Immersive Systems II
HMD Optics

Developing Immersive Applications

Created by: Chek Tien TAN

Learning Objectives:

- explain the image formation process in typical XR HMDs
- apply thin lens optics principles to understand HMD design
- describe how IPD affects stereo rendering and user comfort
- explain lens distortion and correction in HMDs

Why VR Optics Matters

1. **See It Right** — Accurate optical design ensures realistic virtual worlds
2. **Render Smarter** — Understanding optics helps optimize graphics rendering
3. **Works Across Devices** — Optics knowledge transfers to all HMD platforms
4. **Comfort = Retention** — Proper optics reduces eye strain and motion sickness
5. **Break the Rules** — Know the principles to innovate beyond them

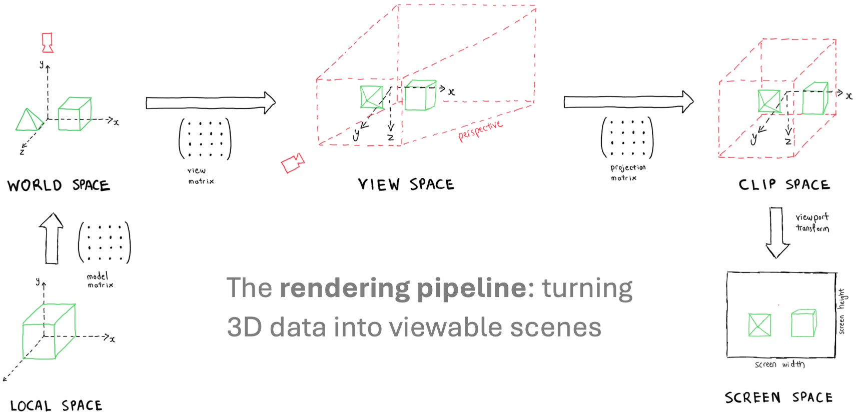
Core Graphics Concepts

The Rendering Pipeline:

- **Application** — Scene setup, game logic
- **Geometry Processing** — Vertex transformations
- **Rasterization** — Converting vectors to pixels
- **Pixel Processing** — Shading, texturing, lighting
- **Output** — Frame buffer display

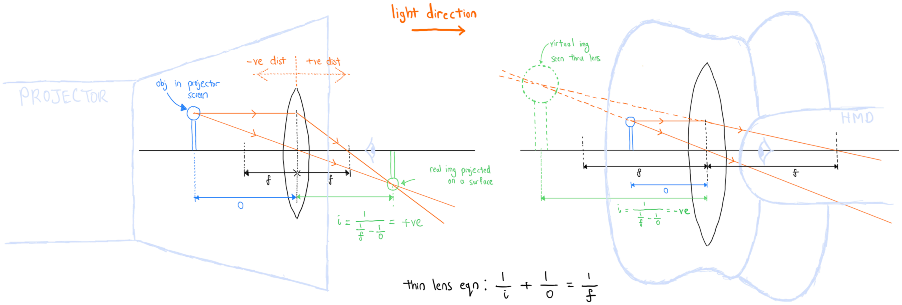
VR requires rendering this pipeline **twice per frame** (once for each eye) at high frame rates (90+ FPS).

Core Graphics Concepts

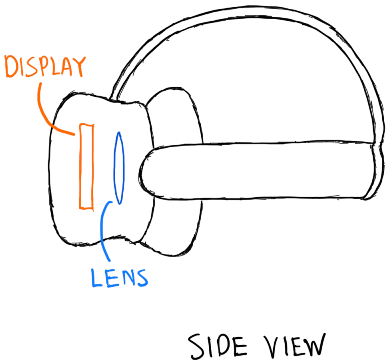


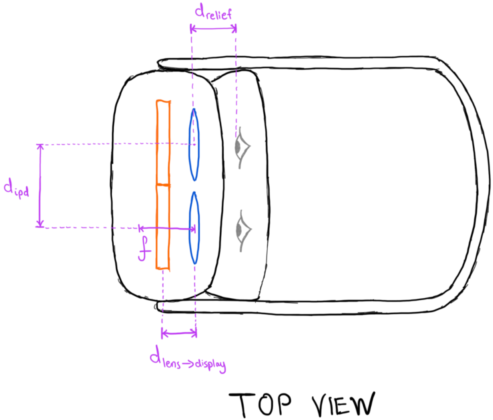
The **rendering pipeline**: turning 3D data into viewable scenes

Core Physics Concepts

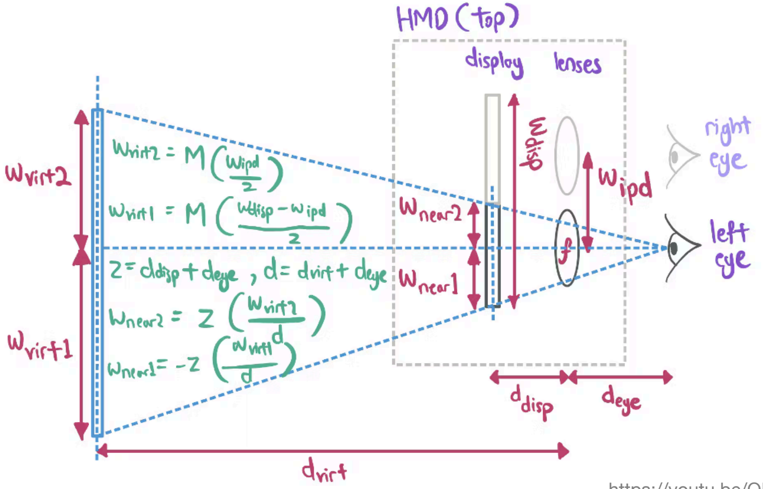


Key HMD Hardware Components





Thin Lens Equations



https://youtu.be/OKD4jrnn4WE?t=406

Vergence-Accommodation Conflict

Natural Vision: Vergence (eye rotation) and accommodation (focusing) are coupled

- Near object: eyes converge inward, lenses thicken
- Far object: eyes parallel, lenses flatten

In VR/AR: Display at fixed distance, but virtual objects at varying depths

- Eyes accommodate to fixed screen distance
- But vergence follows virtual object depth

The Conflict: Eyes must decouple normally linked processes Result: eye strain, fatigue, discomfort

Industry Solutions:

- **Varifocal displays:** Dynamically adjust focal distance to match vergence
- **Light field displays:** Multiple focal planes simultaneously
- **Foveated rendering:** High quality only where user is looking

Key Parameter: IPD

Inter-Pupillary Distance:

- Distance between the centers of your pupils
- Typical range: 58-72mm (average: 63mm)
- **Critical for comfortable stereo viewing**
- Mismatch causes eye strain and depth perception issues
- Most modern HMDs have adjustable IPD

Software implications:

- Affects view matrix for each eye (different camera positions)
- Each eye needs separate projection matrix
- IPD offset = stereo baseline for depth perception

Hint: What happens if IPD is wrong? Think depth perception and eye strain.

Key Parameter: Eye Relief

- Distance from the lens to your eye
- **Affects field of view (FOV)**
- Needs to accommodate glasses wearers
- Typical range: 10-20mm

Trade-off:

- Closer eye relief = wider FOV, more immersive
- Further eye relief = more comfortable, accommodates glasses

Design challenge: Balance immersion vs comfort for diverse users

Hint: Scene looks shrunk or distant? Check your FOV settings.

Key Parameter: Field of View

- Angular extent of the visible world
- Human vision: 210° horizontal, 150° vertical
- VR HMDs: typically 90-110° diagonal
- **Wider FOV = more immersive**, but harder to render
- Depends on lens design, focal length, and eye relief

FOV calculations:

- **Binocular FOV** — overlap region both eyes see
- **Monocular FOV** — total extent including peripheral
- **Nasal FOV** — towards nose (limited by nose bridge)
- **Temporal FOV** — towards temples (wider)

View Frustum in VR

What is a view frustum?

- A truncated pyramid defining the visible 3D space
- Bounded by **near** and **far** clipping planes
- Objects outside the frustum are not rendered (culled)

VR-specific properties:

- **Vertical symmetry** — top and bottom FOV are equal
- **Horizontal asymmetry** — due to IPD offset
- Each eye has a **different frustum** shape
- The frustums overlap in the center (binocular region)

This asymmetry is why projection matrices differ per eye.

Hint: Objects clipped when too close? Think about the near clipping plane.

View Frustum: Key Concepts

Near and far clipping planes:

- **Near plane** — objects closer than this are clipped
- **Far plane** — objects further than this are clipped
- Adjusting near plane affects interaction (e.g., grabbing close objects)

Asymmetry in VR:

- Each eye has **different projection matrix**
- Horizontal asymmetry due to IPD offset
- Vertical symmetry maintained
- This asymmetry is critical for correct stereo rendering

Common issue: Objects clipped when too close? → Adjust near clipping plane in code

Lens Distortion

The Challenge:

- Real lenses introduce distortion (pincushion/barrel)
- Makes straight lines appear curved
- Reduces image quality at edges

The Solution:

- Pre-distort the rendered image
- Lens distortion “undoes” the pre-distortion
- Result: straight lines appear straight to the user

This is why VR content looks distorted on phone screens without cardboard viewers!

Hint: Scene inverted or blurry? Check lens placement and calibration.

Stereo Rendering

Key Concepts:

- Render scene **twice**: once for each eye
- Left and right cameras offset by IPD
- Each eye sees slightly different perspective
- Brain fuses images to perceive depth
- Vergence and accommodation cues

Performance impact:

Doubling the rendering workload requires optimization strategies:

- Foveated rendering (render center sharper than periphery)
- Fixed foveated rendering (hardware-level optimization)
- Multiview rendering (single pass stereo)

Hint: Objects appear flat or hard to judge distance? Check IPD and stereo camera setup.

Projection & View Matrices

Think about it:

Are projection matrices the same for both eyes?
Hint: Consider the view frustum asymmetry we discussed. If each eye has a different frustum shape...

Are view matrices the same for both eyes?
Hint: Remember IPD - the distance between eyes. If cameras are at different positions in space...

Verify your reasoning:

- WebXR provides matrices via `XRView` objects
- Compare `leftView.projectionMatrix` vs `rightView.projectionMatrix`
- Compare `leftView.transform` vs `rightView.transform`

Hands-on: HMD Simulator

Purpose: Bridge hardware optics with software rendering

Try it yourself: `https://hmd.diaversity.org`

What you can explore:

- Adjust IPD — see how stereo images merge/separate
- Change eye relief — observe FOV changes
- Modify focal length — watch image size/distance shift
- View asymmetric frusta visualization
- Experiment with near/far clipping planes

Source code: `https://github.com/singaporetech/hmd`

Course paper: `https://doi.org/10.1145/3757371.3763258`

Summary

Today we covered:

- Why VR optics matters for immersive experiences
- Thin lens optics and image formation process
- Key HMD parameters: IPD, eye relief, FOV
- View frustum properties (asymmetric in VR)
- Lens distortion and correction techniques
- Stereo rendering requirements and performance impact
- Projection and view matrix differences per eye