System Control (Part 2)

Reading: Understanding Virtual Reality (2nd Edition), Ch. 8, pp. 702-723
Diegesis refers to elements that exist within the “world” of a film, game, or other narrative.

These elements may or may not be visualized in 3D space.
Non-diegetic elements are rendered outside the virtual world. They are only visible or audible to the players in the real world.

Examples
health bars, level indicators, quest objectives, maps, etc.
**Spatial** elements are rendered in 3D space, but do not exist in the virtual world. They are only visible or audible to the players in the real world.

**Example**
parabolic arc for teleportation
Meta elements exist in the virtual world, but aren’t visualized spatially.

Example
floating text to show messages or internal thoughts

Sherlock (BBC)
Diegetic elements exist in both the narrative and 3D space of the virtual world. They can potentially be seen or heard by virtual characters.

An ideal VR user interface would be diegetic and easily usable.
Your Diegetic UI Examples
Horizon Zero Dawn
Submitted by Sam Adeniyi
Call of Duty: Zombies
Submitted by Frank Bender
Far Cry 2
Submitted by Debobani Biswas
Halo
Submitted by Michael Boschwitz
Halo
Submitted by Matthew Zent
Legend of Zelda: Breath of the Wild
Submitted by Temay Broadway
Firewatch
Submitted by Cole Davidson
Star Citizen
Submitted by Ryan Delmain
No Man's Sky
Submitted by Jacob Diethert
No Man's Sky VR

Submitted by Isaac Kasahara
Forest
Submitted by Runqiu Guo
Superhot VR
Submitted by Christian Halvorson

Video
Fallout 76
Submitted by Jon-Michael Hoang
Dance Central
Submitted by Jane Huynh
The Division
Submitted by Jeffrey Jia
Subnautica
Submitted by Ishan Joshi, Carol Mikhael, and Hao Wu
Phasmophobia
Submitted by Nick Karlovich and Xizhi Ren

Video
FlyInside Flight Simulator
Submitted by Haining Lan
Apex Legends
Submitted by Jianhua Li
Human Fall Flat
Submitted by Weihao Li
Uncharted 3
Submitted by Nate Meshesha

Video
Echo Arena
Submitted by Zachary Mundt
Metroid Prime
Submitted by Jacob Nelson

Video
Five Nights at Freddy's
Submitted by Kerri Newcomer
Titanfall
Submitted by David Nguyen

Video
MLB The Show 22
Submitted by Jimmy Nguyen
ARK: Survival Evolved
Submitted by Iuliia Pan
Keep Talking and Nobody Explodes
Submitted by Jarod Pivovar and Tobias Moszer

Video
Elite Dangerous
Submitted by Neha Ranjan
Star Wars Jedi: Fallen Order
Submitted by David Sullivan

Video
Half Life: Alyx
Submitted by Haoyu Tan
Metro Exodus
Submitted by Jackson Thellin
Metro Exodus
Submitted by Wenjie Zhang
System Control Interfaces

- Physical Controllers
- Graphical Menus
- Voice Commands
- Gestural Commands
- Virtual Tools
- Multimodal Techniques
Voice Commands

**Speech recognition** techniques are typically used for issuing single commands.

**Spoken dialogue** techniques are focused on promoting discourse between the user and system.
Voice Commands
Design and Implementation

Define the tasks that can performed with a voice interface.
Highly complex applications may need conversational interfaces.
Example: Windows Mixed Reality

Video
Example: Virtual Humans

Video
Voice Commands
Design and Implementation

Remember that voice interfaces are invisible to the user!
You may need to provide users with instruction prompts or reminders.

Image Credit: https://docs.microsoft.com/en-us/windows/mixed-reality/out-of-scope/voice-design
Error rates will increase for applications involving communication between multiple participants. A button to initialize the speech input may be needed.

Image Credit: https://www.howtogeek.com/662035/how-to-enable-push-to-talk-on-steam/
Voice Commands
Practical Application

Speech input is hands-free and natural. This can be useful for symbolic input.

Users may first need to learn the voice commands. The UI design should support discoverability.

Smartphones/tablets have built-in voice recognition. These devices can be leveraged for hybrid interfaces or handheld AR.
System Control Interfaces

- Physical Controllers
- Graphical Menus
- Voice Commands
- Gestural Commands
- Virtual Tools
- Multimodal Techniques
Gestural Commands

Gesture interfaces are considered an integral part of:
- *Perceptual user interfaces* (Turk and Robertson 2000)
- *Natural user interfaces* (Wigdor and Wixon 2011)

Designing a **robust** and easy-to-learn gestural interaction system is challenging.
Gestural Commands

**Figure 9.11** Examples of postures using a Data Glove. (Photograph courtesy of Joseph J. LaViola Jr.)

**Posture**
a static configuration of the hand

**Figure 9.12** Mimic gesture. (Schkolne et al. 2001; © 2001 ACM; reprinted by permission)

**Gesture**
a movement of the hand, sometimes while in a particular posture
Gestural Commands

- **Mimic gestures**
  Gestures that are not connected to speech but are directly used to describe a concept

- **Symbolic gestures**
  Gestures as used in daily life to express things like insults or praise (e.g., “thumbs up”)

- **Sweeping**
  Gestures coupled to the use of marking-menu techniques

- **Sign language**
  Specified set of postures and gestures in communicating with hearing-impaired people

- **Speech-connected hand gestures**
  Spontaneous movements or language-like gestures performed during speech

- **Surface-based gestures**
  Gestures made on 2D multitouch surfaces
Gestural Commands

Design and Implementation Issues

Gestural interaction depends heavily on the available input devices.
Example: VR + Kinect
Gestural Commands
Design and Implementation Issues

Gesture-based control shares many characteristics of speech input.
Gestures should have clear delimiters that indicate initialization and termination.

Users may need to discover the actual gesture or posture language.
Include tutorials or example visualizations to help users learn!

Gesture recognition is still not always reliable.
Low gesture recognition accuracy may require larger menu items.
Gestural Commands

Practical Application

Figure 9.14 A user performing a 3D climbing gesture in a video game application. (Image courtesy of Joseph LaViole.)

Figure 9.15 A user controlling an unmanned aerial vehicle (UAV) with 3D gestures. (Image courtesy of Joseph LaViole.)

Gestural interfaces are becoming more common in video games and other applications such as telerobotics and health care.
System Control Interfaces

- Physical Controllers
- Graphical Menus
- Voice Commands
- Gestural Commands
- Virtual Tools
- Multimodal Techniques
**Tools** leverage correspondences with real world objects for direct interaction. In many applications, real-world devices can improve usability.

**Physical tools** are real world objects (props) that represent a real world tool.

*Figure 9.19* Visualization artifacts—physical tools for mediating interaction with 3D UIs. (Image courtesy of Brygg Ullmer and Stefan Zachow, Zuse Institute Berlin)
Example: CavePainting

Video
Tools

Virtual tools have no physical instantiation.

Tangibles can also be abstract shapes to which functions are connected.
Example: Reactable

Video
Tools
Design and Implementation

The tool's form communicates function to the user.
Carefully consider the physical form when developing props!

Compliance between real/virtual is important.
However, they do need to match exactly.
Physical tools are very specific devices. In many cases, they perform only one function.

In complex applications, tools can still be useful, but may not apply to all tasks. There is a tradeoff between the tool specificity and switching.

Public installations can greatly benefit from tools. Physical props can be immediately usable and also afford eyes-off operation.
System Control Interfaces

- Physical Controllers
- Graphical Menus
- Voice Commands
- Gestural Commands
- Virtual Tools
- Multimodal Techniques
Multimodal techniques connect multiple input streams.
This can be effective for system control in certain situations.

Figure 9.21. A car wheel is selected, rotated, and moved to its correct position using voice and gestures. (Photographs courtesy of Marc Erich Latoschik, AI & VR Lab, University of Bielefeld; Latoschik 2001)
Multimodal Techniques

Possible Uses

Decoupling
Using a different input channel that differs from the main input channel can decrease user’s cognitive load.

Error reduction and correction
The use of multiple input channels can be very effective when the input is ambiguous or noisy.

Flexibility and complementary behavior
Control can be more flexible when users can use multiple input channels to perform the same task.
Multimodal Techniques

Microsoft designed the HoloLens for multimodal input.

credit: https://www.microsoft.com/en-us/hololens/developers
Multimodal Techniques
Design Principles

Combination of modalities is task-dependent. Impact on cognitive resources depends upon the particular application.

Switching modalities may affect the interaction flow.
Multimodal Techniques
Practical Application

Complex applications can benefit from the complementary nature of multimodal techniques. Allows for more flexible input or potentially reducing errors.

Multimodal techniques are applicable to scenarios that mimic natural behavior.
System Control Design Guidelines

Logically structure the functions in an application.

Avoid disturbing the interaction flow.

Prevent unnecessary mode and context switching.

Consider using multimodal input to reduce errors.

Design the UI for **discoverability**.
Participation Exercise

Identify a system control task in your final project.

Describe how you could implement this task using design principles and best practices that we learned in class.

Note that this is hypothetical. For the purposes of this exercise, you can consider using devices or modalities beyond what is possible on the Quest.
Image Credits

- 3D User Interfaces: Theory and Practice (2nd Edition)
- Understanding Virtual Reality: Interface, Application, and Design (2nd Edition)