Immersive Systems III — Slido

Developing Immersive Applications

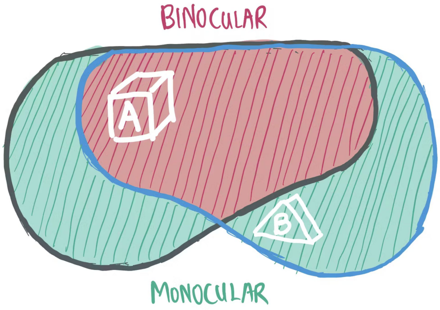
Created by: Chek Tien TAN

While exploring the simulator, you notice nearby objects disappear when the HMD gets too close. What should you adjust in `hmd.ts` to fix this?

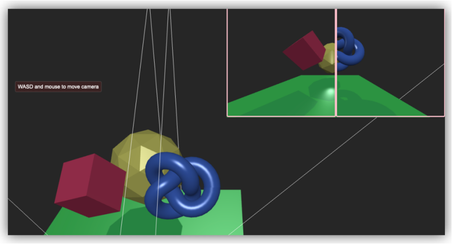
- Increase the IPD value
- Decrease the focal length
- Reduce the near clipping plane distance
- Increase the far clipping plane distance

Which object is easier for me to reach out and grab with my hands in VR?

- A
- B
- Both are equally reachable



A user testing your prototype headset reports, “I cannot sense depth, everything looks like two separate 3D images floating side by side, rather than a cohesive immersive scene.” What adjustments would you make to correct it?



- Increase the IPD setting
- Decrease the IPD setting
- Change the focal length of the lenses
- Reduce the near clipping plane

After toggling both frustum visualizations and viewing from above, you notice the left and right frustums are not symmetric. Why are the frustums asymmetric horizontally?

- Because the barrel distortion shader warps the frustum shape
- Because each eye is offset from the display center by IPD/2, making the nasal side narrower than the temporal side
- Because the left and right displays have different resolutions
- Because Babylon.js uses a left-handed coordinate system

What is the primary trade-off when increasing the field of view (FOV) in an HMD?

- Wider FOV always improves both immersion and comfort equally
- Wider FOV increases immersion but may increase optical distortion at edges and cybersickness risk
- Wider FOV reduces rendering cost because fewer pixels need to be shaded
- Wider FOV has no effect on cybersickness

After reducing eye relief in the simulator and observing the stats panel changes, what is the primary visual effect of reducing eye relief in an HMD?

- Narrower field of view
- Better comfort for glasses wearers
- Wider field of view
- Improved color accuracy

After adjusting the focal length slider in the simulator, which calculated value in the stats panel changes to reflect the distance at which your eyes must accommodate (focus)?

- `aspectRatio`
- `distEye2Img`
- `displayWidth`
- `near`

In current HMDs, the virtual image is at a fixed distance determined by the optics, but objects appear at varying depths. What is this mismatch called?

- Binocular rivalry — the two eyes receive conflicting images
- Vergence-accommodation conflict (VAC) — the eyes converge at one distance but accommodate (focus) at another
- Stereoblindness — the brain cannot fuse the two images
- Motion sickness — visual-vestibular mismatch during movement

In the HMD simulator codebase, which module is primarily responsible for computing the stereo camera setup (left/right eye view and projection matrices)?

- `app.ts` — scene management and camera creation
- `hmd.ts` — HMD parameter management and stereo rendering
- `ui.ts` — slider controls and user interface
- `constants.ts` — layer masks and preset values

Why do HMDs apply barrel distortion correction to the rendered image before displaying it?

- To increase the resolution of the display
- To pre-compensate for the pincushion distortion introduced by the lenses
- To reduce motion-to-photon latency
- To improve color accuracy at the edges

Look carefully at the edges of the left/right PIP viewports in the simulator. Compare straight lines (e.g., the grid or box edges) at the center vs near the edges. What visual effect do you observe?

- Lines appear perfectly straight everywhere
- Lines near the edges bow outward (barrel distortion), and a greyish-pink fallback border is visible beyond the warped region
- Lines near the edges bow inward (pincushion distortion)
- The center of the image is darker than the edges

Why do we need to view the 3D scene through the Google Cardboard viewer to perceive depth? Isn’t just viewing the side-by-side simulated output on screen enough?

- The screen is too small to show depth without magnification
- Without the viewer, both eyes see the same combined image; the Cardboard lenses ensure each eye sees only its own offset view, enabling stereopsis via binocular disparity
- The lenses add special depth information to the image
- Viewing on screen already provides stereopsis; the Cardboard just adds comfort

In the HMD simulator, the scene is rendered to a fixed-resolution render target for each eye. If you increase the display width (thus widening the FOV) without changing the render target resolution, what happens to the perceived image quality?

- Quality improves because more of the scene is visible
- Quality degrades because the same number of pixels are spread over a wider FOV, reducing angular pixel density
- Quality stays the same because the barrel distortion shader compensates
- Quality improves because wider displays always have more pixels

Describe what “impossible” or creative rendering configuration you tried in the HMD simulator and what visual effect you observed.

(Open-ended — type your answer in Slido)

Examples to try:

- Setting focal length below lens-to-display distance (inverted image)
- Extreme IPD values (hyper/hypo stereopsis)
- Focal length very close to lens-to-display distance (extreme magnification)
- Extreme display dimensions (ultra-wide or ultra-tall FOV)