Lecture 23:

Rendering Challenges of VR

Visual Computing Systems
CMU 15-869, Fall 2014

Today

- **Basics of Oculus Rift implementation of VR**
  - Rift is best documented of modern prototypes
  - Focus on rendering issues for these emerging platforms

Oculus Rift DK2

Sony Project Morpheus

Oculus Rift DK2 Headset

Image credit: ifixit.com

Oculus Rift DK2 Headset

Image credit: ifixit.com

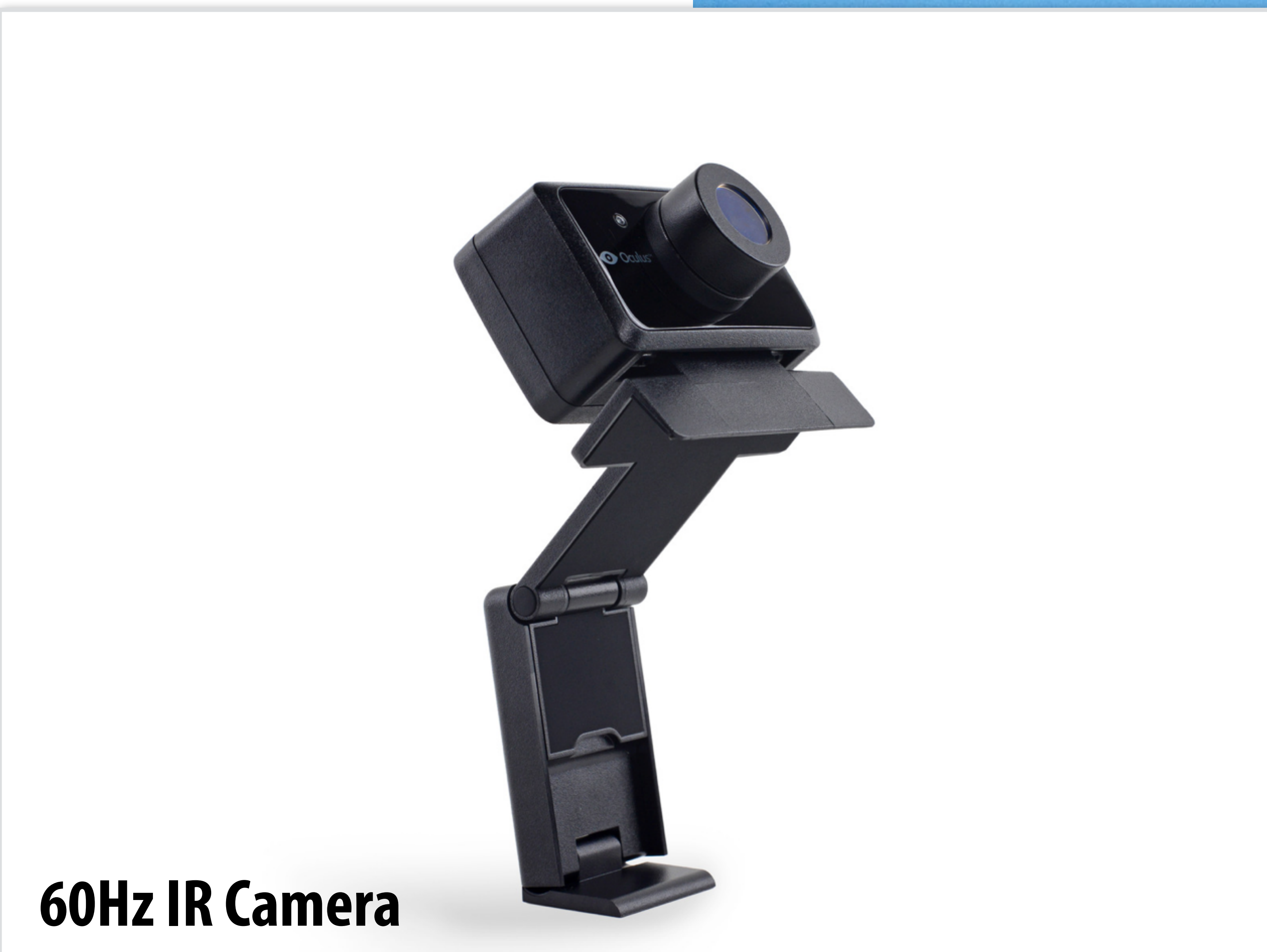
Oculus Rift DK2 display

5.7” 1920 x 1080 OLED display
75 Hz refresh rate

(Same display as Galaxy Note 3)

Image credit: ifixit.com

Oculus DK2 IR camera and IR LEDs



60Hz IR Camera

Headset contains:

40 IR LEDs
Gyro + accelerometer
(1000Hz)

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 photos hit your eyes
    - Measure movement
    - Update scene/camera position
    - Render image
    - Transfer image to headset, then to display in headset
    - Display image on display
  - VR latency goal: 10-25 ms
    - Requires low-latency head tracking
    - Requires low-latency rendering and display

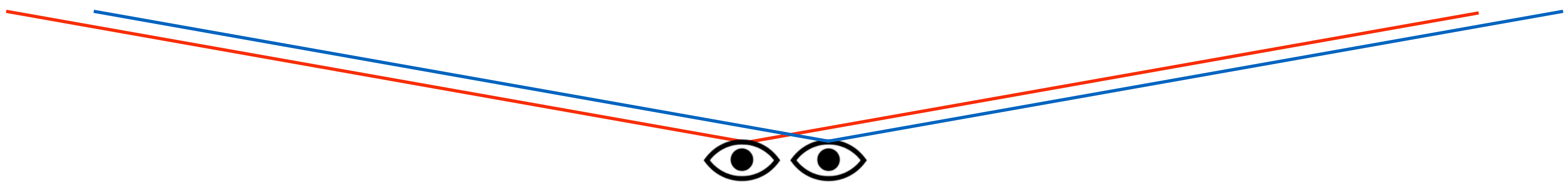
Latency thought experiment

- Consider a 1,000 x 1,000 spanning 100º field of view
  - 10 pixels per degree
- Assume:
  - You move your head 90 degrees in 1 second (modest speed)
  - Latency of system is 50 ms
- Therefore:
  - Displayed pixels are off by 4.5º ~ 45 pixels

Example credit: Michael Abrash

Requirement: high-resolution display

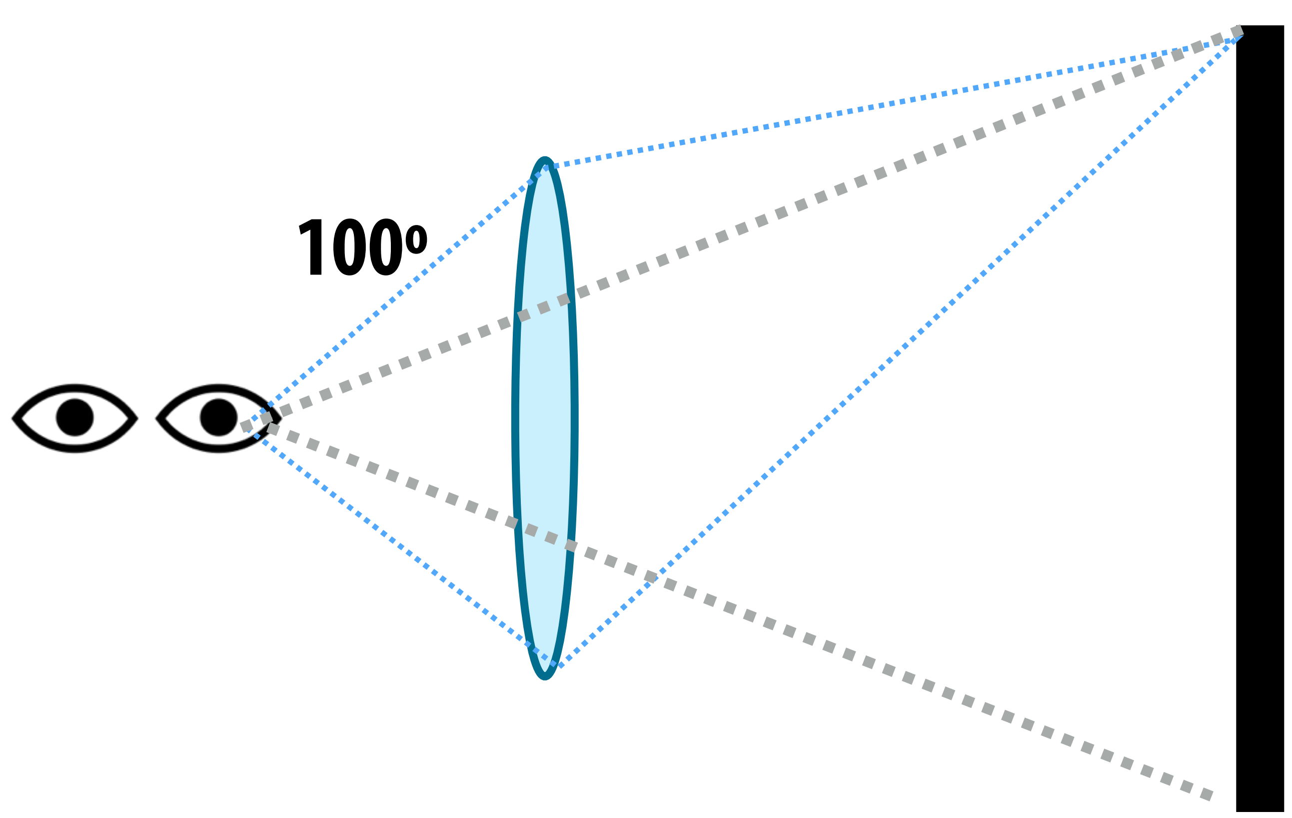
- Human: ~160 degree view of field per eye (~200 degrees overall)
  - Does not count rotation in socket

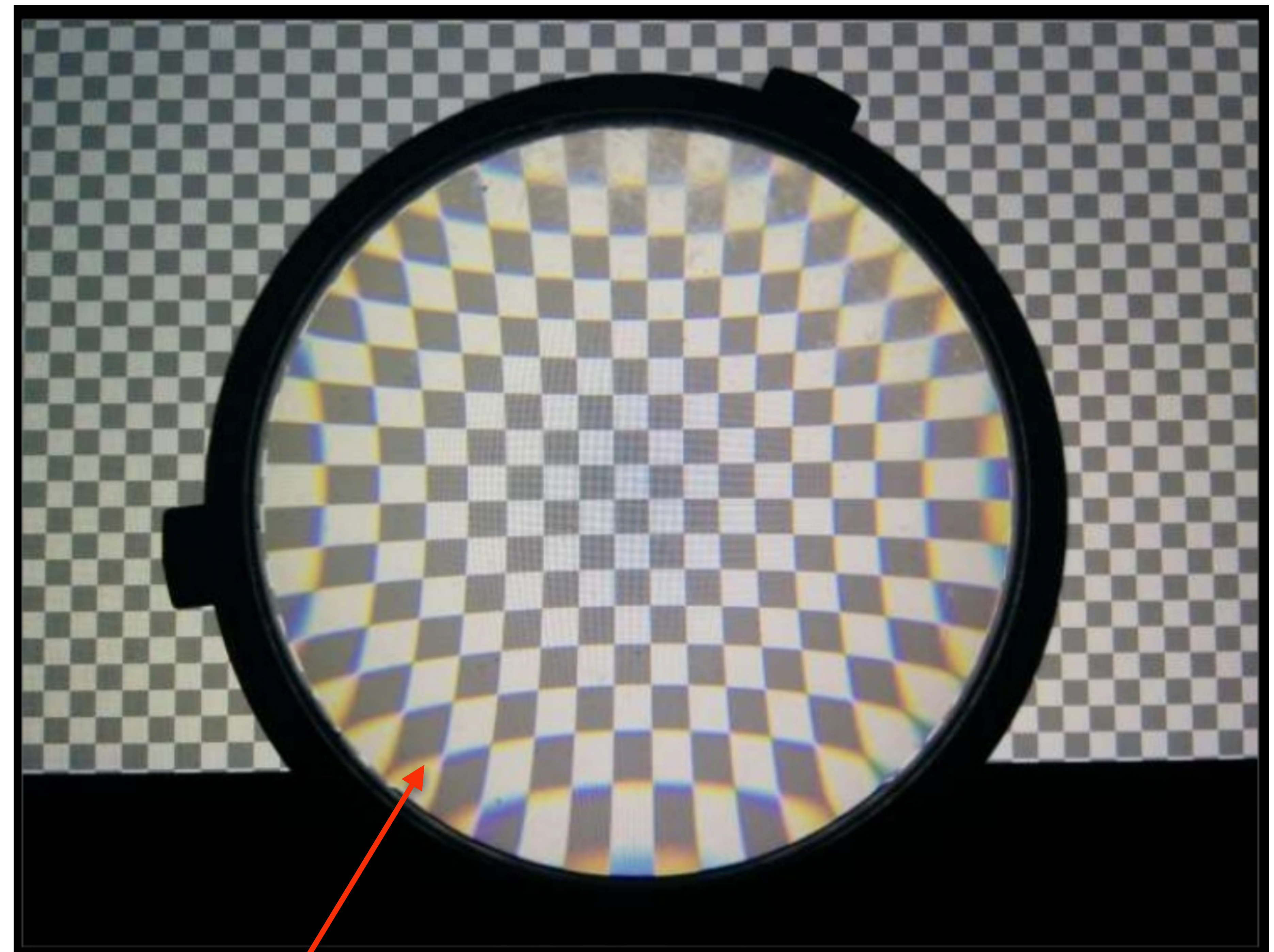


- Oculus DK2
  - Resolution: 1920 x 1080, 960 x 1080 per eye
  - 100 degree field of view (~10 pixels/degree)
- Compare to my Macbook Pro Retina display:
  - Resolution: 2880 x 1800
  - Spans ~ 30 degrees at normal viewing distance at a desk (~100 pixels/degree)
- Retina resolution VR display filling 160 degree field of view: 16K x 16K display per eye
  - Wow.
  - 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

Requirement: wide field of view



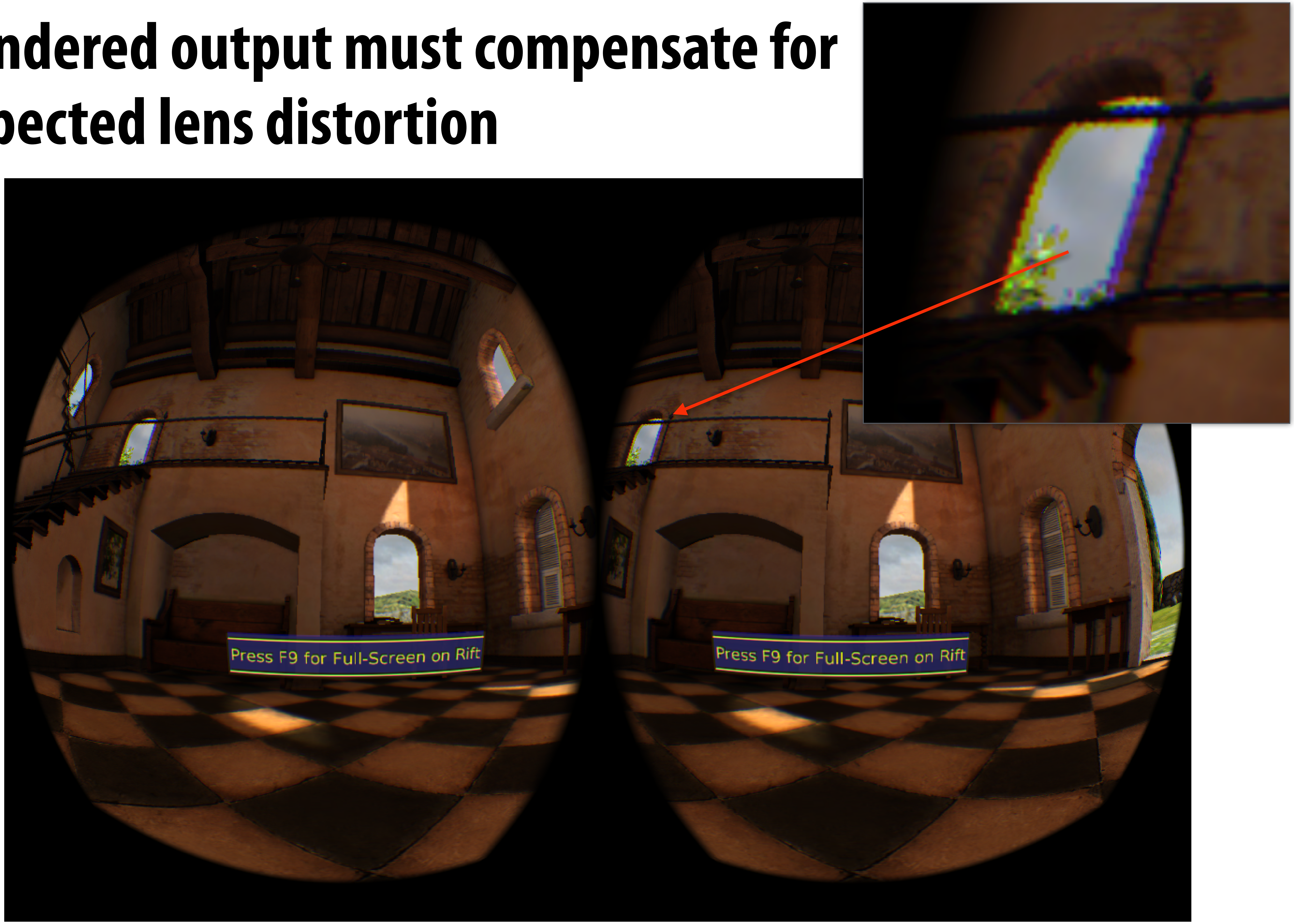


View through Oculus lens

- Lens introduces distortion
  - Pincushion distortion
  - Chromatic aberration (different wavelengths refract by different amount)

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

Rendered output must compensate for expected lens distortion



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

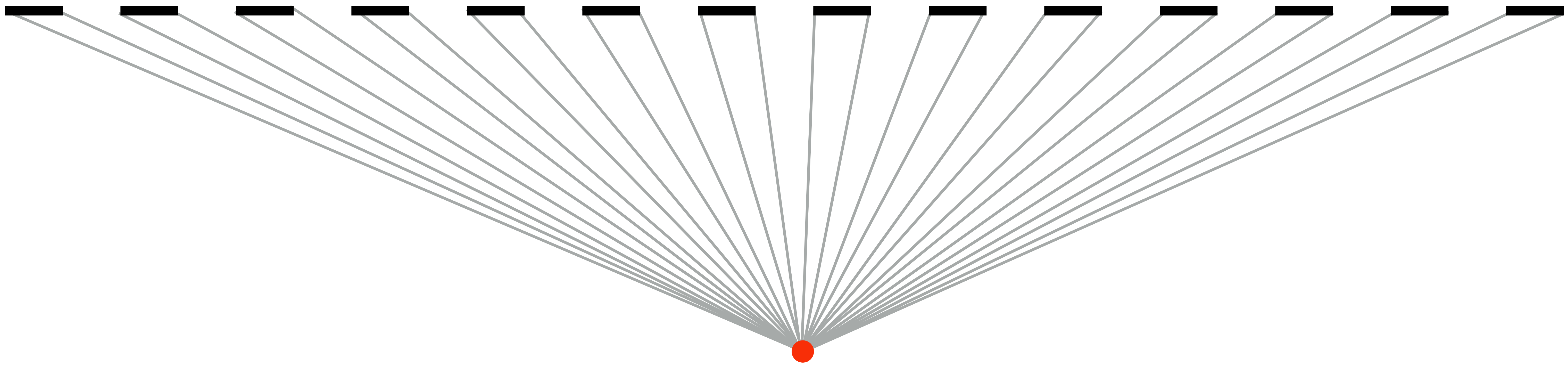
Step 2: distortion pass warps image so that it is viewed properly after lens distortion

Can apply different distortion to R, G, B to approximate correction to chromatic aberration

Image credit: Oculus VR developer guide

Challenge: rendering via planar projection

- Rasterization-based graphics is based on projection to plane
  - Reasonable for modest FOV, but distorts image under high FOV
  - Recall: VR rendering spans wide FOV

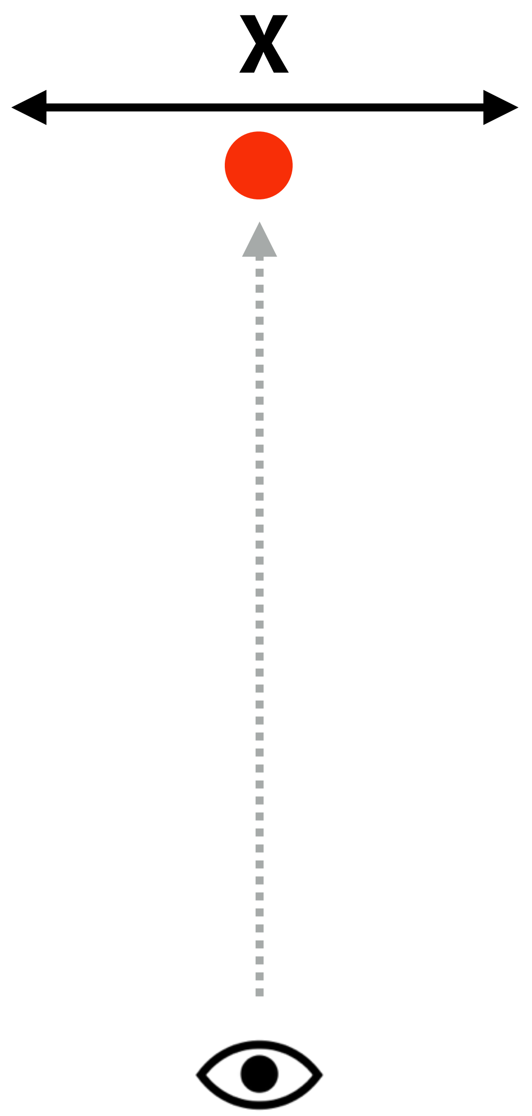


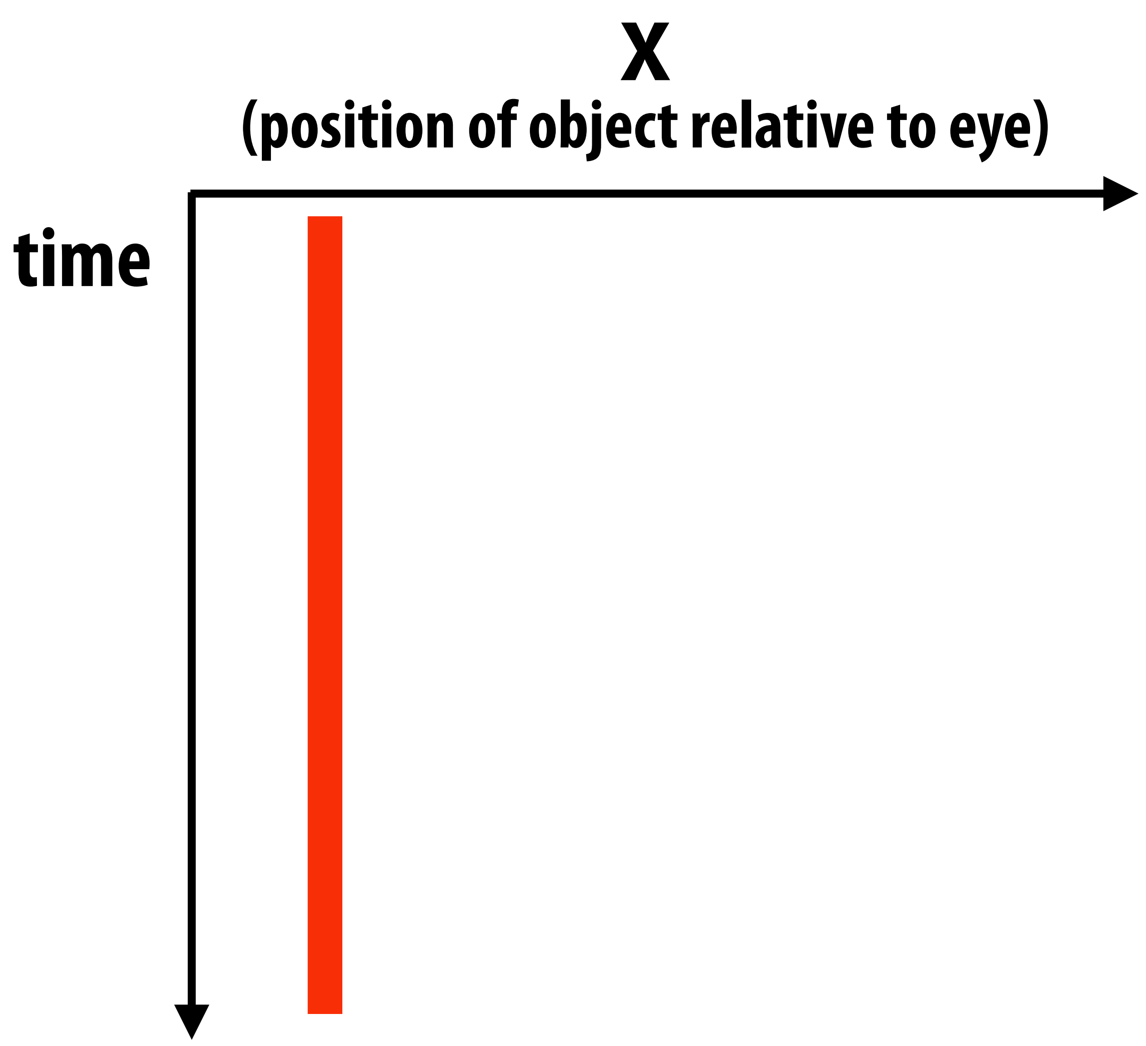
Pixels span larger angle in center of image

Possible improvements:
Consider curved displays, ray casting, rendering with piecewise linear projection plane (different plane per tile in a tiled renderer)

Image credit: Cass Everitt

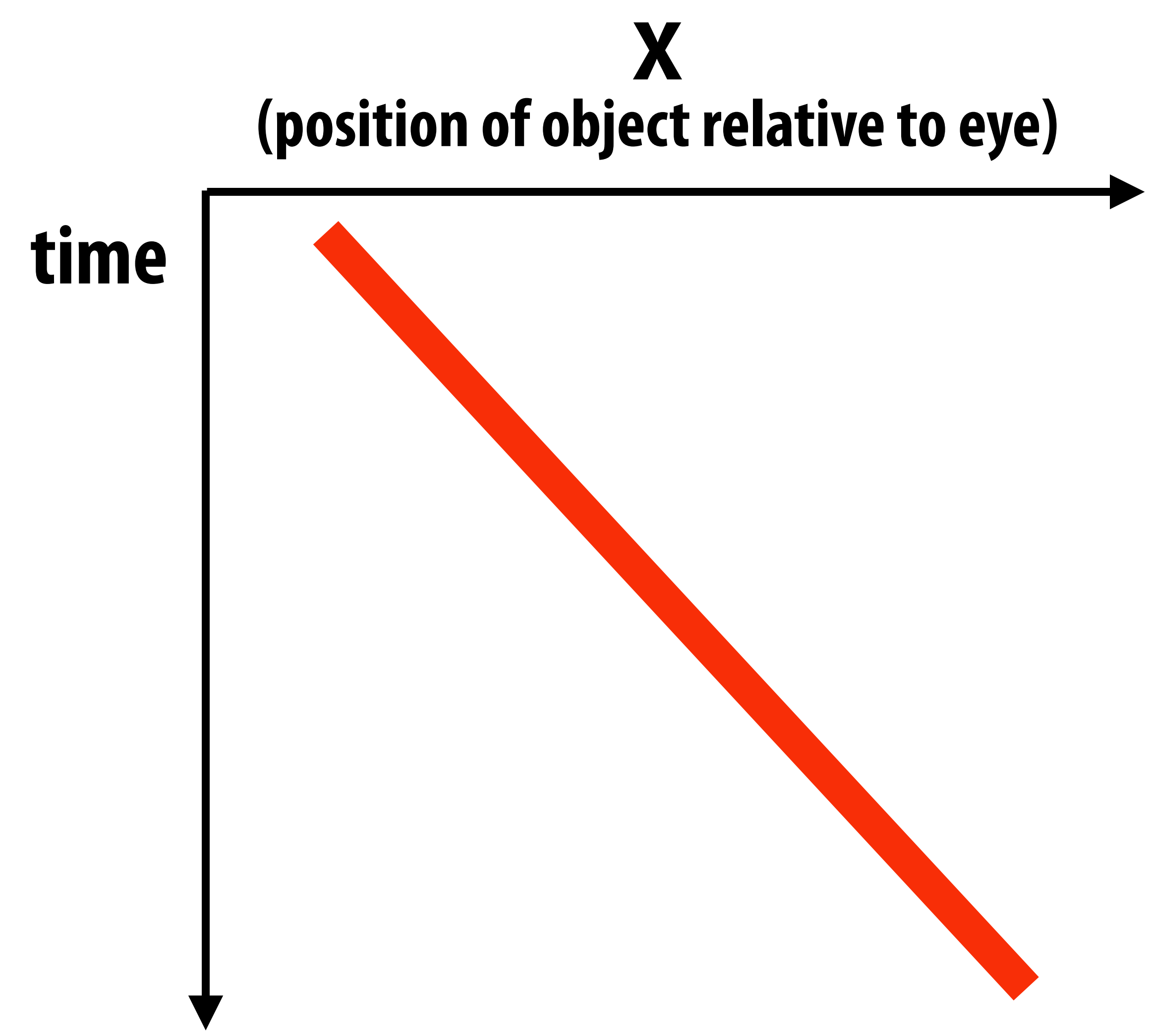
Consider object position relative to eye





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:

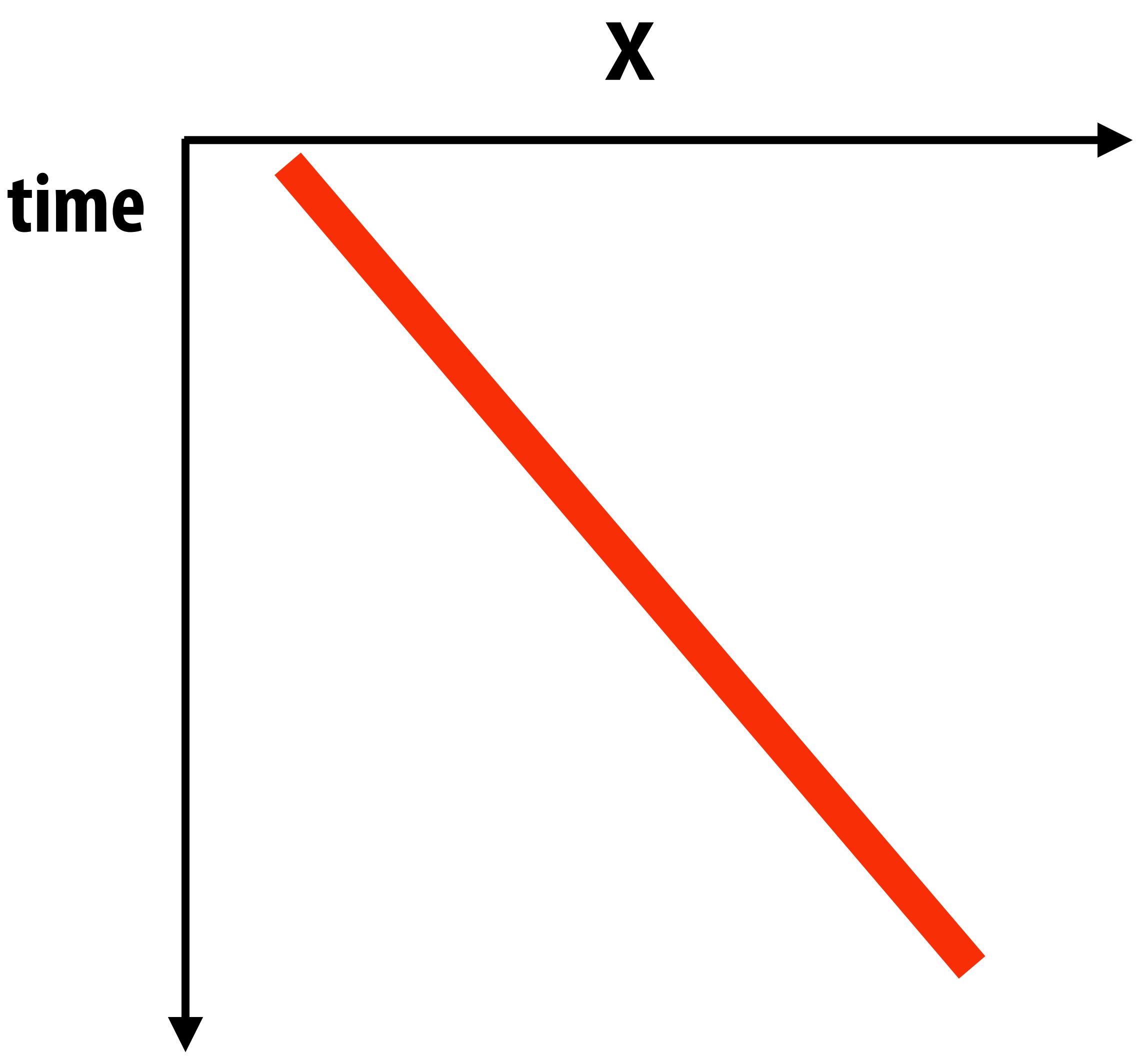
(red object moving from left to right but
eye stationary, i.e., it's focused on a different
stationary point in world)

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

RAPID HEAD MOTION WITH EYES TRACKING AN OBJECT IS A FORM OF CASE 1!!!

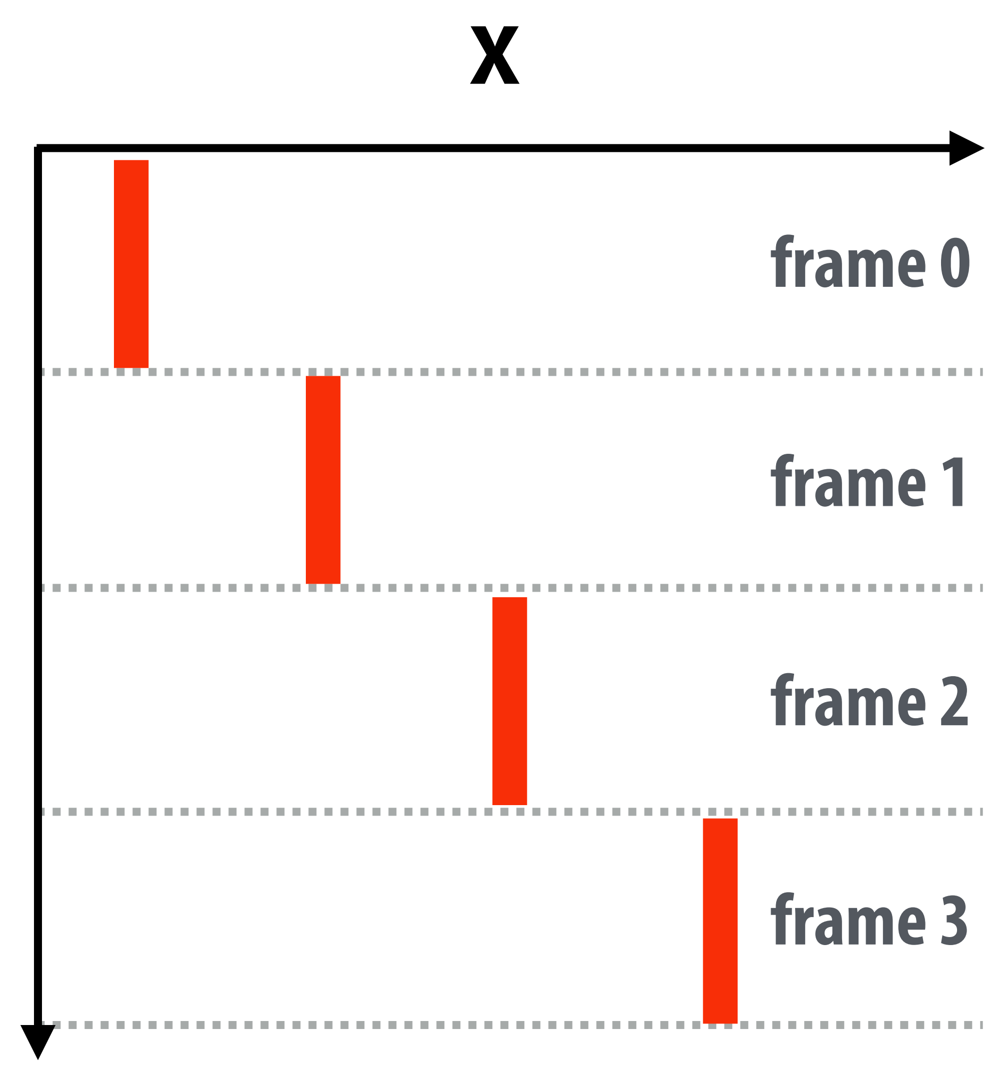
Spacetime diagrams adopted from presentations by Michael Abrash
Eyes designed by SuperAtic LABS from the thenounproject.com

Effect of latency: judder



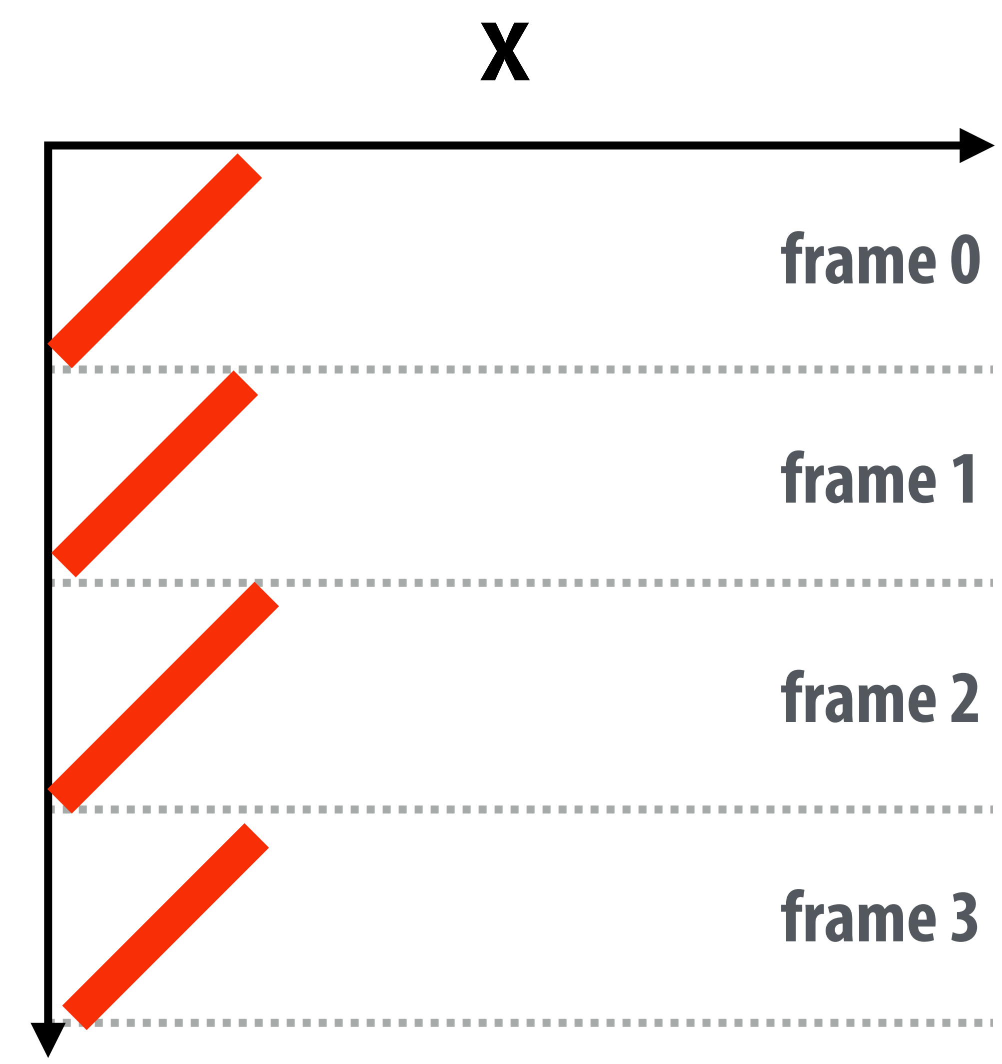
Case 2: object moving from left to right, eye stationary (<u>eye stationary with respect to display</u>)

Continuous representation.



Case 2: Object moving from left to right, eye stationary (<u>eye stationary with respect to display</u>)

Light from display (image is updated each frame)



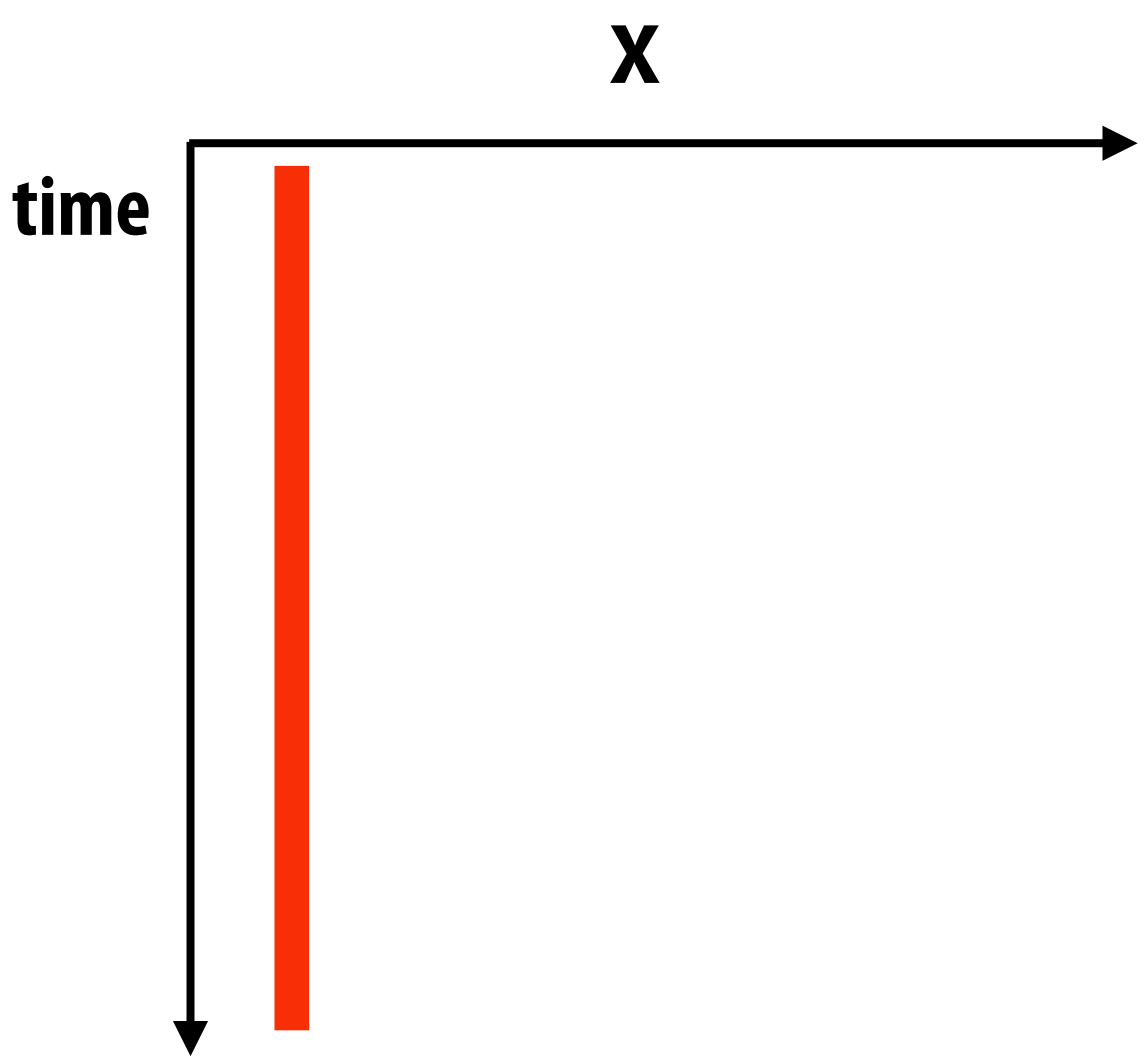
Case 1: object moving from left to right, eye moving continuously to track object (<u>eye moving relative to display!</u>)

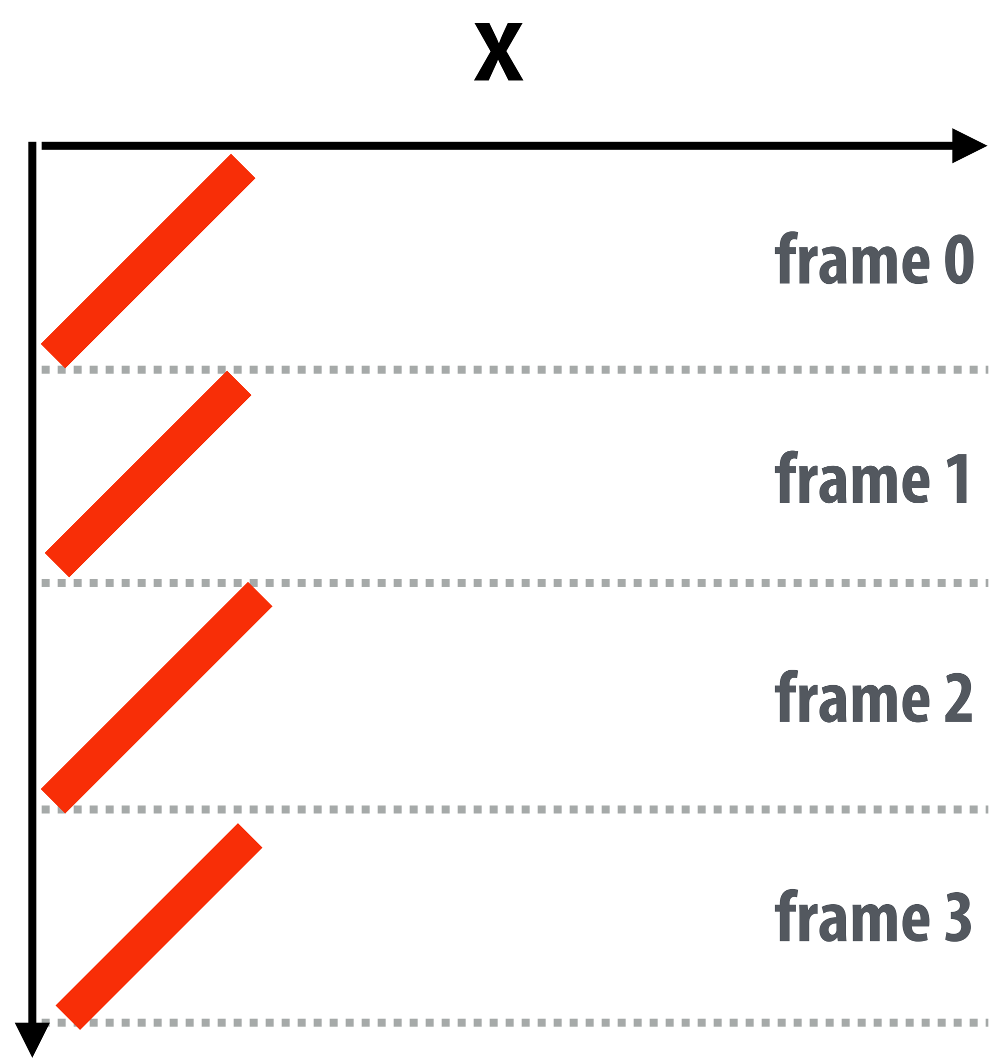
Light from display (image is updated each frame)

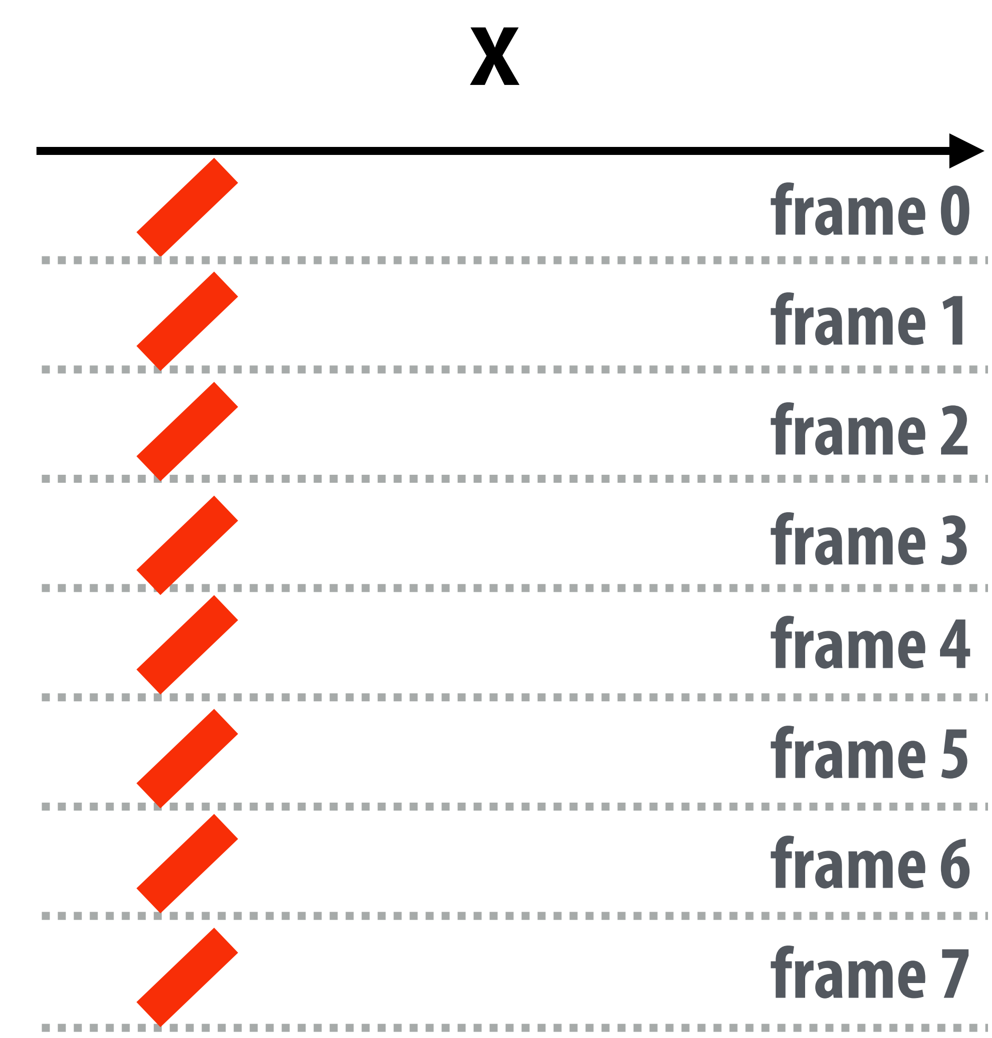
Note: since eye is moving, object's position is relatively constant relative to eye. (as it should be, eye is tracking it)
But due discrete frame rate, object falls behind eye, causing a smearing/strobing effect ("choppy" motion blur)
Recall from earlier slide: 90 degree motion, with 50ms latency results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

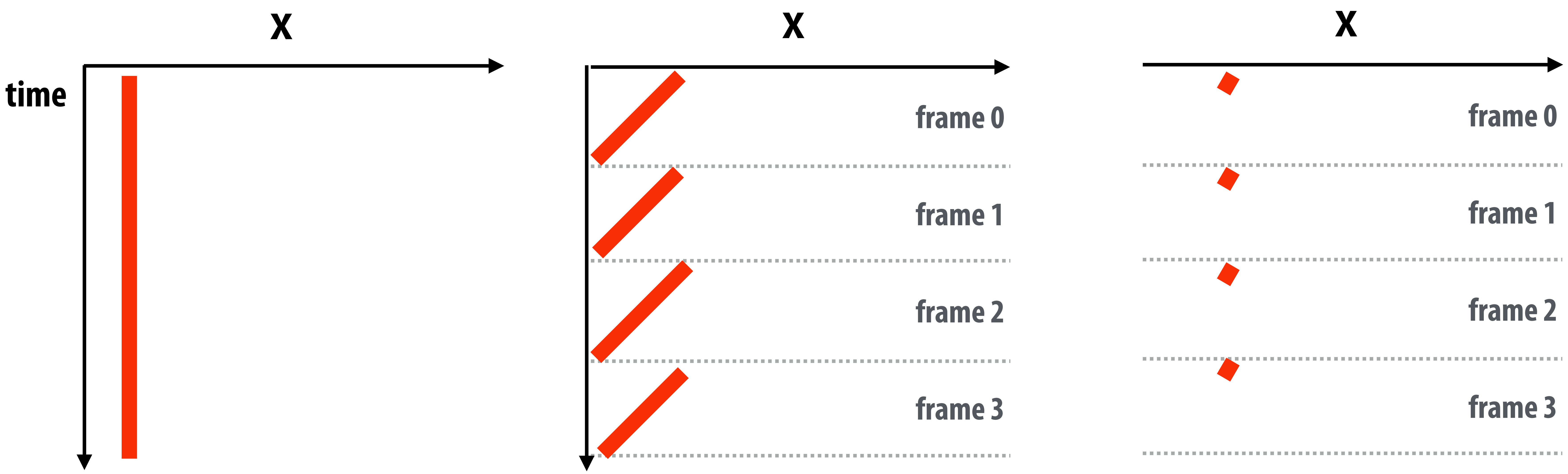
**Light from display
(image is updated each frame)**

**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

Light from full-persistence display

Light from low-persistence display

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

Full-persistence display: pixels emit light for entire frame

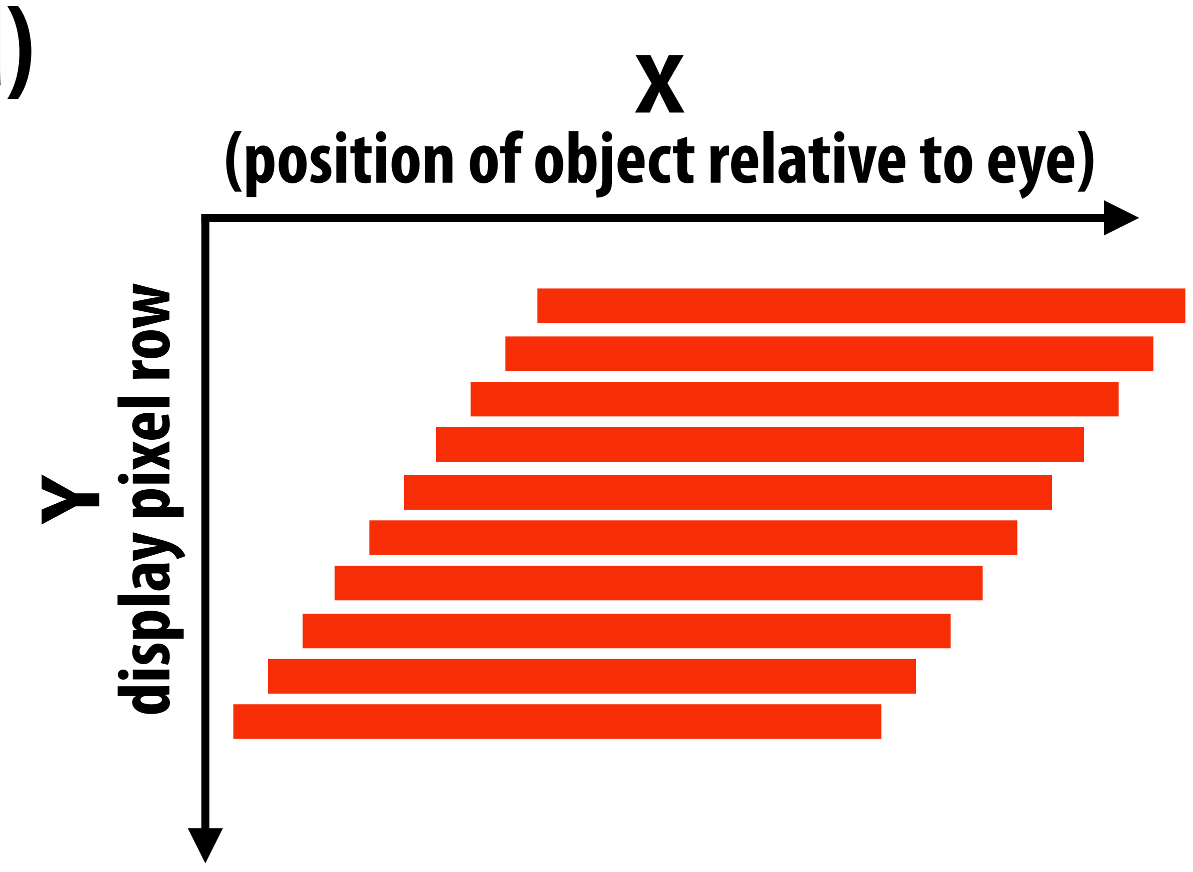
Low-persistence display: pixels emit light for small fraction of frame

Oculus DK2 OLED low-persistence display

- 75 Hz frame rate (~13 ms per frame)
- Pixel persistence = 2-3ms

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 1 photos hit eye at $t_0 + \Delta t + 2r$
- Implication: photos from bottom of display are "more stale" than photos from 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 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 the predicted different time photos will hit eye**
  - Suggests exploration of different rendering engines that are more amenable to fine-grained temporal sampling, e.g., ray caster?

Increasing frame rate using reprojection

- **Goal: maintain as high a frame rate as possible under challenging rendering conditions:**
  - Stereo rendering: both left and right eye views
  - Render to high resolution outputs
  - Render extra pixels due to barrel distortion warp
  - Cost saving "rendering hacks" (bump mapping, billboards, etc.) are less effective

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

Potential future VR system components

Low-latency image processing
for subject tracking

Computation for high-resolution rendering

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

Exceptionally high bandwidth connection between
renderer and display:
4K x 4K per eye at 75-90 fps!

In headset motion/accel
sensors + eye tracker

On headset graphics processor
for sensor processing and
reprojection

Summary

- Virtual reality presents many new challenges for graphics systems
  - Primary challenge = minimize latency of head movement to photons *
    - Requires low latency tracking
      - 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: considering image-based (reprojection) techniques
      - Rendering must compensate for constraints of display system:
        - Optical distortion (geometric, chromatic)
        - Temporal offsets in pixel
  - Not discussed today:
    - Alternative display technologies to flat screens with lenses in front of them: e.g., light field displays

* = "primary" in the context of this class: other very-hard challenges include: tracking full-bodies, determining what applications VR is a preferred medium for, etc.