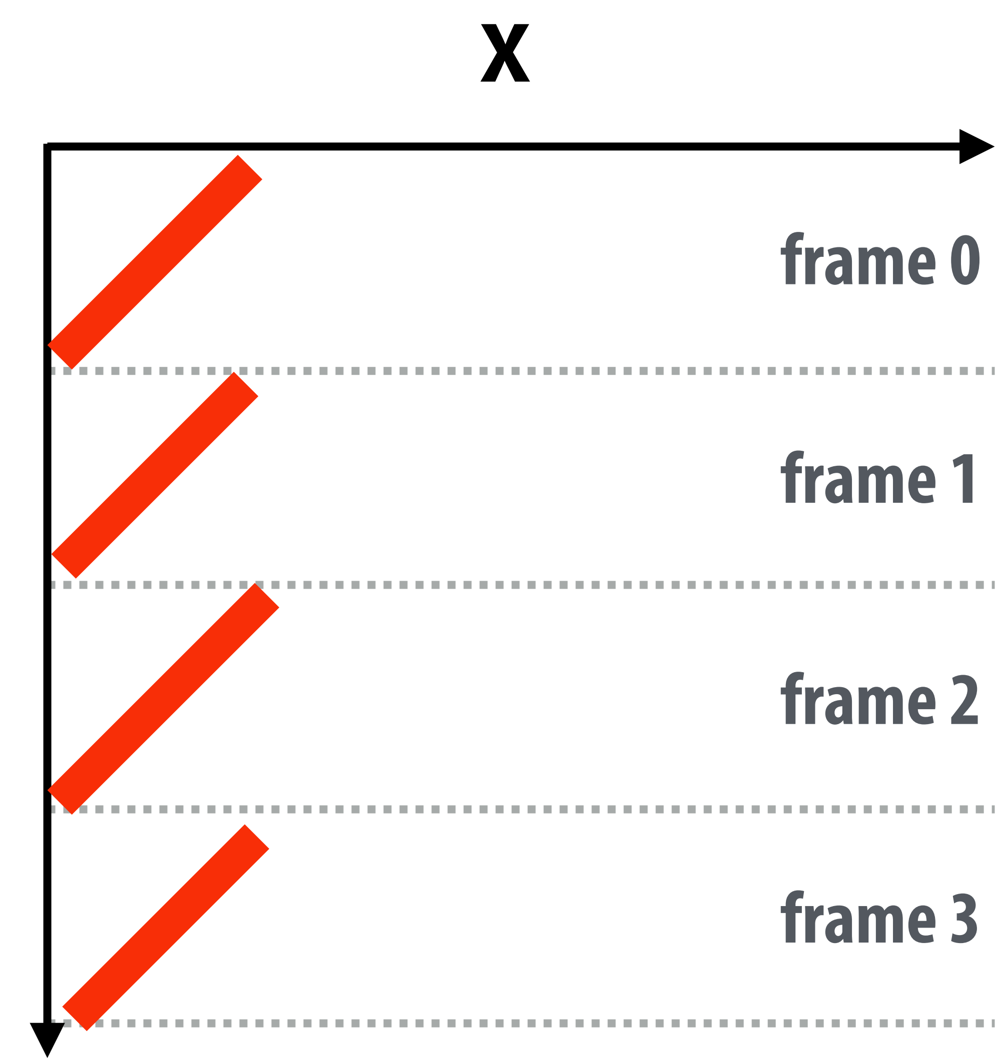
Case 2: object moving from left to right, eye stationary (eye stationary with respect to display)

Continuous representation.



Case 2: object moving from left to right, eye stationary (eye stationary with respect to display)

Light from display (image is updated each frame)



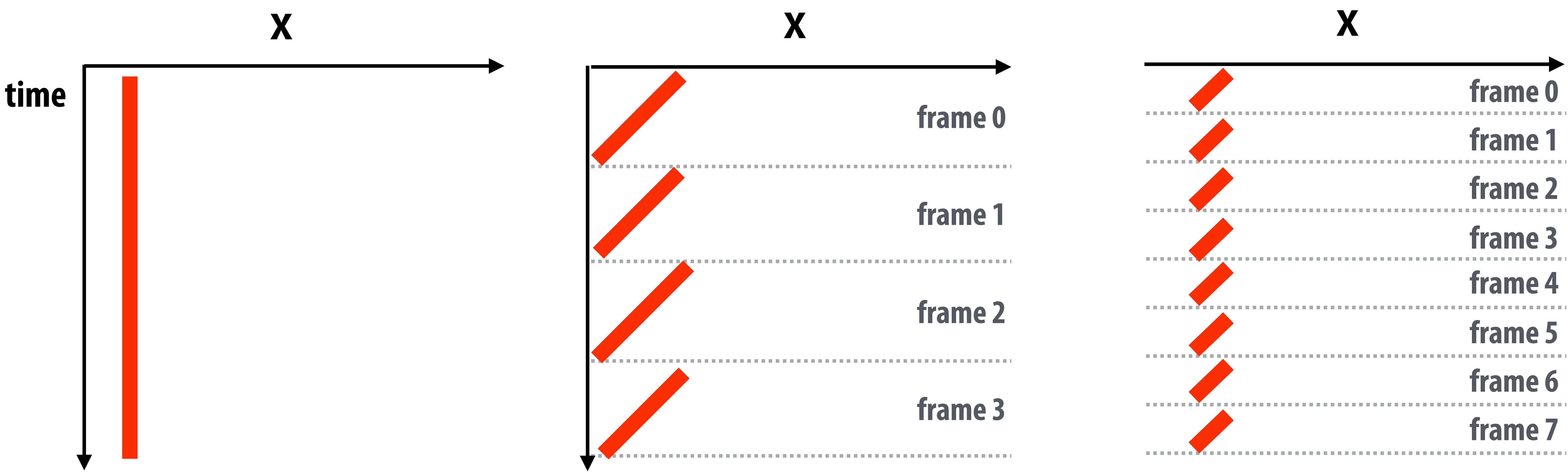
Case 1: object moving from left to right, eye moving continuously to track object (eye moving relative to display!)

Light from display (image is updated each frame)

Explanation: since eye is moving, object's position is relatively constant relative to eye (as it should be, eye is tracking it). But due discrete frame rate, object falls behind eye, causing a smearing/strobing effect ("choppy" motion blur). Recall from earlier slide: 90 degree motion, with 50 ms latency results in 4.5 degree smear

Spacetime diagrams adopted from presentations by Michael Abrash

Reducing judder: increase frame rate



Case 1: continuous ground truth

red object moving left-to-right and eye moving to track object
OR
red object stationary but head moving and eye moving to track object

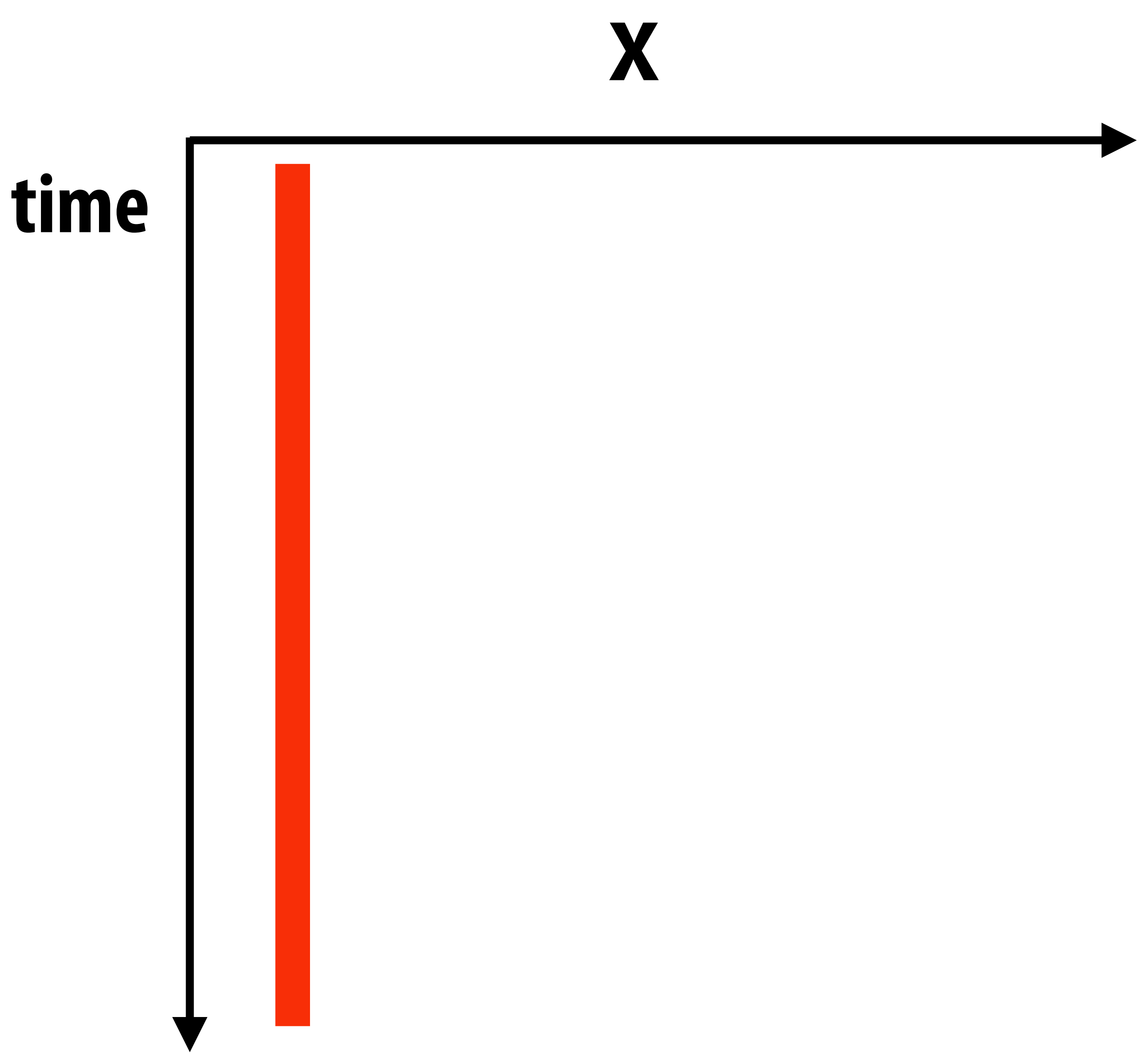
Light from display
(image is updated each frame)

Light from display
(image is updated each frame)

Higher frame rate results in closer approximation to ground truth

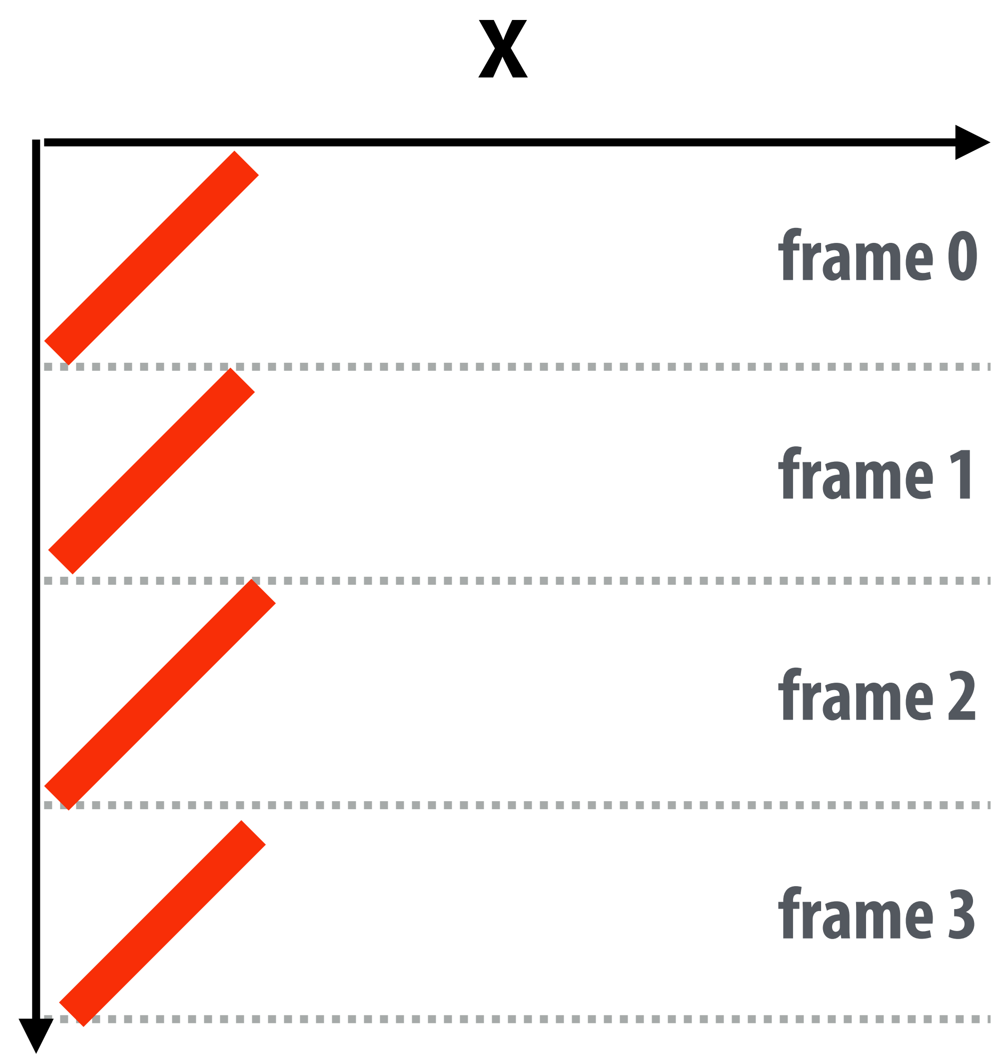
Spacetime diagrams adopted from presentations by Michael Abrash

Reducing judder: low persistence display

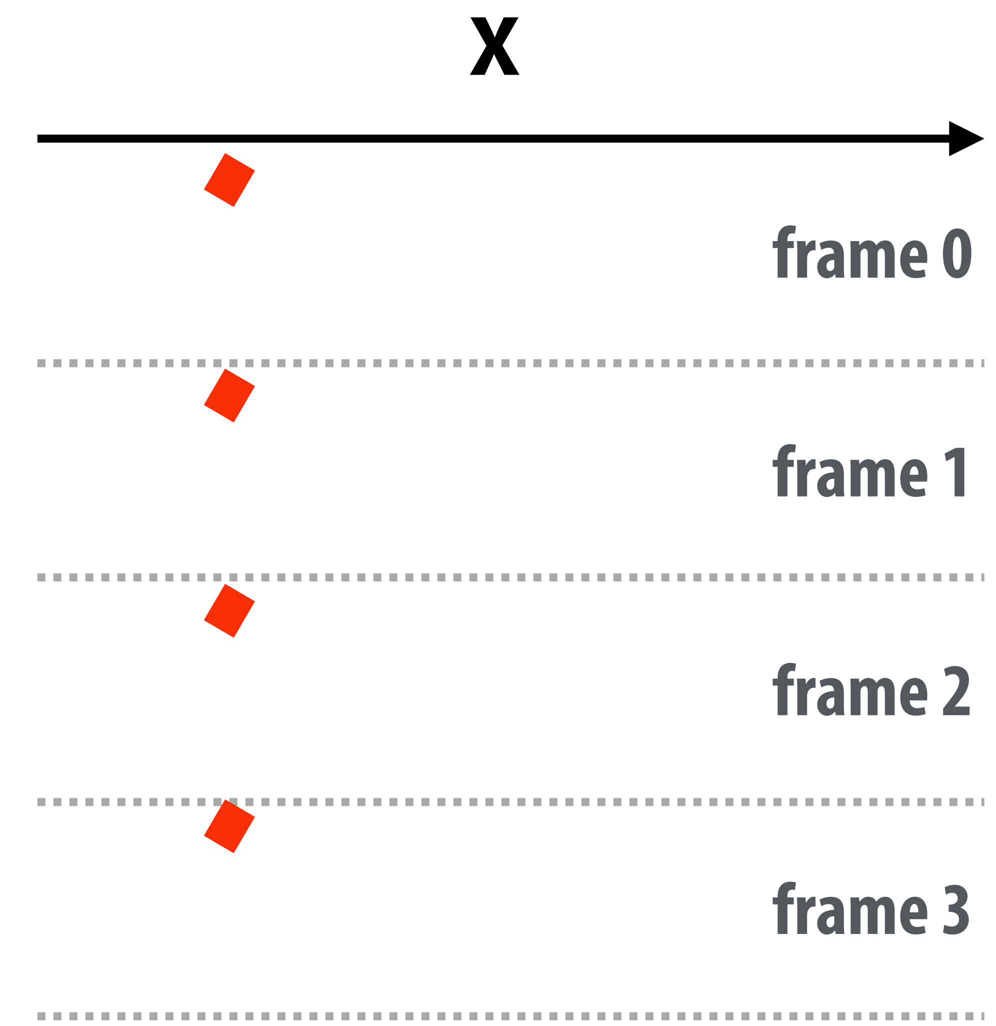


Case 1: continuous ground truth

red object moving left-to-right and
eye moving to track object
OR
red object stationary but head moving
and eye moving to track object



Light from full-persistence display



Light from low-persistence display

Full-persistence display: pixels emit light for entire frame
Low-persistence display: pixels emit light for small fraction of frame
Oculus DK2 OLED low-persistence display
- 75 Hz frame rate (~13 ms per frame)
- Pixel persistence = 2-3ms

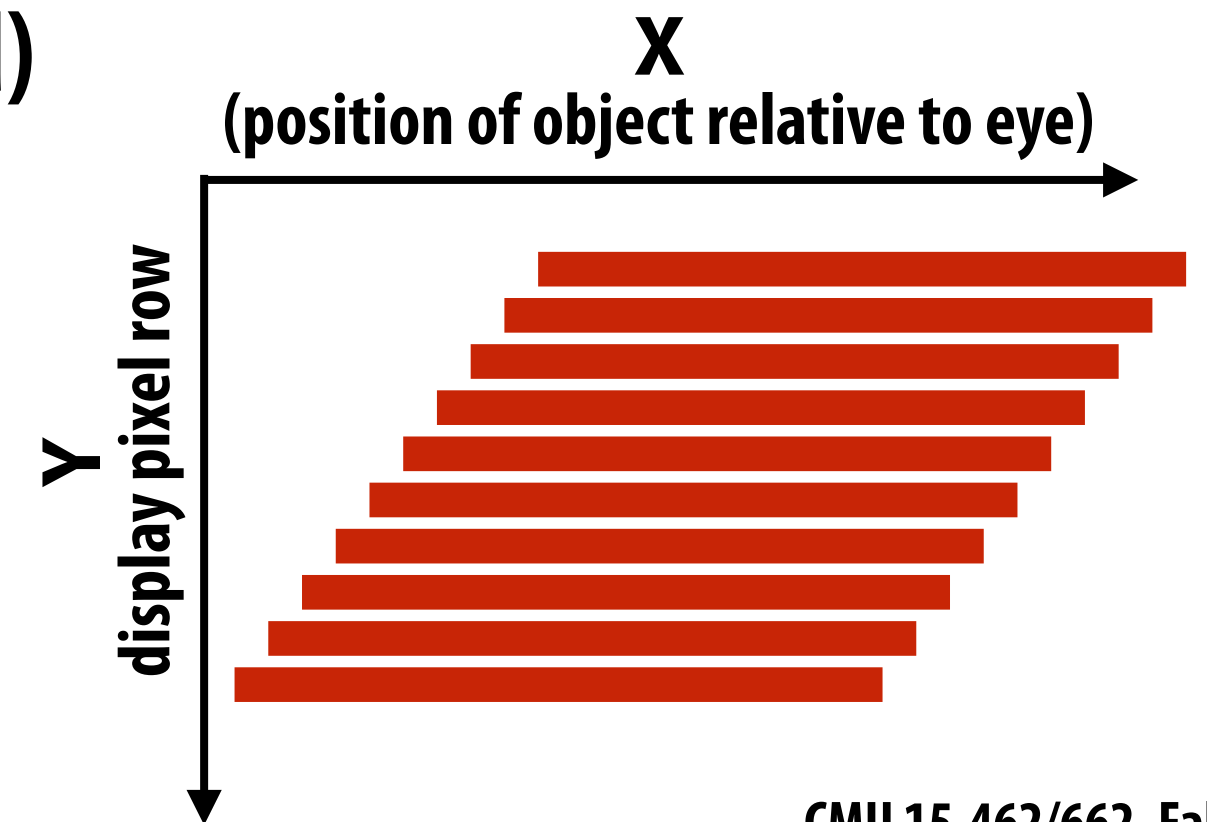
Spacetime diagrams adopted from presentations by Michael Abrash

Artifacts due to rolling OLED backlight

- Image rendered based on scene state at time t_0
- Image sent to display, ready for output at time $t_0 + \Delta t$
- "Rolling backlight" OLED display lights up rows of pixels in sequence
  - Let *r* be amount of time to "scan out" a row
  - Row 0 photons hit eye at $t_0 + \Delta t$
  - Row 1 photos hit eye at $t_0 + \Delta t + r$
  - Row 2 photos hit eye at $t_0 + \Delta t + 2r$
- Implication: photons emitted from bottom rows of display are "more stale" than photos from the top!
- Consider <u>eye moving horizontally relative to display</u> (e.g., due to head movement while tracking square object that is stationary in world)

Result: perceived shear!

Recall rolling shutter effects on modern digital cameras.



Compensating for rolling backlight

- Perform post-process shear on rendered image
  - Similar to previously discussed barrel distortion and chromatic warps
  - Predict head motion, assume fixation on static object in scene
    - Only compensates for shear due to head motion, not object motion

- Render each row of image at a different time (the predicted time photons will hit eye)
  - Suggests exploration of different rendering algorithms that are more amenable to fine-grained temporal sampling, e.g., ray caster? (each row of camera rays samples scene at a different time)

Increasing frame rate using re-projection

- **Goal: maintain as high a frame rate as possible under challenging rendering conditions:**
  - Stereo rendering: both left and right eye views
  - High-resolution outputs
  - Must render extra pixels due to barrel distortion warp
  - Many "rendering hacks" (bump mapping, billboards, etc.) are less effective in VR so rendering must use more expensive techniques

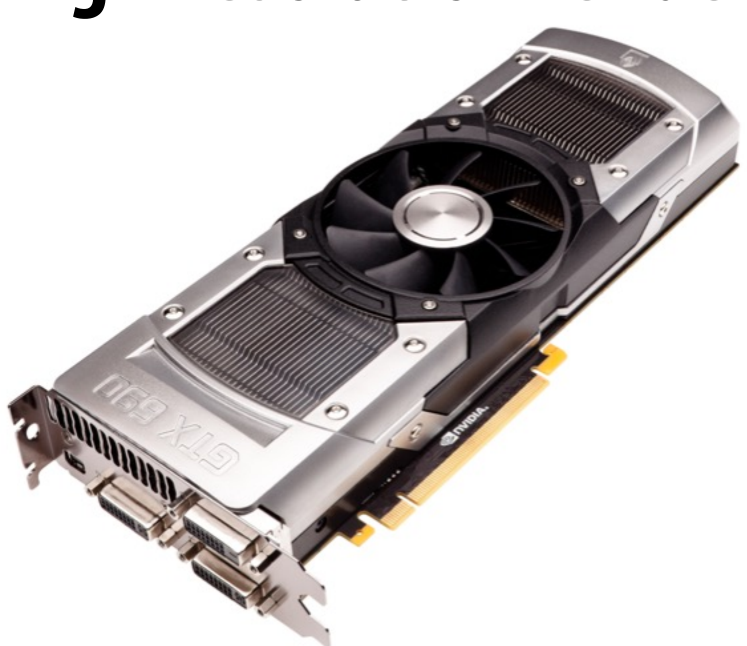
- **Researchers experimenting with reprojection-based approaches to improve frame rate (e.g., Oculus' "Time Warp")**
  - Render using conventional techniques at 30 fps, reproject (warp) image to synthesize new frames based on predicted head movement at 75 fps
  - Potential for image processing hardware on future VR headsets to perform high frame-rate reprojection based on gyro/accelerometer

Near-future VR system components

Low-latency image processing
for subject tracking

High-resolution, high-frame rate,
wide-field of view display

Massive parallel computation for
high-resolution rendering

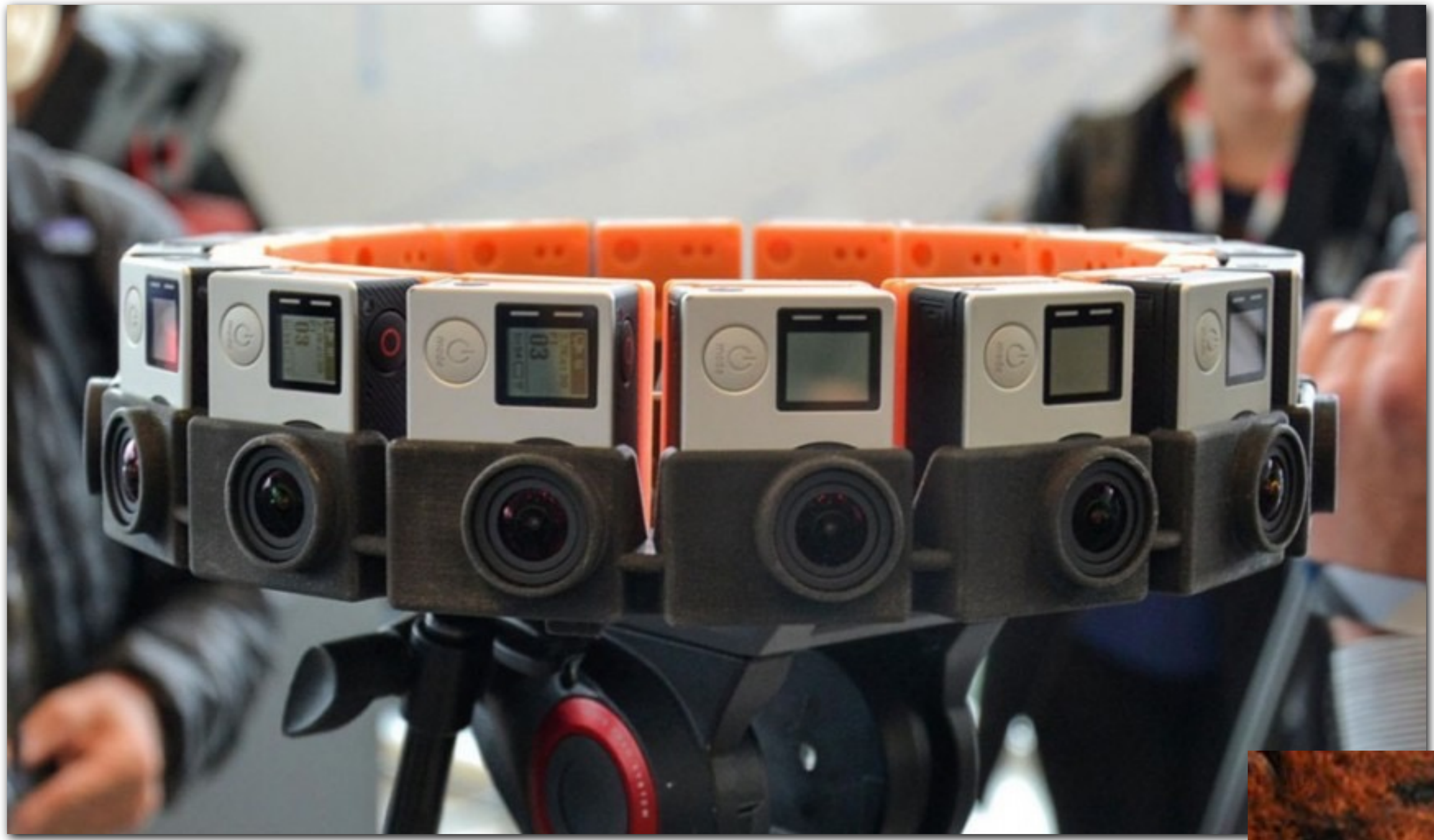


Exceptionally high bandwidth connection
between renderer and display:
e.g., 4K x 4K per eye at 90 fps!

In headset motion/accel
sensors + eye tracker

On headset graphics
processor for sensor
processing and re-
projection

Interest in acquiring VR content



Google's JumpVR video:
16 4K GoPro cameras

Consider challenge of:
Registering/3D align video stream (on site)
Broadcast encoded video stream across the country to 50 million viewers

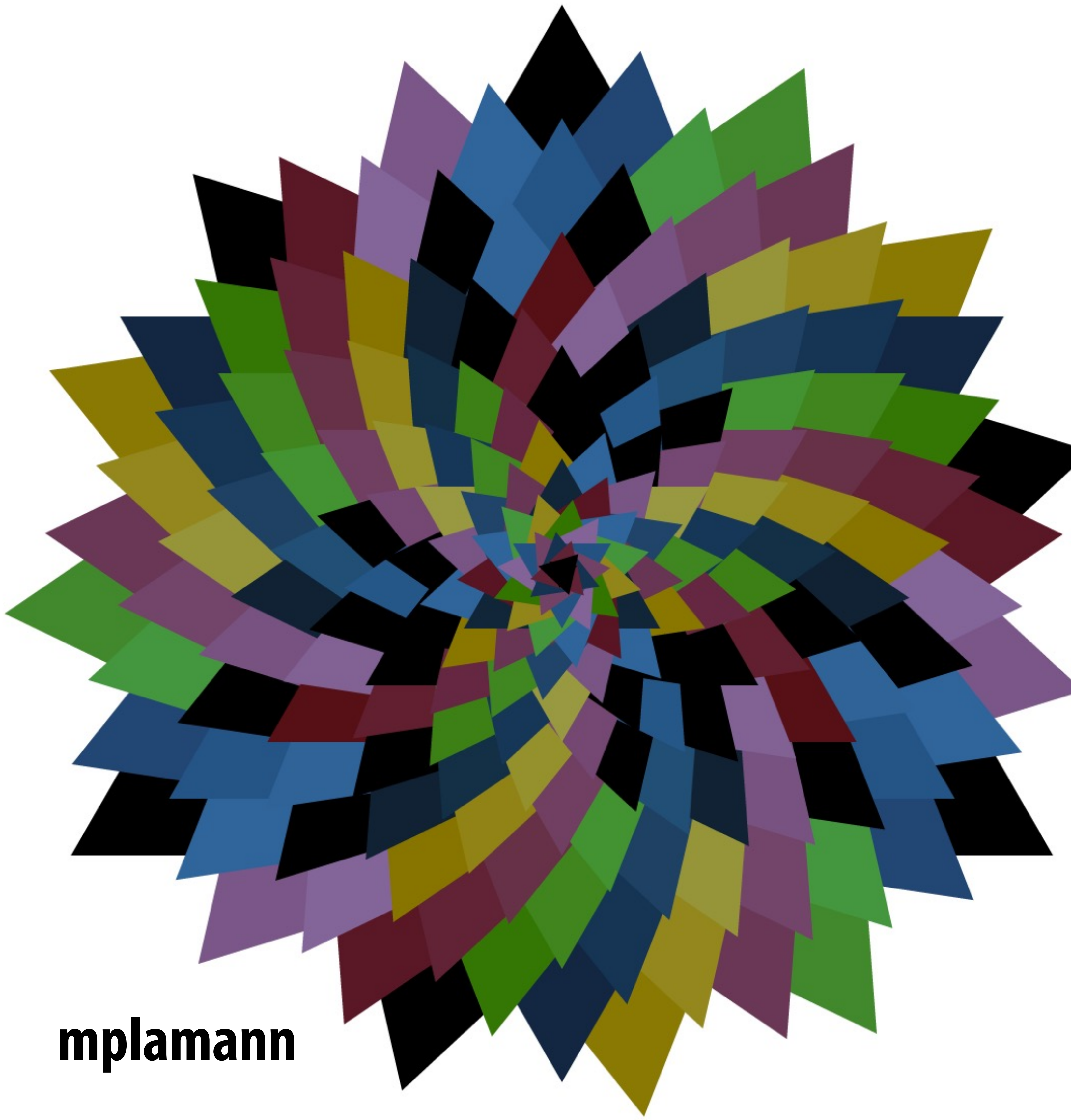
Lytro Immerge
(leveraging light field camera technology to acquire VR content)

Summary: virtual reality presents many new challenges for graphics systems developers

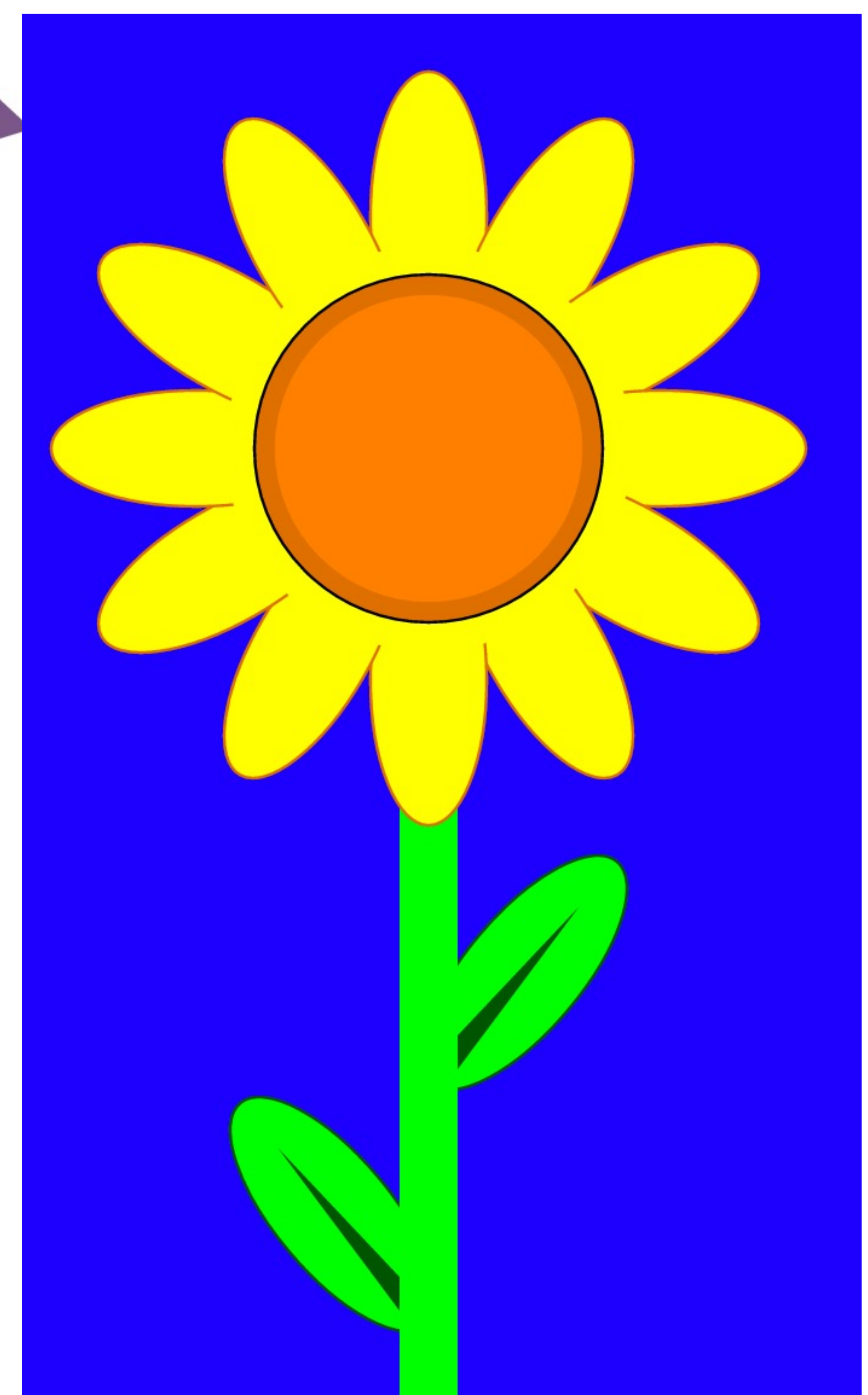
- Major goal: minimize latency of head movement to photons
  - Requires low latency tracking (not discussed today)
    - Combination of external camera image processing (vision) and high rate headset sensors
    - Heavy use of prediction
  - Requires high-performance rendering
    - High-resolution, wide field-of-view output
    - High frame-rate
    - Rendering must compensate for constraints of display system:
      - Optical distortion (geometric, chromatic)
      - Temporal offsets in rows of pixels
- Significant research interest in display technologies that are alternatives to flat screens with lenses in front of them

Course wrap up

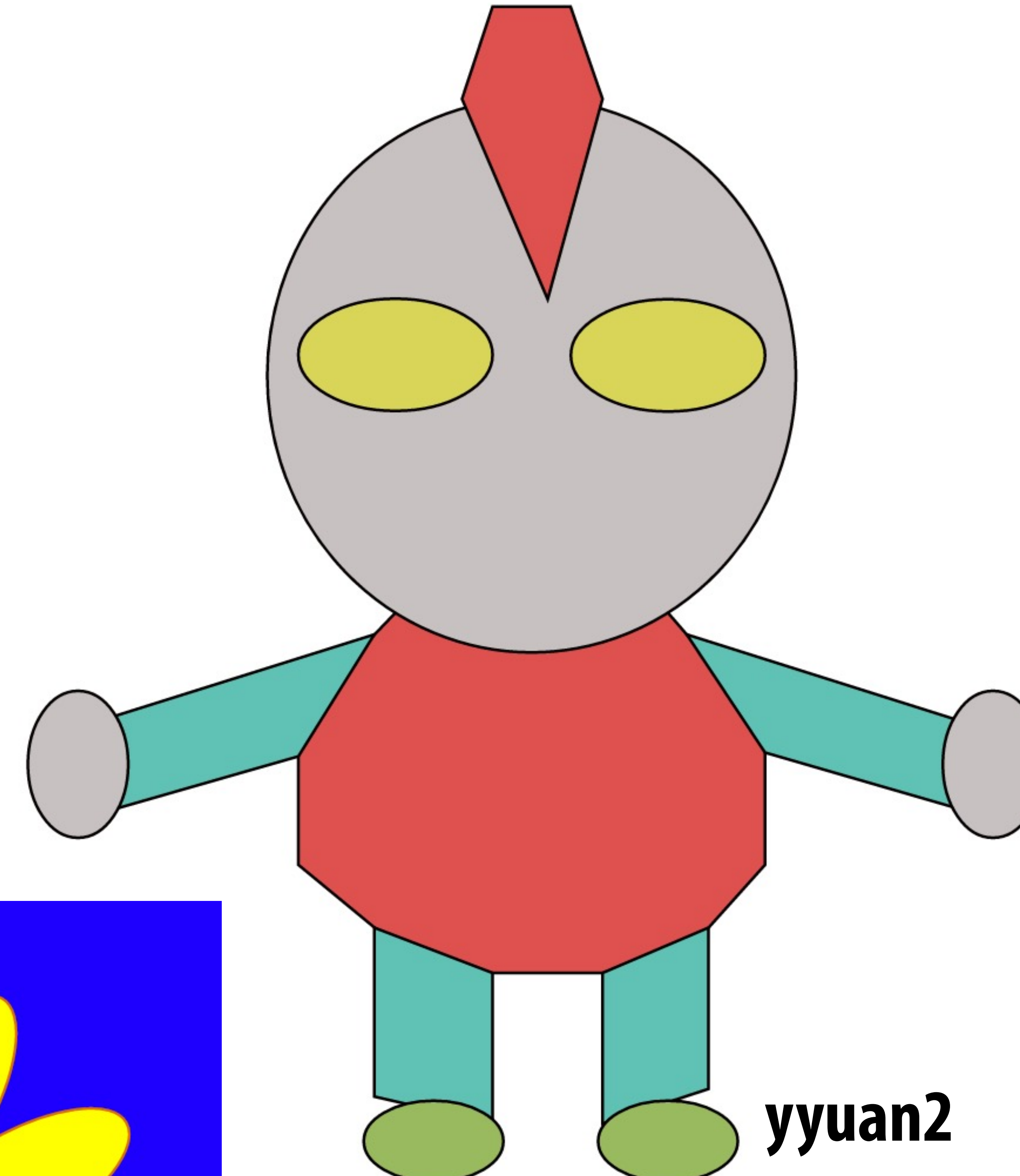
Student project demo reel!



mplamann

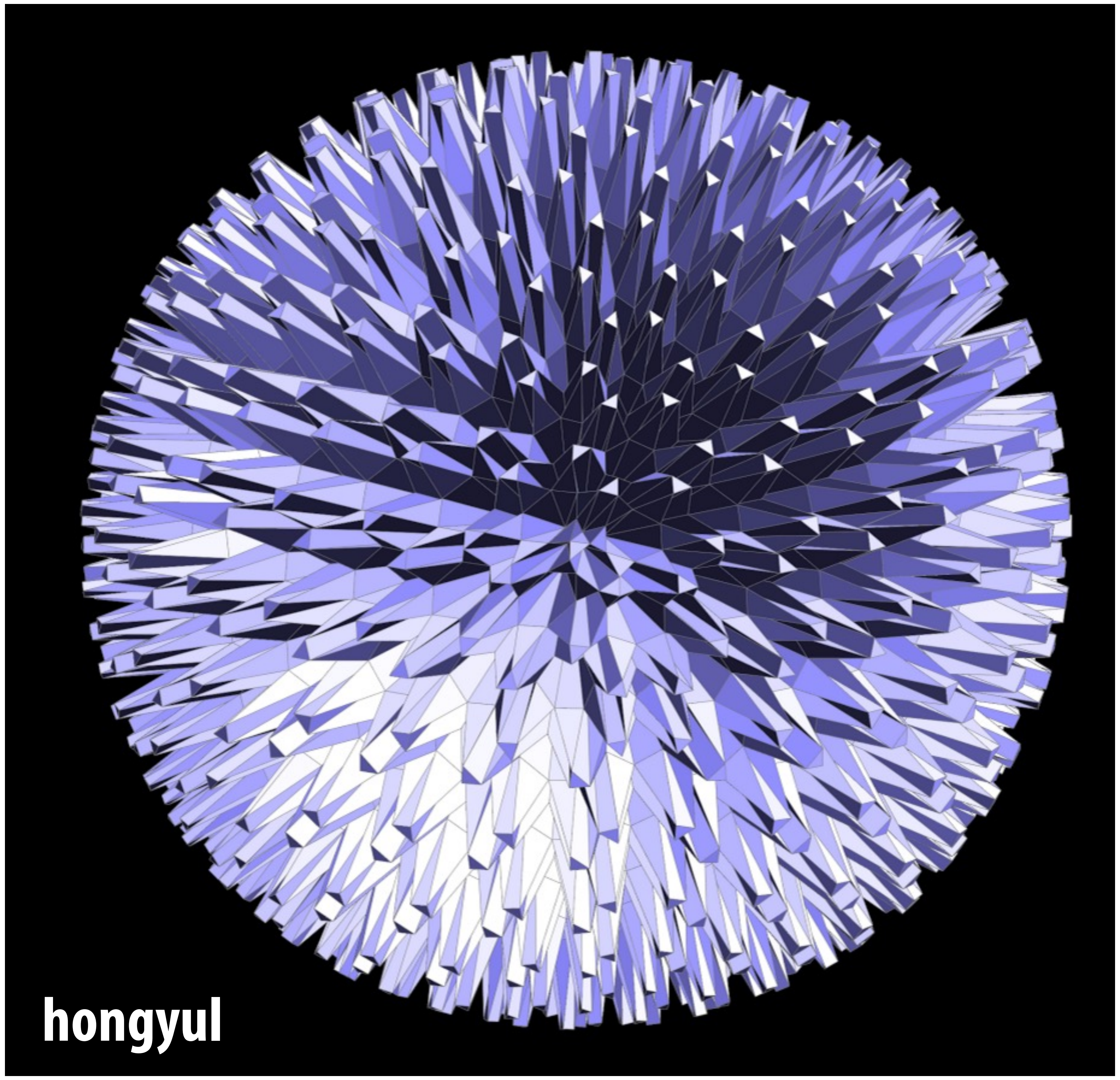


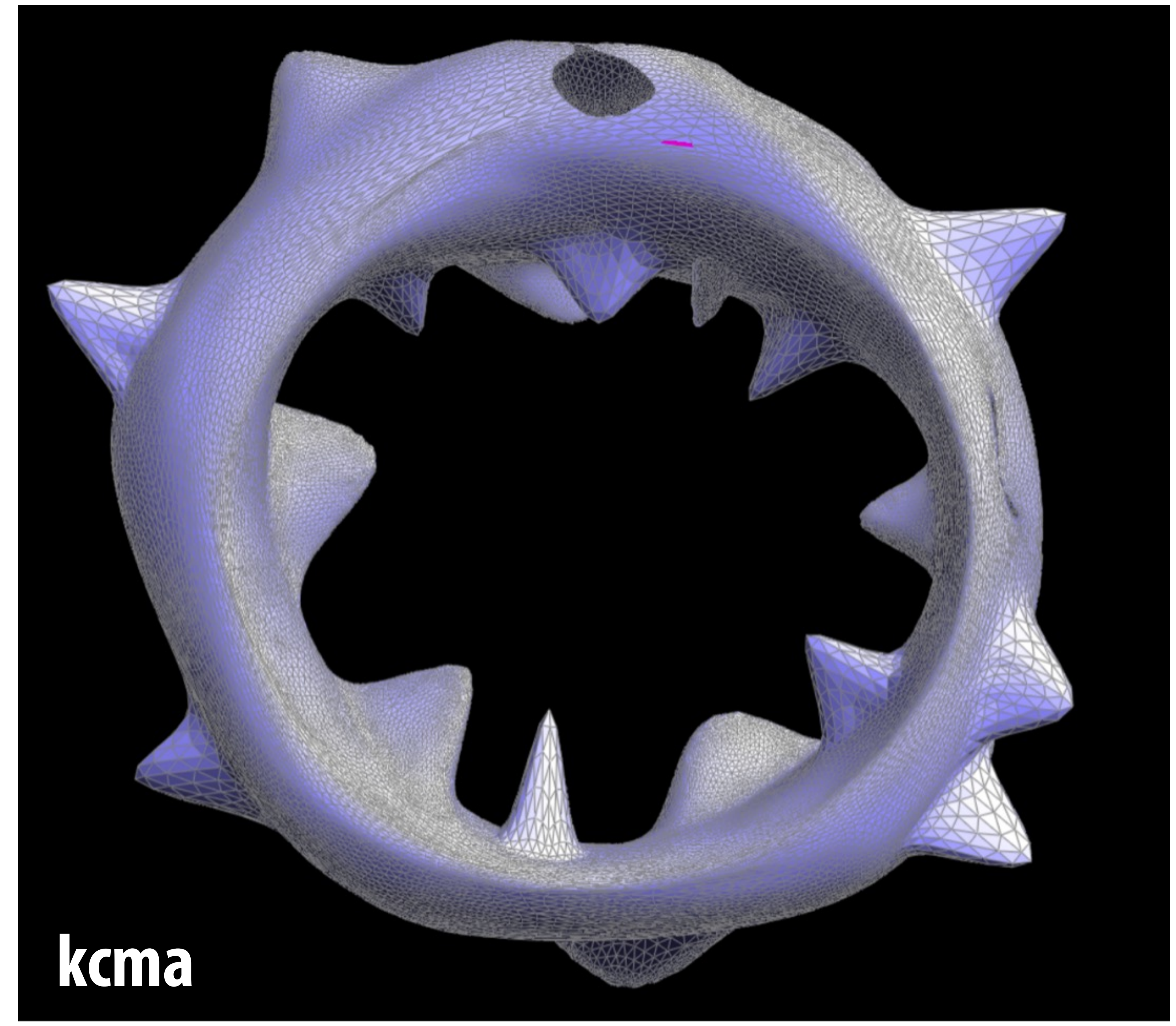
jmrichar

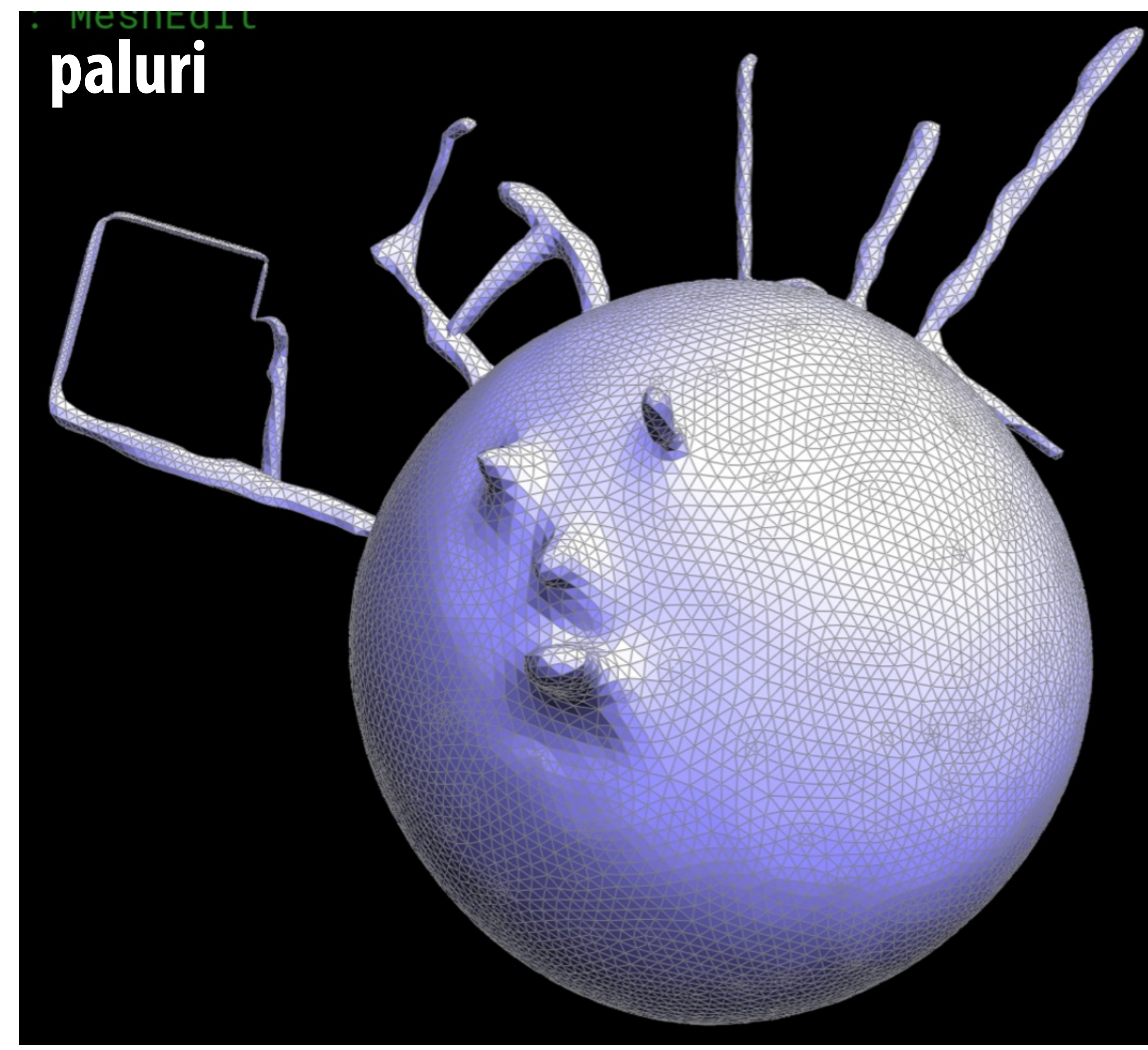


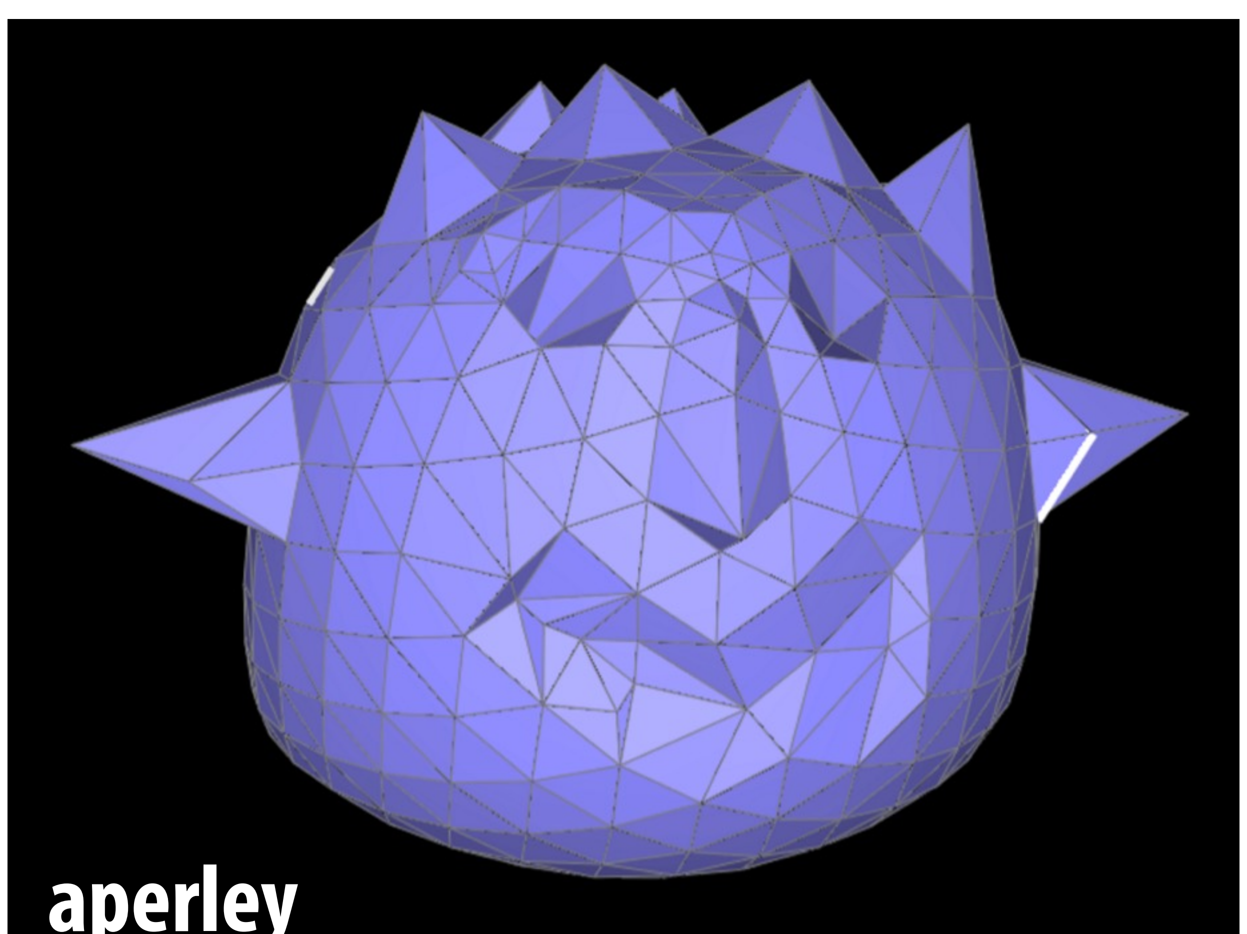
yyuan2

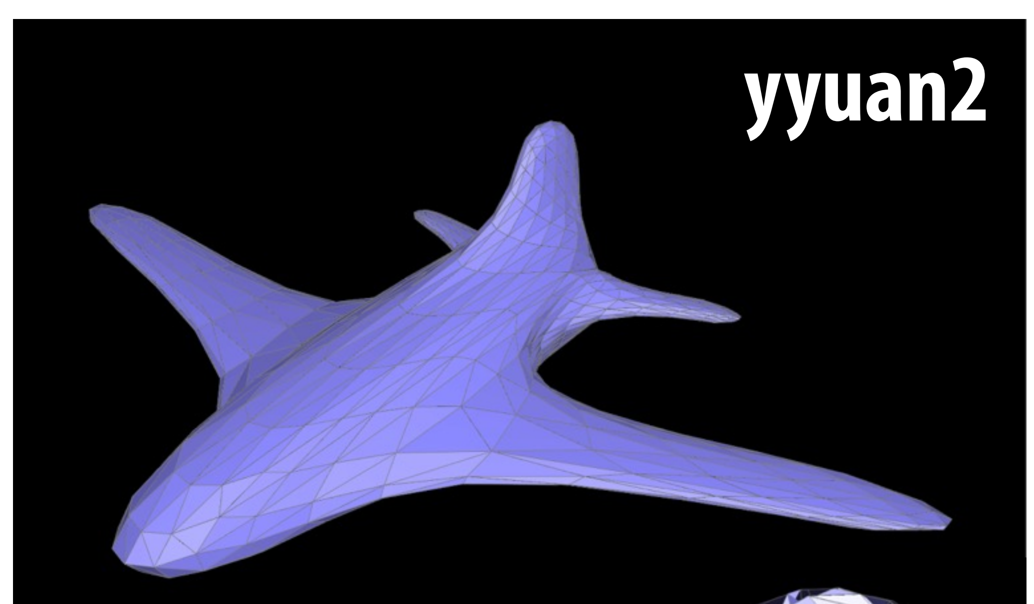
Student project demo reel!



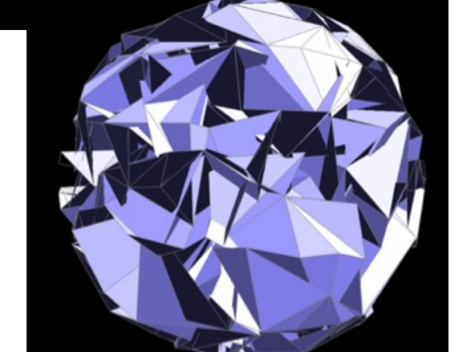




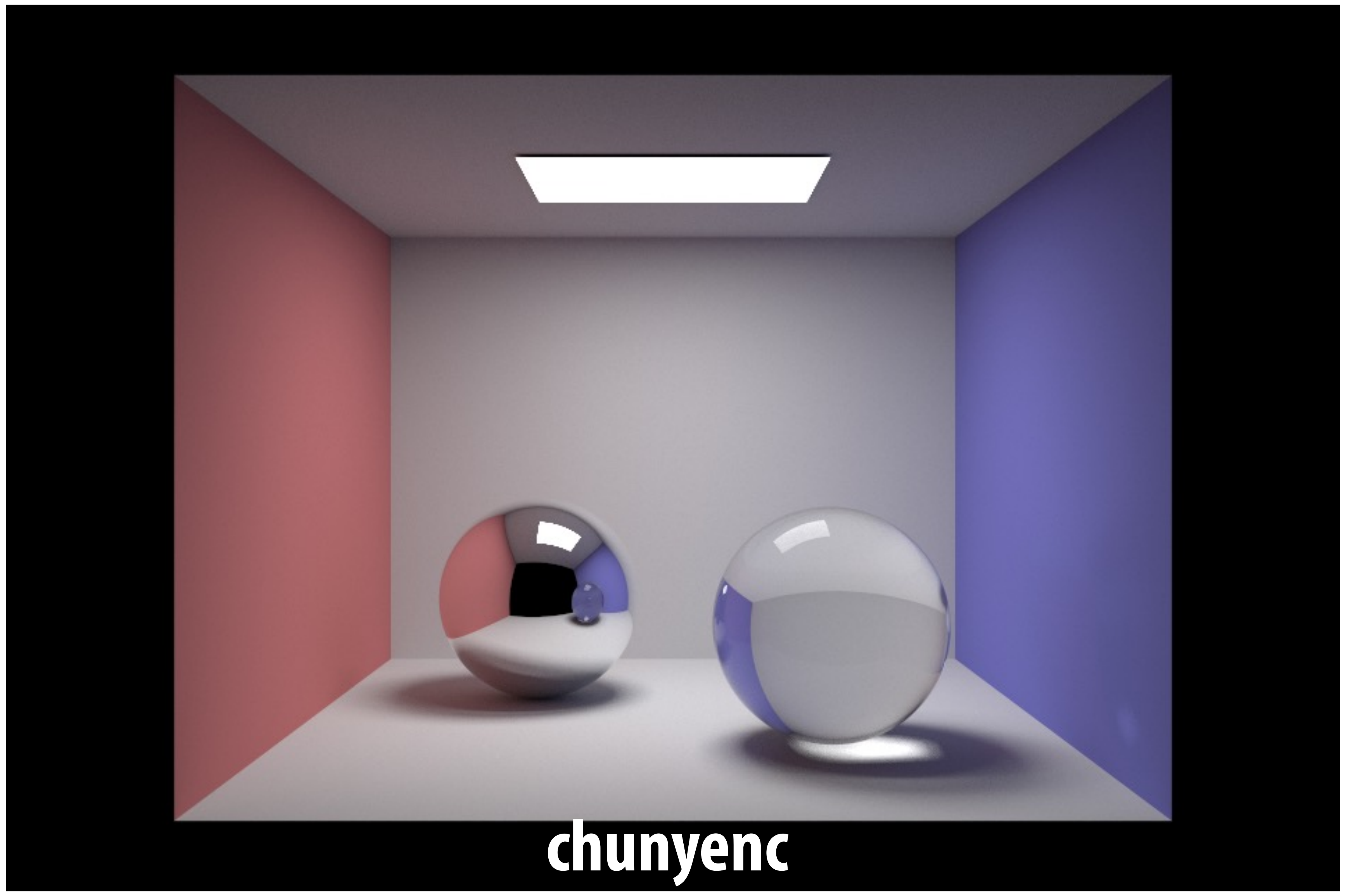


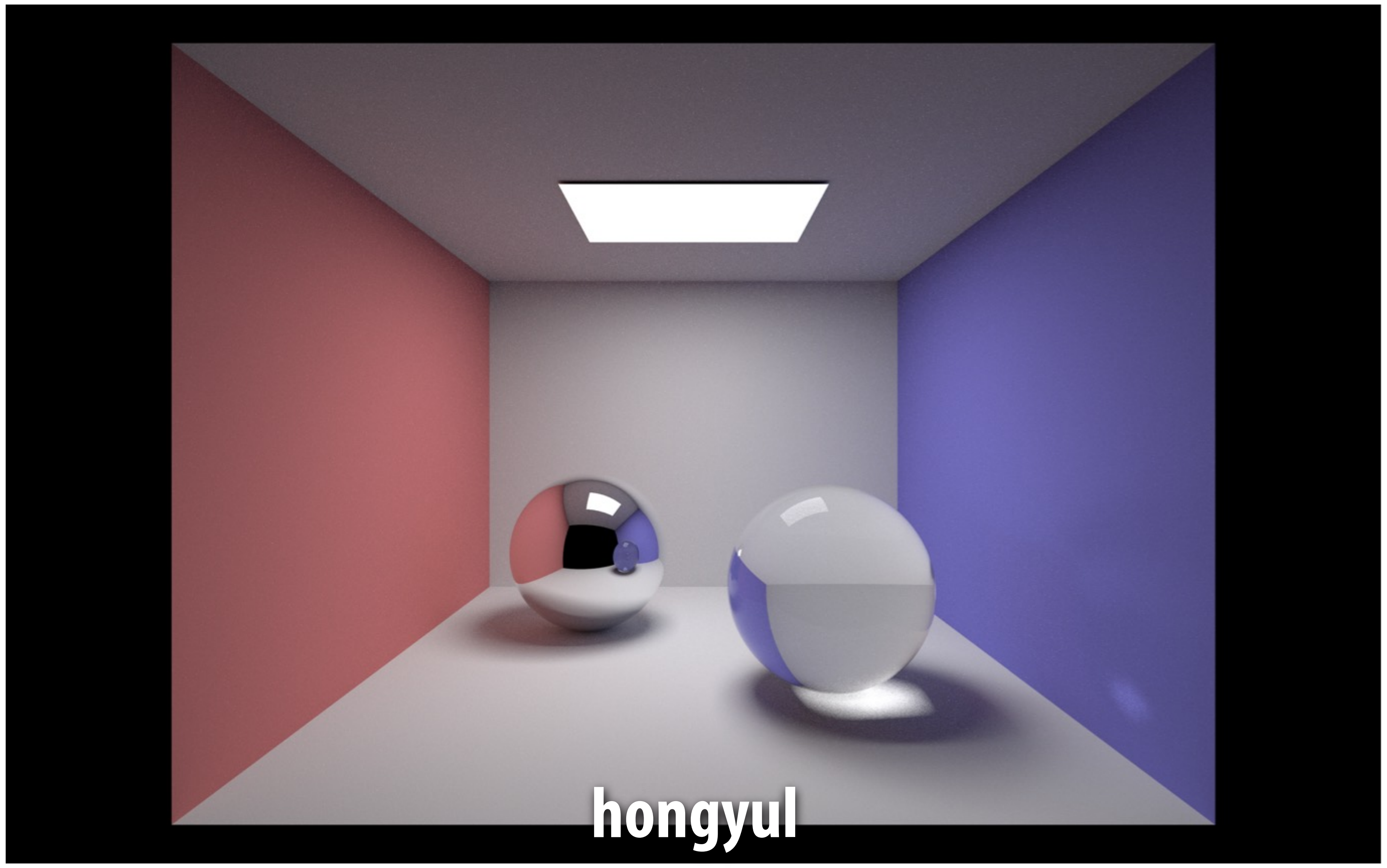


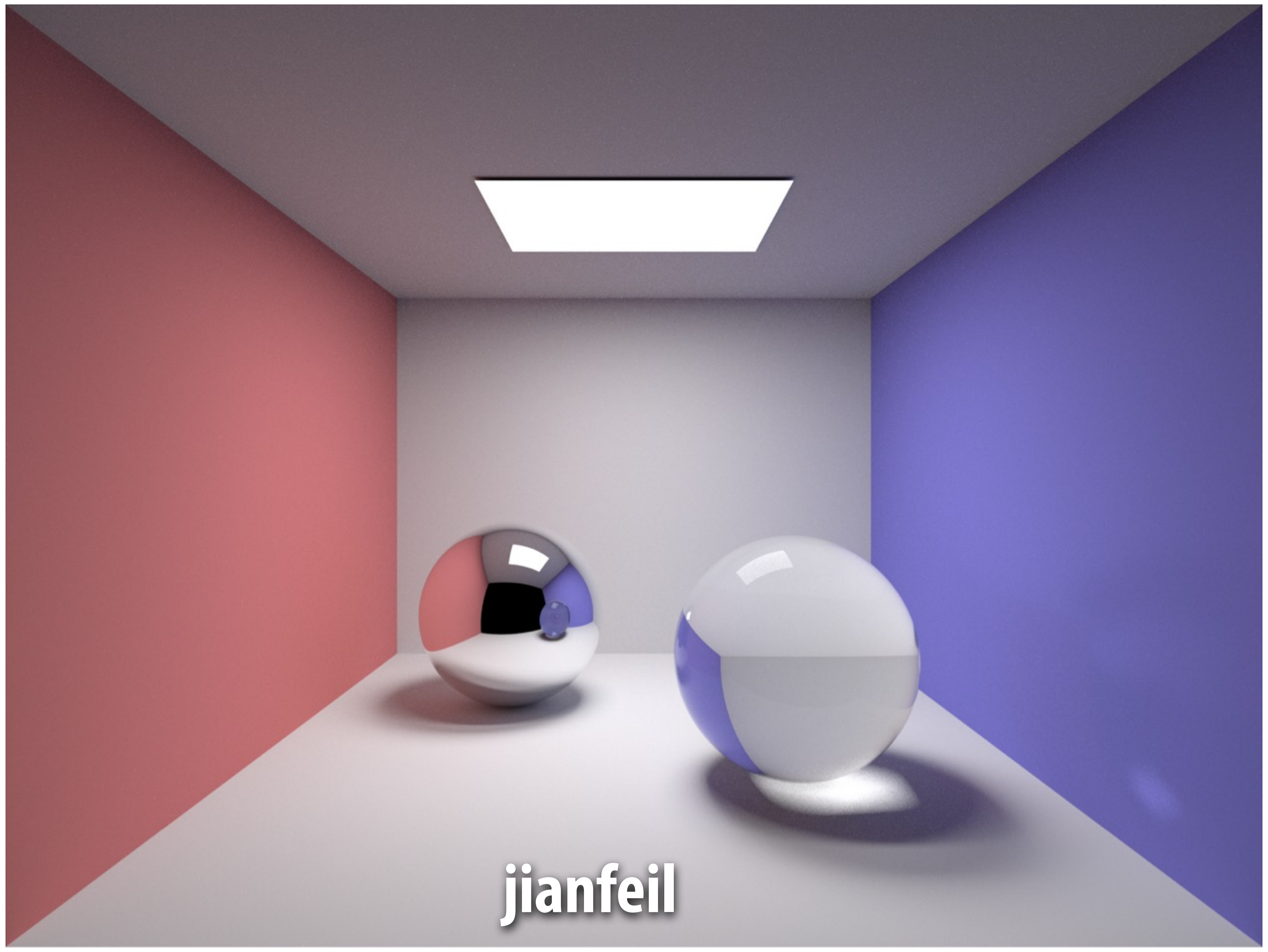
jianfeil

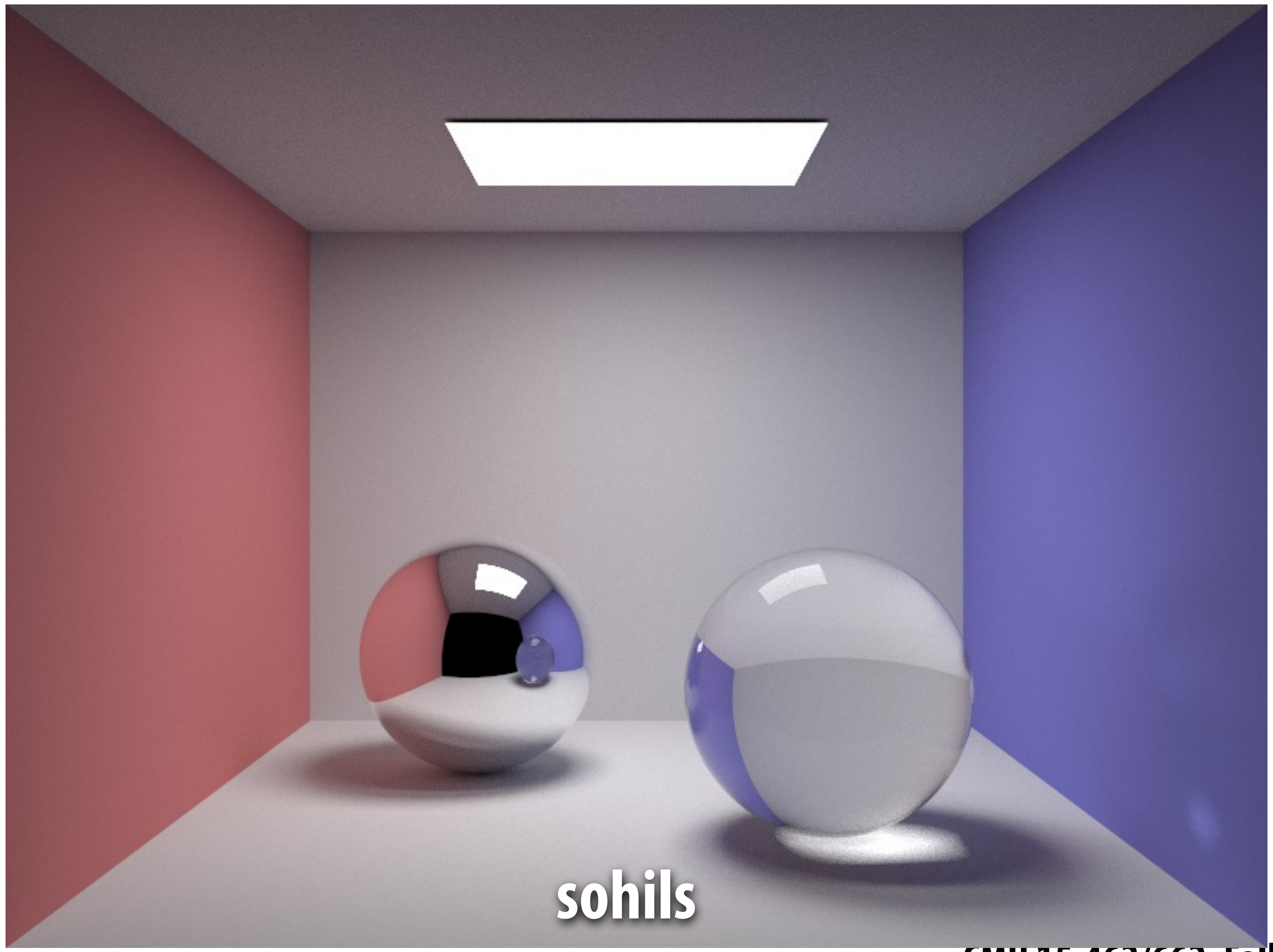


Student project demo reel!



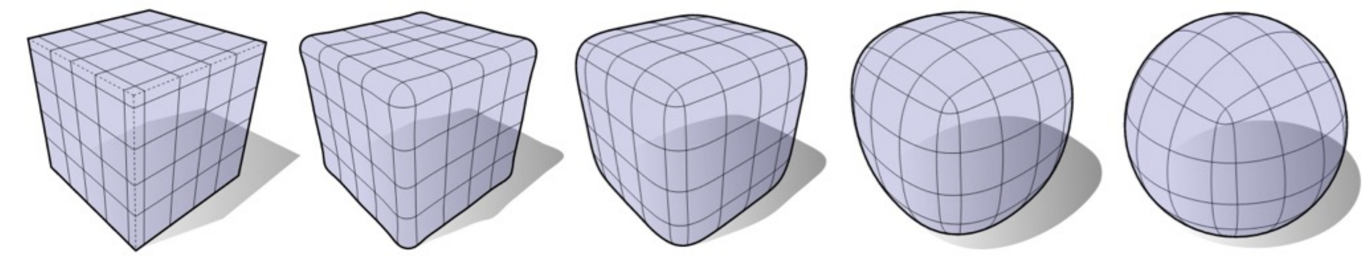






Other cool graphics-related courses

- 15-869: Discrete Differential Geometry (Keenan Crane)



- 15-463: Computational Photography
- 15-467: Simulation Methods for Animation and Digital Fabrication (Stelian Coros)

- 15-465: Animation Art and Technology (Hodgins/Duesing)
- 15-661: Interaction and Expression using the Pausch Bridge Lighting
- 15-418/618: Parallel Computer Architecture and Programming (Kayvon Fatahalian)

TAs and independent study!

- 15-462 next semester is looking for TAs!
  - Email us if interested, and we'll direct you to Prof. Pollard

- Students that did well in 462 have a great foundation for moving on to independent study or research in graphics
  - Come talk to Keenan and I!

Beyond assignments and exams

- Come talk to Keenan or I (or other professors) about participating in research!
- Consider a senior thesis!
- Pitch a seed idea to Project Olympus
- Get involved with organizations like Hackathon or ScottyLabs

Thanks for being a great class!
See you on Monday! (study hard, but don't stress too much)

Credit: Inside Out (Pixar)