Virtual Reality Systems

EPFL Immersive Interaction Group

EPFL

IIG

Outline

- Overview of Head-Mounted Display (HMD)
- Tracking System
- Input Devices
- Software Environment
- Project description
- Example of Tutorial
- Project deadlines

Head Mounted Display

Optical solution: Head-Mounted Display (HMD)

➔Providing 3D visual stimuli through head-worn systems

- Displaying the correct image directly to **each eye**
- Sutherland thesis (60s), commercially available in the 80s (Lanier)

Most common 3D format : **Side by side**

- Simultaneous display of left and right eye images at each frame
- Provide full frame rate at the cost of image resolution

Head Mounted Display

Some old models are visible in IIG

The rest of the slides focus on recent models

Head Mounted Display


- Oculus Series (Rift, Rift S, Quest & Go)
- HTC Series (Vive, Vive pro eye, Cosmos)
- Apple vision
- Samsung Gear VR
- Pimax (5k Plus, 8K X and Plus)
- Playstation VR
- Google Cardboard
- Nintendo Labo VR
- Valve Index





Head Mounted Display

Oculus Quest

- All-In-One VR Gaming
- Oculus Insight Tracking
- OLED Display
- Selling point: No PC, No wire, No limits

	Oculus Quest	Oculus Quest 2	Oculus Quest 3
Starting price	$299	$399	$550
Pixels per eye	1440 x 1600	1832 x 1920	2064 x 2208
Screen refresh rate	72 Hz	120 Hz	120 Hz
Field of view	100°	100°	110°

Head Mounted Display

HTC Vive Pro Eye

Valve Index

- PC powered
- motion tracking with base stations

Selling point: Embedded Tobbi Eye tracker

→ Gaze-based + Blink-based interactions

Head Mounted Display

Device comparison

Features	Oculus Quest / Quest2 / Quest 3	HTC Vive Pro (Eye)	Valve Index	PiMax 5k Plus
Minimal requirements	A smartphone for the setup only	GTX 1070 Quadro P5000	GTX 970 AMD RX480	GTX 1070
Display technology	OLED	OLED	LCD	CLPL
Remote connection	Limited	DisplayPort 1.2+ USB 3.0	DisplayPort 1.2+ USB 3.0	USB 2.0/3.0 + DP1.4
HMD sensors	IMU, Gyroscope, Cameras	IMU, Gyroscope, (eye tracking -> IPD)	IMU, Gyroscope	IMU, Gyroscope
Controllers inputs	Buttons | Hand tracking	buttons| eye tracking	Capacitive touch / Force sensors	
Field of View	~ 100 / 110 / 110 degrees	~ 110 degrees	~ 130 degrees	~ 200 degrees
Resolution (per eye)	1440 x 1600 px / 1832x1920 px / 2064 x 2208 px	1440 x 1600 px	1440 x 1600 px	2560 x 1440 px
Refresh Rate	72 Hz / 120 Hz / 120 Hz	90 Hz	90 / 120 / 144 Hz	120 Hz
Price (AVG)	< 600 CHF	CHF 1700	CHF 1100	CHF 810

Tracking system

Camera based

- Marker based active tracking
- Marker based passive tracking
- Markerless tracking

Pros

- Absolute position without drift over time
- relatively accurate devices

Cons

- Occlusions

Camera free

- Mechanical capture (exoskeleton)
- IMU (accelerometers, gyroscope)
- Magnetic sensors
- Deformable gauges (mostly used in gloves)

Pros

- No occlusions

Cons

- Lower accuracy (all)
- distortion induced by metallic objects (magnetic)
- Drifts (IMU)

Tracking system

Lighthouse / Base station

- Active tracking marker based
- Rotating laser @6000rpm
- Range of 7m per base station
- FoV : 160° x 115°
- 4 Base stations can cover up to 10 x 10 m surface
- The device scan the environment to identify without error the ID of each device

HTC Vive Tracker

Tracking system

Vicon Shogun

- Passive marker based solution
- High refresh rate
- High accuracy
- Unable to identify markers without context
- Expensive system
- Targets a professional market

Performer equipped with passive suits for motion capture using Vicon Shogun

Tracking system

Oculus Quest Tracking

- **Inside-out Computer vision tracking**
- Use computer vision with wide angle camera based sensors to locate the headset in space
- Doesn’t requires external devices
- These cameras also provides a **markerless finger tracking**

Input Devices

Oculus Touch

Each controller contains

- One joystick
- Two press buttons
- *Two trigger buttons*
- One meta button
- Infrared tracking
- IMU and Gyroscope
- Vibrators

Input Devices

Vive Controller

Each controller contains

- A trigger
- Two meta buttons
- *A tactile button pad*
- Two lateral buttons
- IMU and Gyroscope
- Infrared tracking
- Vibrators

Input Devices

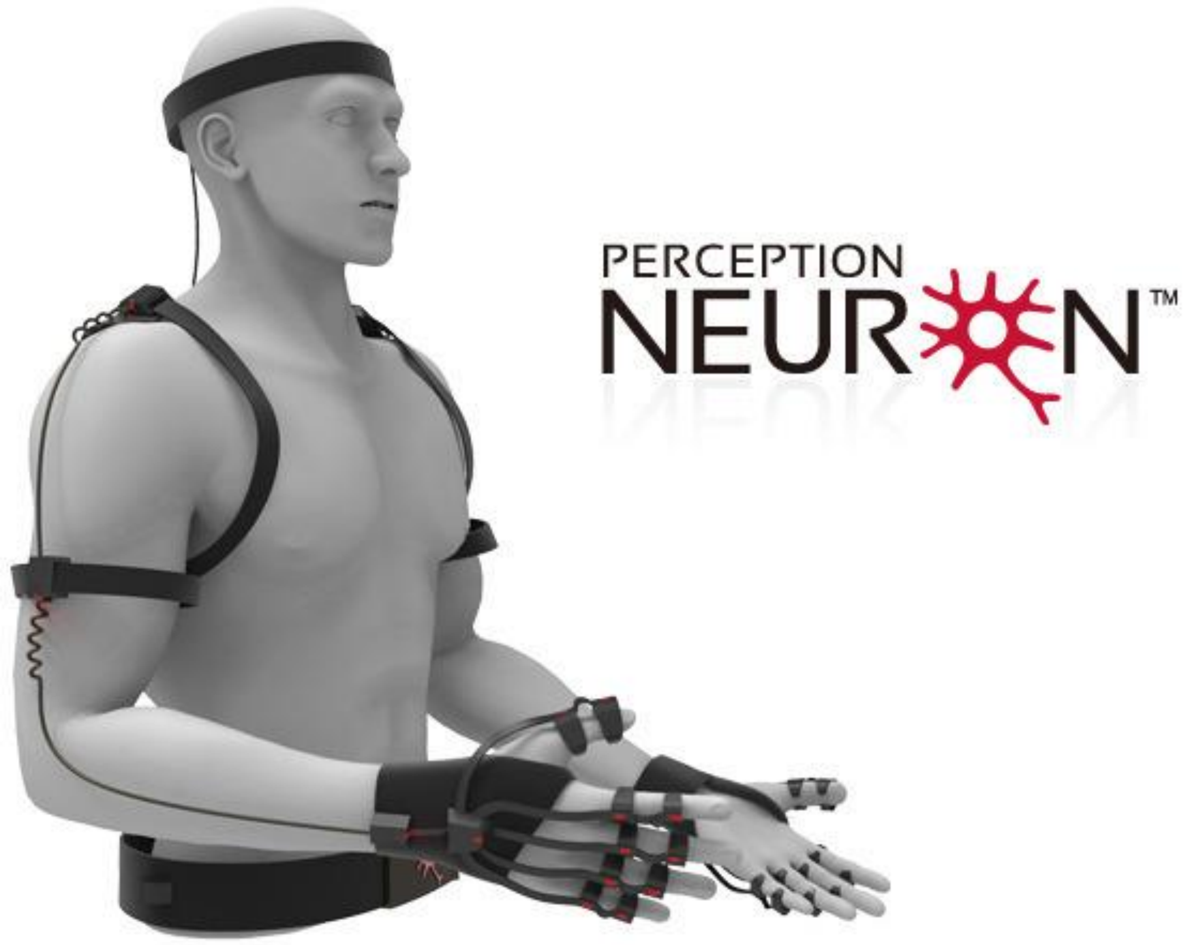
Knuckles

Each controller contains

- One joystick
- Two press buttons
- One trigger
- One meta button
- *Finger tracking through proximity sensors*

Input Devices

Miscellaneous Inputs



Manus VR

PlayStation Controllers

Windows Mixed Reality

Input Devices

Miscellaneous Devices (+/- non finished prototypes)

Dexmo Glove

Infinadeck's Bi-directional treadmill

Software Environment : Game Engine / Editor

- Handles the core of the Game such as
  - Frames
  - Rendering
  - Sound
  - Collisions
  - Physics
  - Etc.

- Provides a framework for developers

Software Environment : Game Engine / Editor

Common Games Engines

Proprietary		Open Source
UNREAL ENGINE	unity	GODOT Game engine
lumberyard	CRYENGINE®	PANDA3D

Software Environment : Game Engine / Editor

Unity 3D

- Widely used (**many forums** available with tips)
- Licensed software (**free** for education / personal use)
- **Multi-target support** (Linux, **Android**, Windows, Mac, PS4, Switch, ...)
- Scripting in **C#** (or JS)
- Perfect integration with **Visual Studio**
- Many resources through the **asset store**
- Technology we use within the IIG

Project description: evaluation criteria

Evaluating Interactions

- **Basic Functionality**: Does it achieve the basic intended action? how well is it achieved? Students were careful to make the interaction stable

- **Quality / Usability** : Is the interaction as intuitive as possible within the context of the game? Is the way the interaction is triggered coherent with the other interactions implemented?

- **Feedback** : Is the interaction visually pleasing, is the audio feedback appropriate ? Students used passive/rumbling haptics as a mechanism to improve the interaction and its usability

Evaluating Locomotion

Same 3 criteria

+ Motion Sickness Inducing criteria: Is the motion highly susceptible to motion sickness ? Students were careful to consider the theoretical concepts relevant to motion sickness induction and attempted to develop something around it

Project description: evaluation criteria

Evaluating the game

- **Creativity and Quality of the Game:**
  - How original is the game? e.g. is it just a copy of an existing game?
  - How well the interactions are integrated into the game?
  - Is the game fun to play?
  - Is the soundscape appropriate ?

- **User Friendly:**
  - How friendly is the game to new players.
  - Is play intuitive?
  - Are the controls easy to use (i.e. not frustrating)?

- **Tutorial Quality:**
  - Can the game teach us how to play?
  - How well does the game explain its concepts to the player and how self-sufficient it is without any external help?

- **Visual Fidelity/Quality:**
  - How does the game look, and does it have a thematic consistency.
  - Did students take some degree of care about the visuals of the game?


➔ Feedback is the key for the project

➔ Do not underestimate the amount of work needed

- Organisation for a group-project (oral exams: all group members must know how the whole project works, not only what they personally developped)
- New software with a specific way of handling inputs in a frame-by-frame way
- New language to learn (C#)
- Designing well-done interactions takes time
- Debugging can be very time-consuming
- Building a nice 3D environment takes time, even with premade assets - The soundscape and the light matter for the game atmosphere
- Building a nice scenario takes time
- Need for user feedbacks to iterate

You can start installing the project environment with the Hands-On HO1-a