3D User Interfaces

C. Andújar, N. Pelechano
UPC 2020

1

Contents

- Introduction
- Selection and Manipulation
- Navigation
- Application control

2

Application control

3

Introduction to App Control

Informal definition:

- Access to the application's functionalities (eg Open a file)
- Desktop systems make use of a variety of widgets and function keys.

Types in VR [Bowman04]:

- Request the application to perform a particular action (eg Open file)
- Change the mode of interaction (e.g. brush tool → eraser tool)
- Change the application state (e.g. modify a parameter)

5

Introduction to App Control

Relevance in 3D UI:

- Much of the "real work" in an application consists of *selection*, *manipulation* and *navigation* tasks BUT…
- ... application control is critical because is the link that allows the user to control the interaction flow between above tasks.

→Selection, Manipulation and Navigation tasks have to be combined with many small app control tasks.

6

Introduction to App Control

Are traditional 2D GUI elements appropriate for VR?

- Desktop-based VR → Yes
- Immersive systems → Depends on many factors:
  - 6-DOF input vs. 2-DOF input
  - Different input devices (e.g. lack of physical support)
  - Different output devices (e.g. stereoscopic display)

7

Human Factors to consider

- **Perceptual issues:**
  - Visibility, like occlusion and legibility
  - Focus switching
  - Choice of feedback modalities
- **Cognitive issues:**
  - Functional breadth and depth of the system, structuring of tasks to lower cognitive hurdles

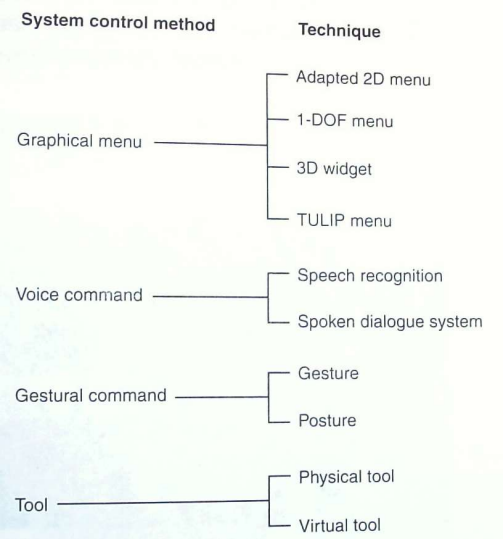
8

Factors that influence effectiveness

- **Human factors and ergonomic issues:**
  - Control placement, and the pose, grip, and motion types a particular device
  - Shape, size, and location of controls can highly affect system control performance
  - visual representation and labeling, method of selection.
- **Input device**
  - Example: pen-and-tablet interface. Constraints provided by physical surfaces increase efficiency.
- **Application's complexity**
  - Techniques for accessing 5 commands can be unusable for 100 commands.

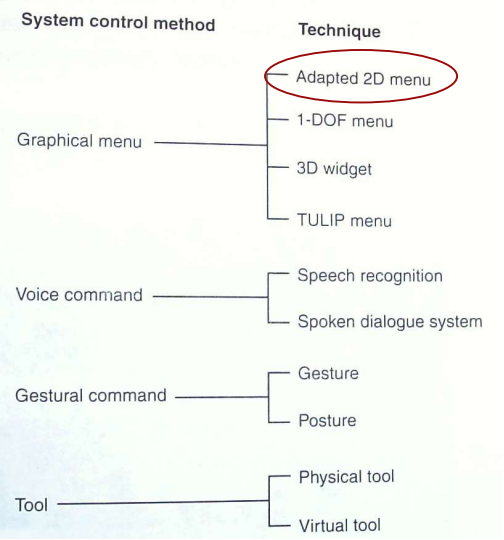
9

Classification [Bowman04]



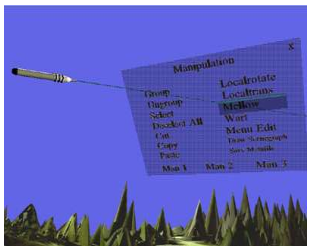
10

Adapted 2D menus



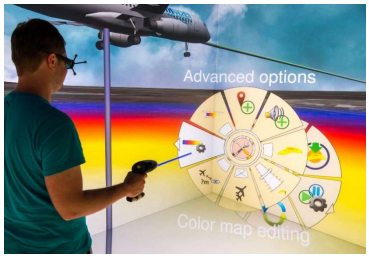
11

Adapted 2D menus



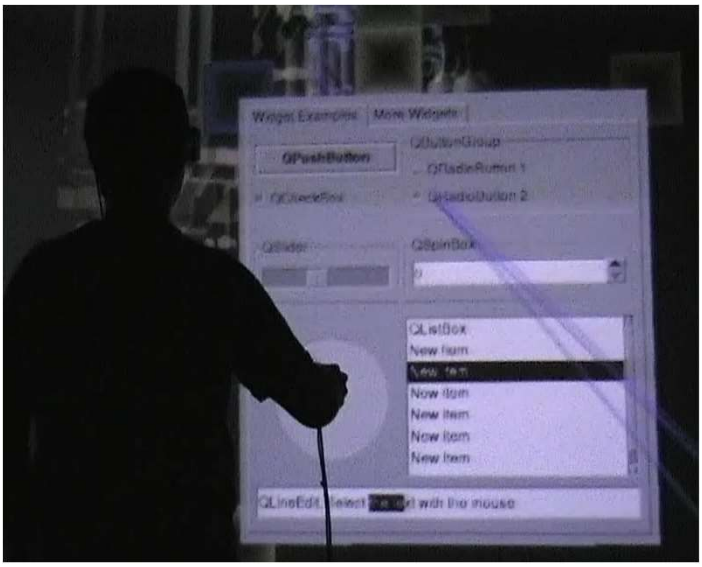
Classic 2D menus displayed on a 3D world.

- Different versions (pull-down, pop-ups, floating)
- Often are semitransparent to reduce occlusion.
- A technique is to attach the menu to a tracked physical surface.
- Users instantly recognize these elements as menus and know how to use them.
- Can occlude the environment.
- Users may have trouble finding the menu.
- ~~Selection of menu items can be difficult using 3D selection techniques.~~



12

Video: adapted 2D menus



13

Video: adapted 2D menus

14

Video: adapted 2D menus

15

Video: adapted 2D menus

16

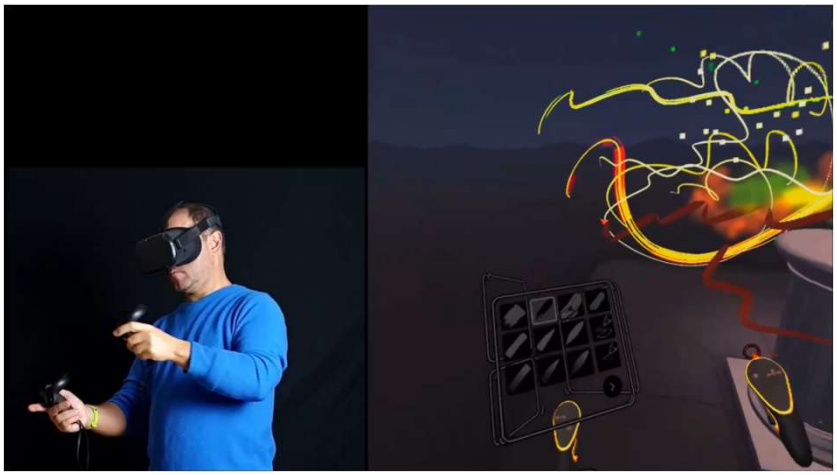
Adapted 2D menus

Hand-held windows

17

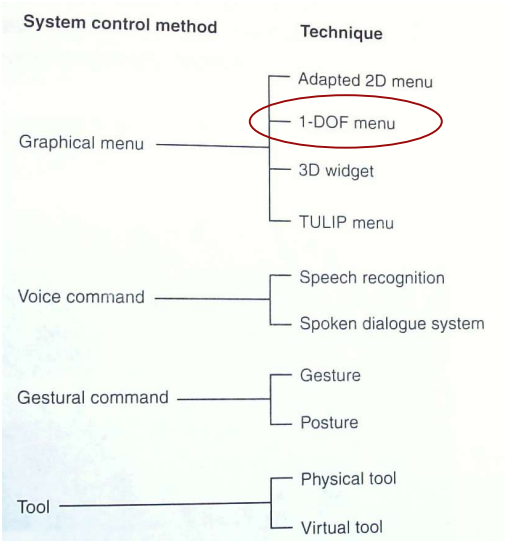
Adapted 2D menus

Hand-held windows



18

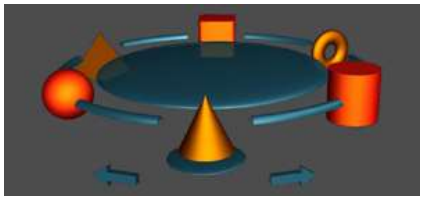
1-DOF menus



19

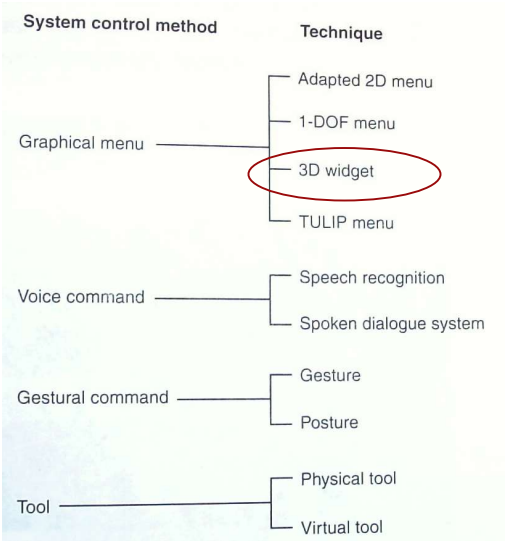
1-DOF menus

- Menus using only 1-DOF
- Example: ring menus [Liang and Ring 94]
  - Items arranged in a circle around the hand.
  - Hand rotation causes all the items to rotate.
  - The selected item is the one within a selection basket.
- Effective for few items



20

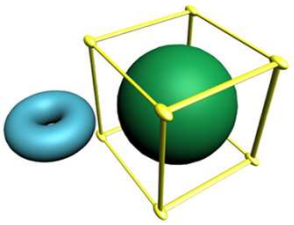
3D widgets

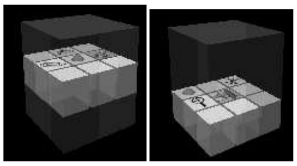


21

3D widgets

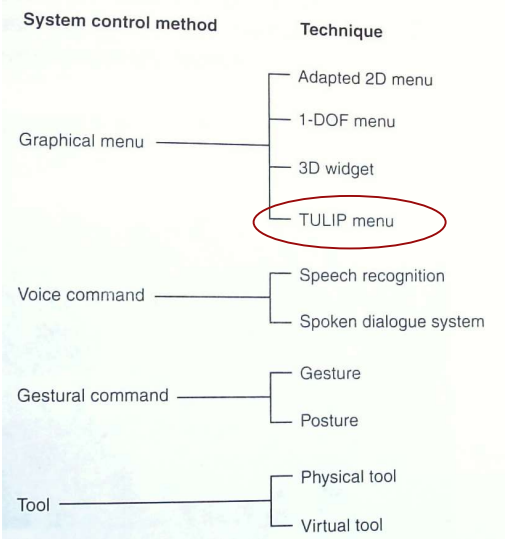
- Colocated
  - Appear close to the selected object
  - Typically used for changing geometric parameters
  - Combine mode selection and manipulation in a single step.
- Non-colocated
  - Not associated to a particular object.





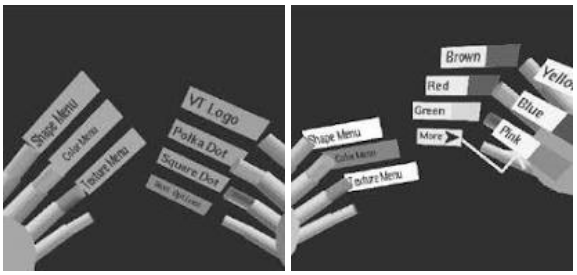
22

3D widgets



23

Tulip menu / Fingertip menu



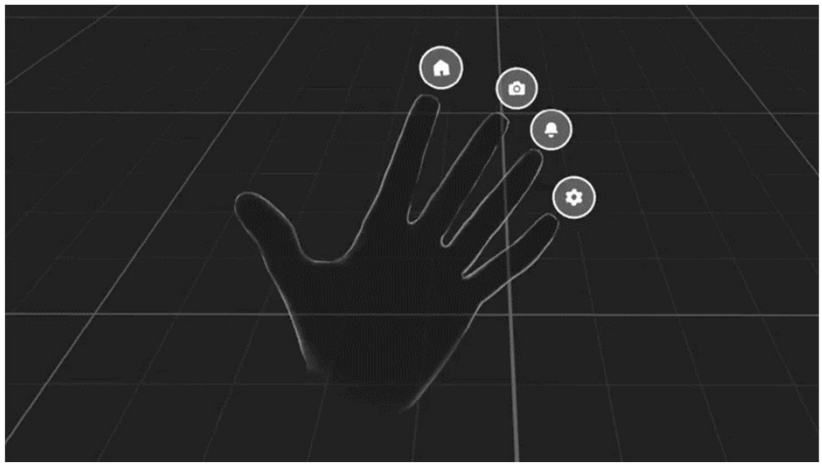
The first level menu items are shown on the left hand. The second level items are shown on the right hand and they can be scrolled together.

To select the item, the user plucks the item from the finger.

Bowman, Doug A., and Chadwick A. Wingrave. "Design and evaluation of menu systems for immersive virtual environments." *Proceedings IEEE Virtual Reality 2001*. IEEE, 2001.

24

Tulip menu / Fingertip menu



25

Tulip menu / Fingertip menu

- Accuracy problems / UX
- Keep the number of buttons small
- Use hand menu for quick actions
- Combine with 2D GUIS that pop up
- Button/panel angle should be correctly oriented towards the head

26

Graphical Menus

Design and Implementation Issues

- Placement
  - The placement of the menu influences the user's ability to access the menu and the amount of occlusion of the environment
  - Hybrid systems combining 2D and 3D interaction can be good choice
  - Non-collocated menus can cause focus switching
  - Occlusion of menu over graphical content can be big issue

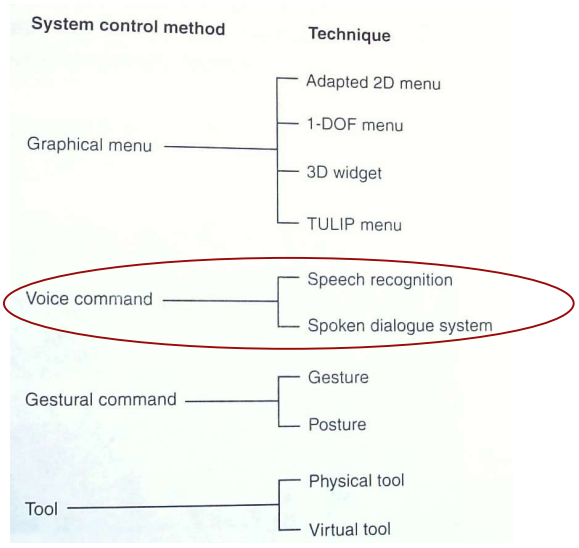
27

Menus on a separate device

- Access a 2D GUI displayed on a separate device (PDA, tablet PC).
- Access to complex 2D GUIs using classic interaction techniques.
- Only suitable for non-immersive displays (not for HMD)
- Cumbersome

28

Voice recognition



29

Voice recognition

Pros:

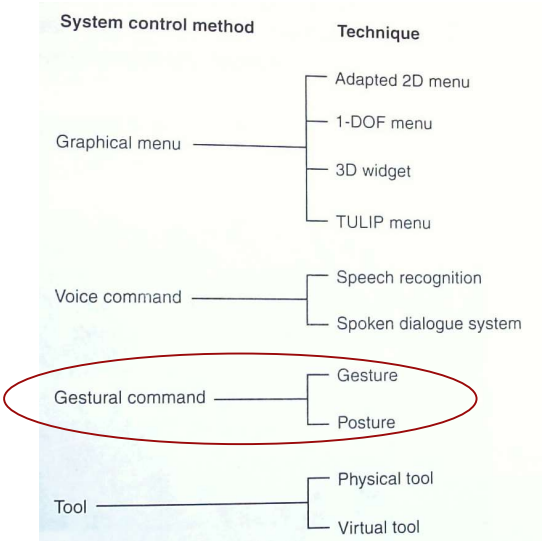
- Hands-free
- Natural

Cons:

- High error rate.
- Can be disturbing to other users.
- User has to learn voice commands.
- Suitable for simple commands (not for adjusting parameters).

31

Gestural recognition



32

Gestural Commands

- Gesture interfaces are often thought of as an integral part of *perceptual user interfaces* (Turk and Robertson 2000) or *natural user interfaces* (Wigdor and Wixon 2011)
- Designing a truly well performing and easy-to-learn system is a challenging task
- Gestural commands can be classified as either **postures** or **gestures**
  - Posture: a static configuration of the hand
  - Gesture: a movement of the hand, perhaps while it is in a certain posture

34

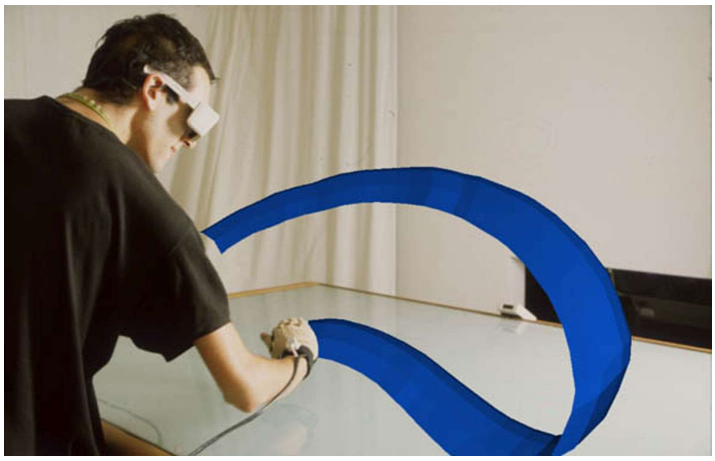
Gestural Commands

Techniques

- In everyday life, we use many different types of gestures that may be combined
- We identify the following gesture categories:
  - Mimic gestures
  - Symbolic gestures
  - Sweeping
  - Sign language
  - Speech-connected hand gestures
  - Surface-based gestures
  - Whole-body interaction

35

Gestural Commands



Mimic gesture.
(Schkolne et al. 2001; © 2001 ACM.
Reprinted with permission from ACM)

36

Gestural Commands

TOUCHEO—combining 2D and 3D interfaces.
(© Inria / Photo H. Raguet).

37

Gestural Commands

Design and Implementation Issues

- Gestural interaction depends heavily on input device
- Gesture recognition is still not always reliable
- When a menu is accessed via a gestural interface, the lower accuracy of gestures may lead to the need for larger menu items
- Gesture-based system control shares many of the characteristics of speech input discussed in the previous section
  - Combines initialization, selection, and issuing of the command
  - Gestures should be designed to have clear *delimiters* that indicate the initialization and termination of the gesture
- Users may need to discover the actual gesture or posture language

Gestural Commands

Practical Application

- Entertainment and video games are just one example of an application domain where 3D gestural interfaces are becoming more common
- Medical applications used in operating rooms are another area where 3D gestures have been explored, to maintain a sterile environment
- Gesture interfaces have also been used for symbolic input

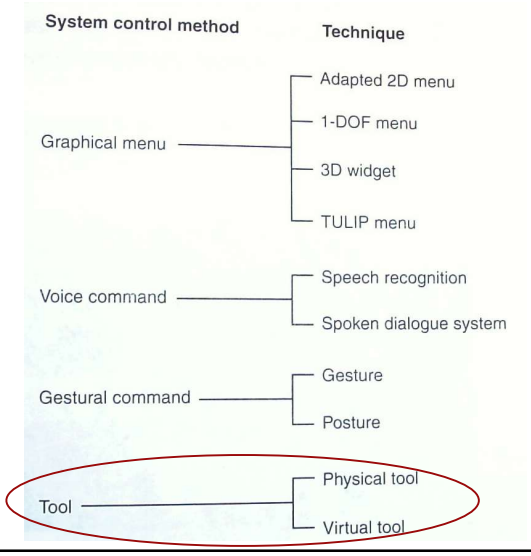
Gestural recognition

Cons:

- Often requires wearing gloves (camera-based recognition suffers from occlusion problems)
- The user has to learn the gestures.
- Effective for a few commands

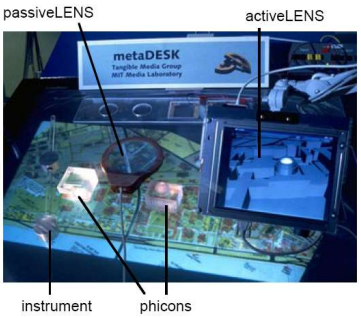
40

Tools



41

Tools



Use of familiar devices for 3DUI, exploiting the real-world correspondence: brush, erasers

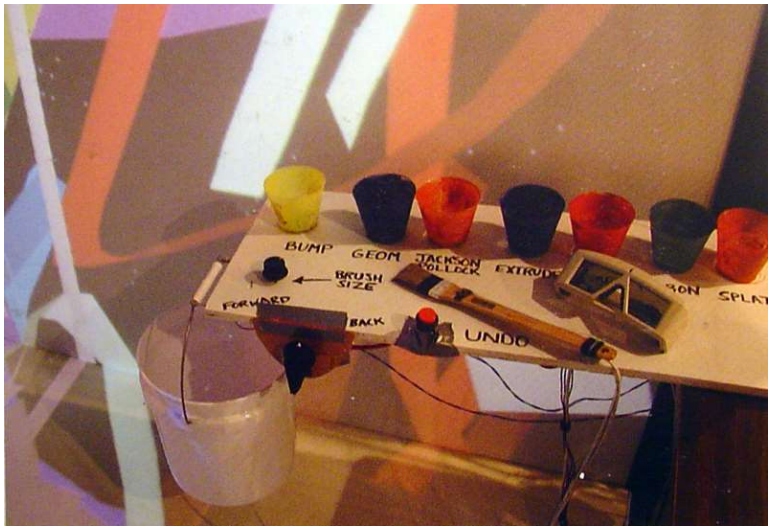
Types:

- Virtual tool
- Physical tool (=props)
  - The user holds the tool physically.
  - The system draws the corresponding virtual representation.
  - The user accesses a physical tool by simply picking it up and using it.
  - Tangible User Interfaces (TUIs)

[Video Metadesk MIT VideoLab]

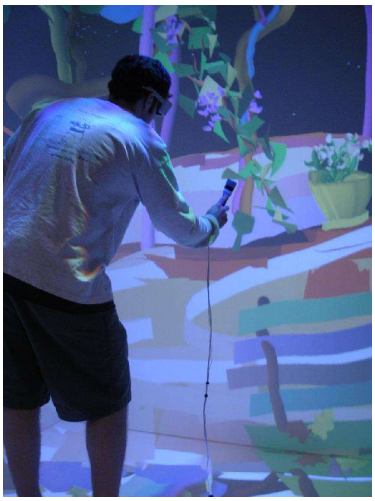
42

Tools video



43

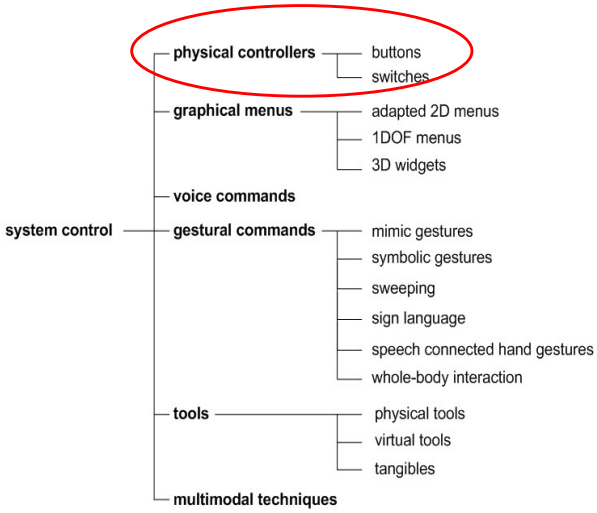
Tools



44

Classification (by LaViola, Jr et al. "3D User Interfaces")

- System control techniques can be classified as follows



45

Physical Controllers

Techniques

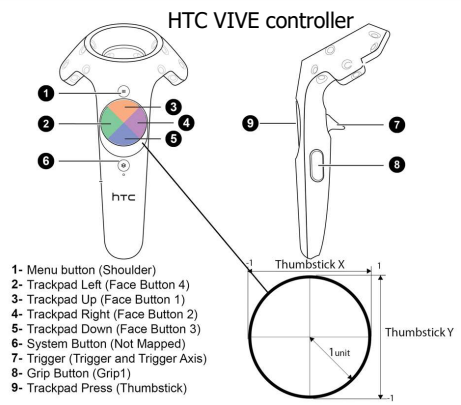
- Physical controllers such as buttons and switches offer a lightweight solution for performing system control
  - Analogous to function keys in desktops
  - Direct way of changing a mode in an application
  - In contrast to using a pointing device to select, for example, an item from a menu, the physical controller allows the user to directly switch the mode between different states

A flight joystick deploying numerous switches and buttons (© Guillemot Corporation S.A. All rights reserved. Thrustmaster ® is a registered trademark of Guillemot Corporation S.A.)

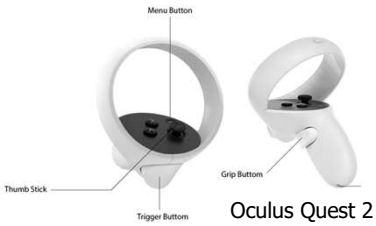
Physical Controllers

Design and Implementation Issues

- Placement and control
  - When built-in controllers are used, you should carefully validate their placement and the potential need for regrasping a device to access the button
  - Critically reflect physical form and quality, as some buttons and switches are difficult to control



Physical Controllers



Oculus Quest 2 controller

Design and Implementation Issues

- Representation and structure
  - Buttons and switches are not necessarily connected to any menu-like structure
    - Structure is based on the placement of buttons and their interrelationship
  - Button locations are often defined by accessibility (ergonomic placement) rather than by functional structure
    - Feedback changes should be clearly communicated to the user
  - Placing a small label or pictogram on the button can indicate its usage

Physical Controllers

Practical Application

- Buttons and switches are particularly useful
  - When users need to switch frequently between functions: can be lightweight, quick, and straightforward
  - In applications that are used for short durations by inexperienced users, function keys may be very useful, but only with a small functional space
  - If users have the time and motivation to learn more complicated sets of functions, this may come with a great increase in performance

Physical Controllers

- Can be visible in immersive VR
- We can even track the entire body -> body gestures

50

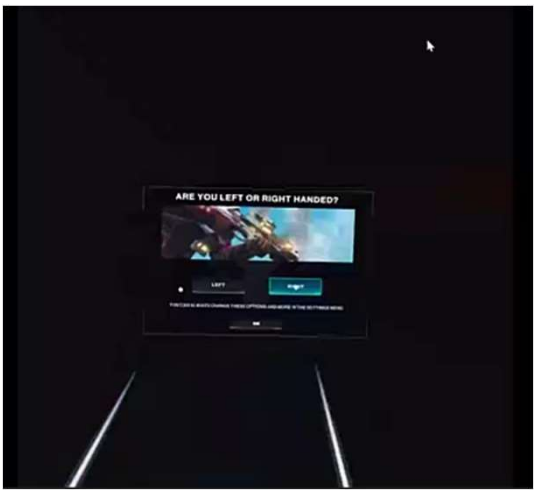
Design guidelines (for app control)

- Avoid disturbing the flow of action on an interaction task.
- Prevent unnecessary changes of the focus of attention.
- Avoid mode errors (provide feedback so that the user knows which interaction mode is currently active)
- Use appropriate spatial reference frame.
- Design for discoverability.
- Consider using multimodal input.

51

Case Study: Population One in Oculus Quest 2

- Tutorial to show the use how the controller buttons map into interaction/navigation/control

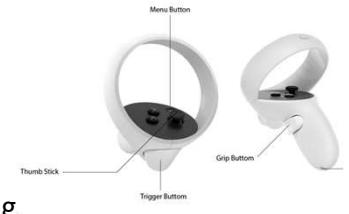


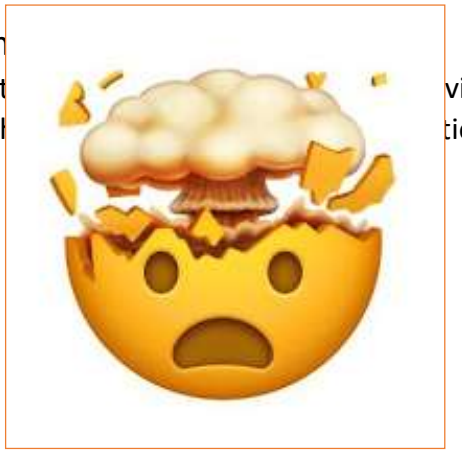
52

Design problems

- Example: Population
  - Trigger button (right
  - Trigger button (left

vig
tion/control)

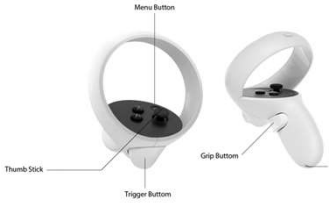




53

Design problems

- You need to remember all the controllers buttons and what they are for
  - While… not visible!
  - Cognitive distance between buttons and affordances (interaction/control/navigation)
  - Pressing accidentally several buttons simultaneously



54

Conclusion

- We have provided an overview of the various system control techniques that can be used for 3D
  - Though system control methods for 3D UIs have been developed and used extensively, many issues are still open for further research
  - There is also a lack of good empirical evidence for the user experience of various system control techniques at the present time

55