VR Technology
TNM086

Interaction and Navigation

Synopsis

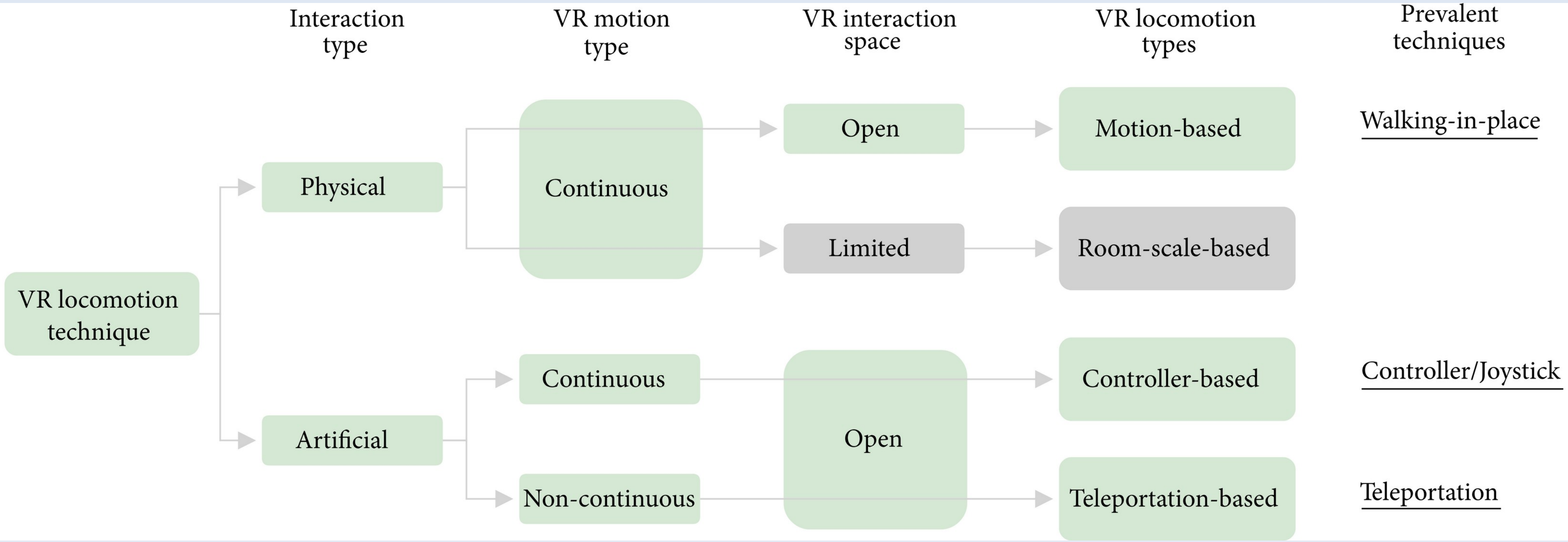
- Methods for navigation
- Methods for selection
- Manipulation
- Widgets

Fundamental Forms of Interaction

- Interaction
  - Navigation
  - Selection
  - Manipulation
  - Menus and widgets

Forms of Navigation

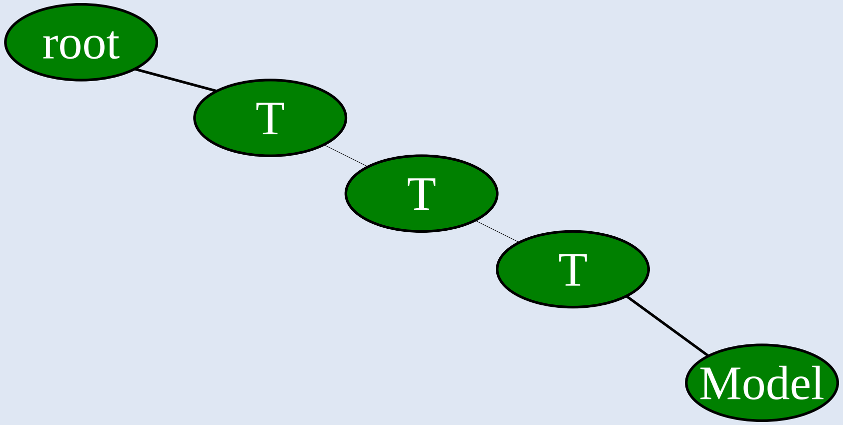
- Physical, limited, implicit navigation
- Physical, open, implicit navigation
- Artificial, explicit navigation
  - instantaneous or continuous

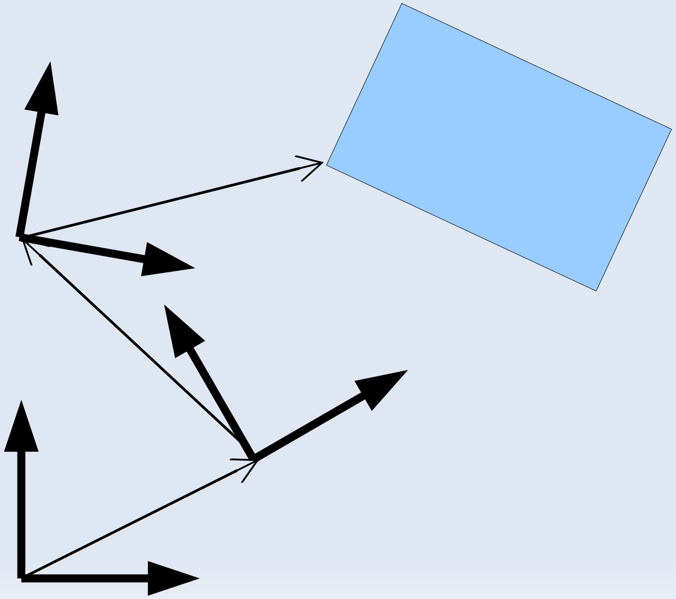


Boletsis and Cedergrens, 2019

Forms of Navigation

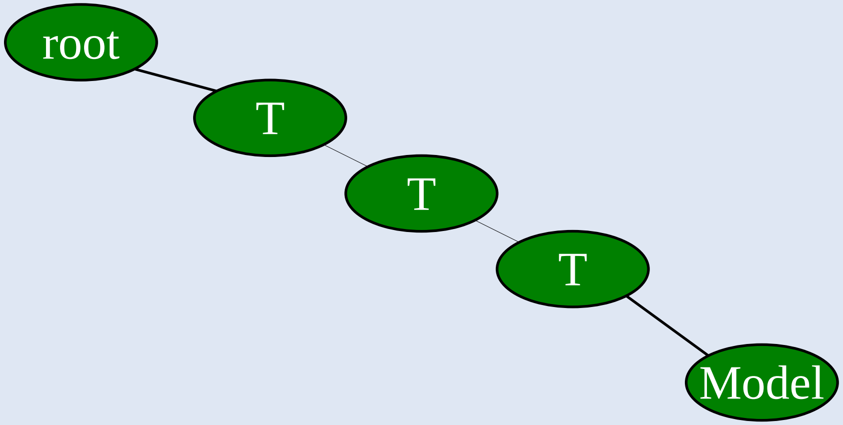
- What is navigation? What is a ”camera”?
    - Typical GL transform x' = P V M x
    - P = projection (frustum)
    - V = view (camera?)
    - M = model transform (everything else)
- Actually: $x' = P\ V\ M_N \cdots M_2\ M_1\ x$

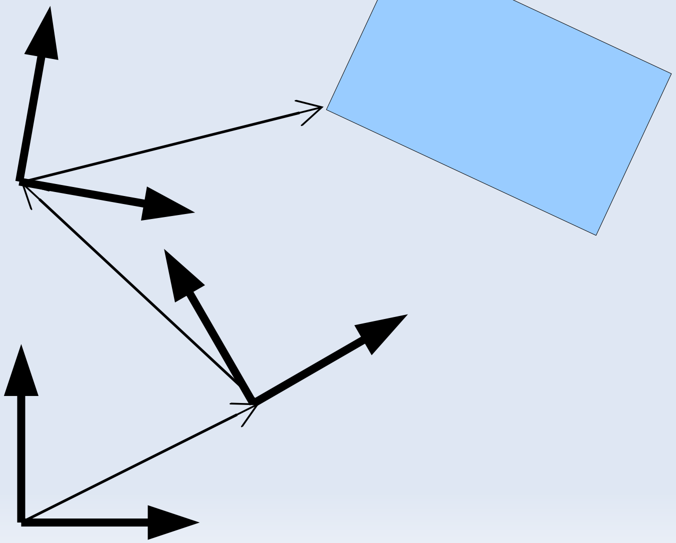




Forms of Navigation

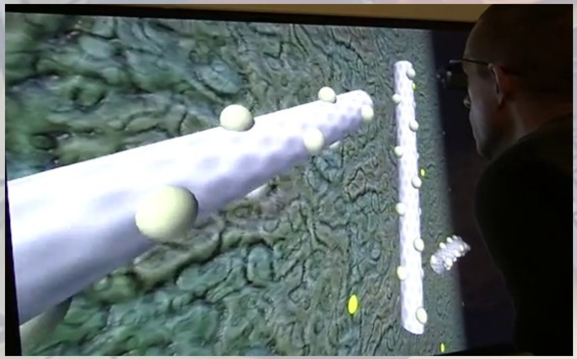
- Head-tracked view navigation
  - + Creative extensions of view navigation
- Instantaneous scene transformation
- Continuous scene transformation

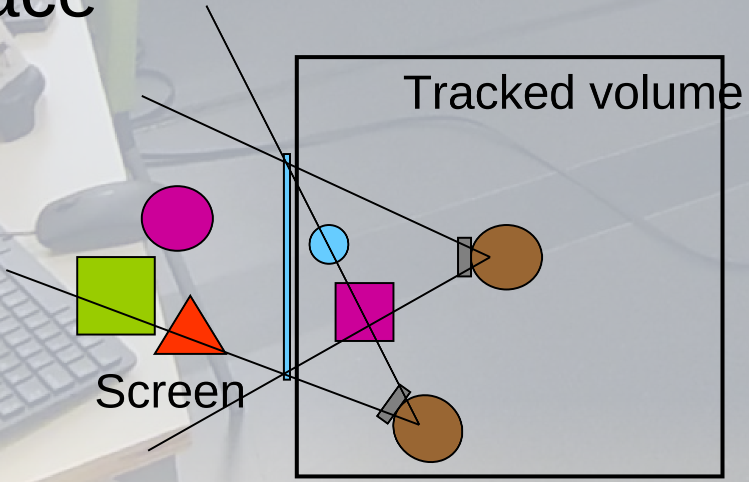




Implicit Navigation

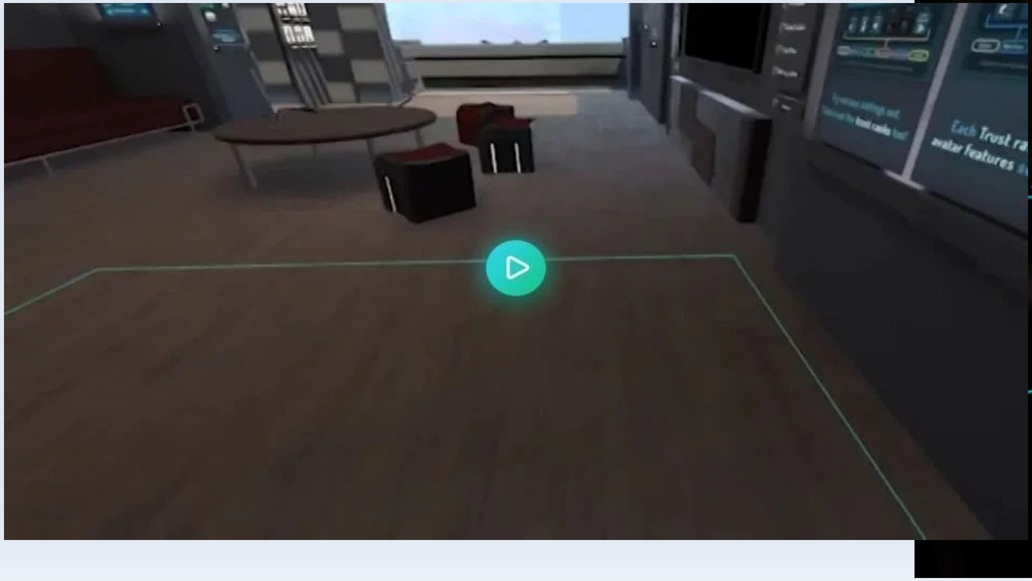
- Immersive display
  - Should give us ‘real world’ interaction
  - Walk to the object, turn around, etc
  - Easily by moving ”Viewpoint” or ”Camera”
- Dependent on display and tracking technology
  - CAVEs and HMDs vs workbench and workstation
  - Available space

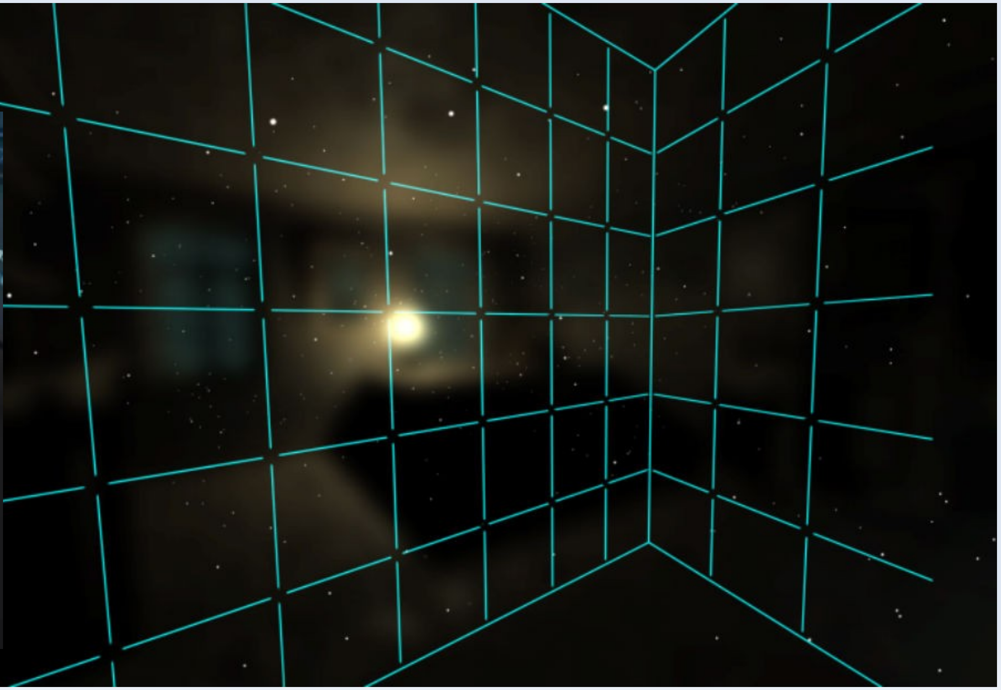




Boundary

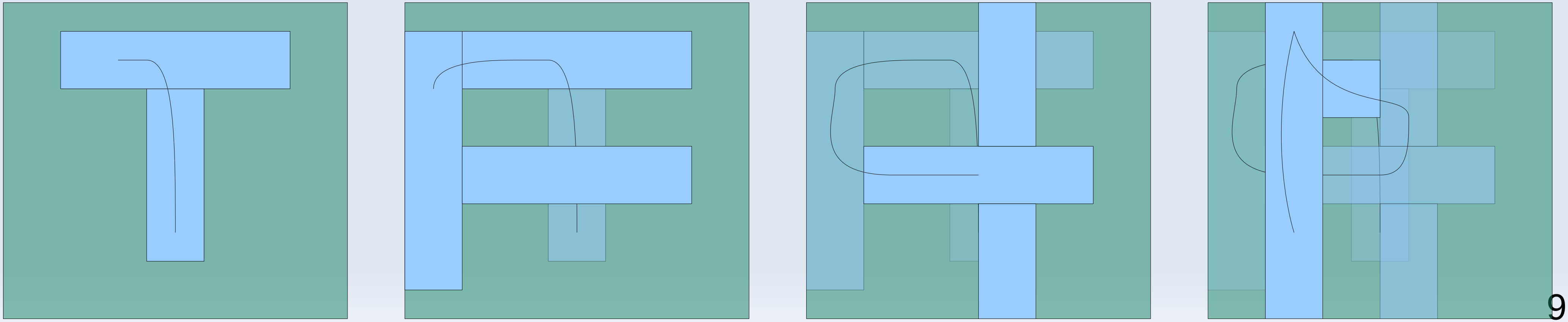
- ”Chaperone”
- Keep user inside of *safe* space
- Without destroying *immersion*





Smart Use of Space

- Redirected Walking
  - Rotate the scene to bend long straight walks
  - 22 m radius circular walk can feel straight
  - But rotation is *really* bad for cyber sickness
- Portals and Corridors
  - Dynamic scene that adapts to the real room
  - Maze-like space ship or dense thickets



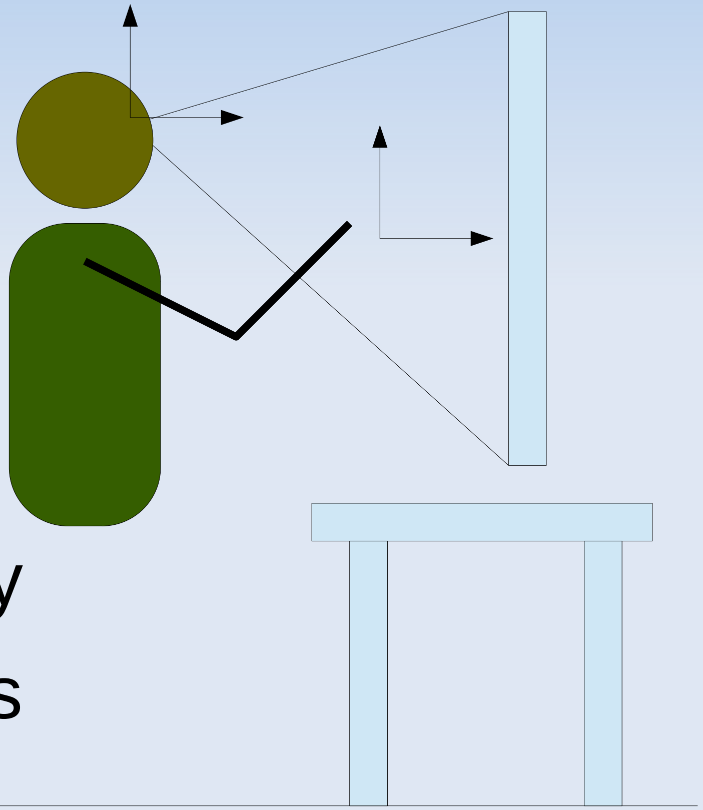
Scale and Navigate

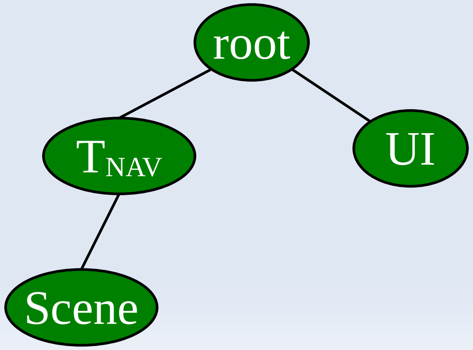
- Scaling and moving
    - to help reaching larger areas
    - e g walking through our solar system
- There is no ”zoom” in VR
    - scaling the world around user
    - (zoom is a change of FoV)

GulliVR … , Krekhov et al, 2018

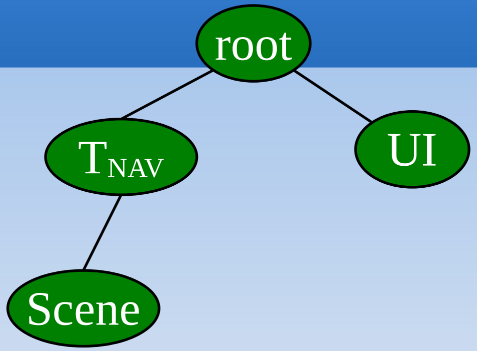
Frames of Reference

- VR display system
  - display system origin
  - head pose / HMD / eye pose(s)
- Navigational system
  - scene origin
  - moves objects relative the display
  - moves the display relative objects

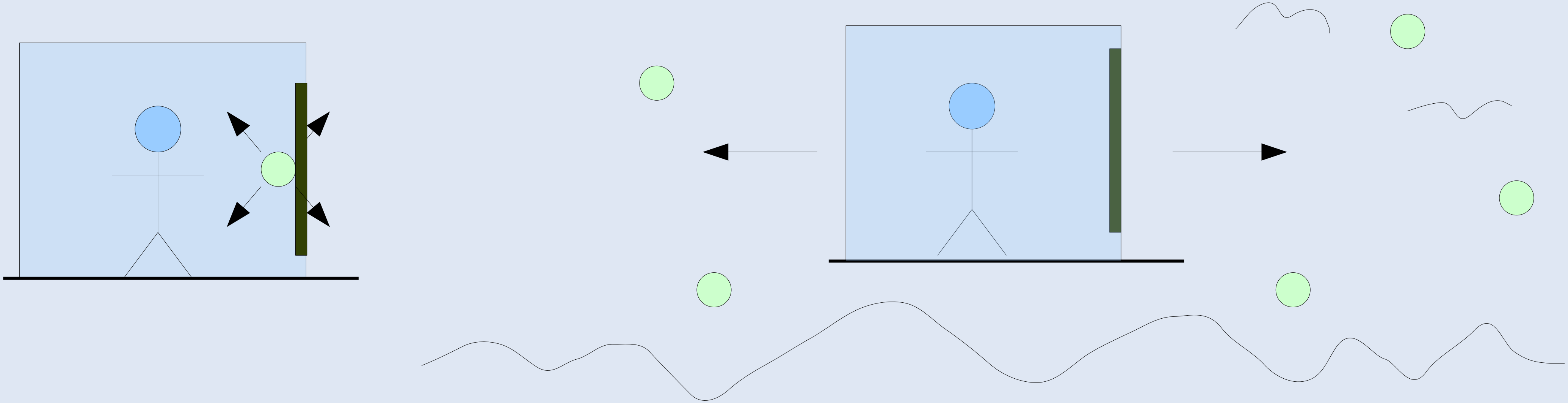




Navigation Transform

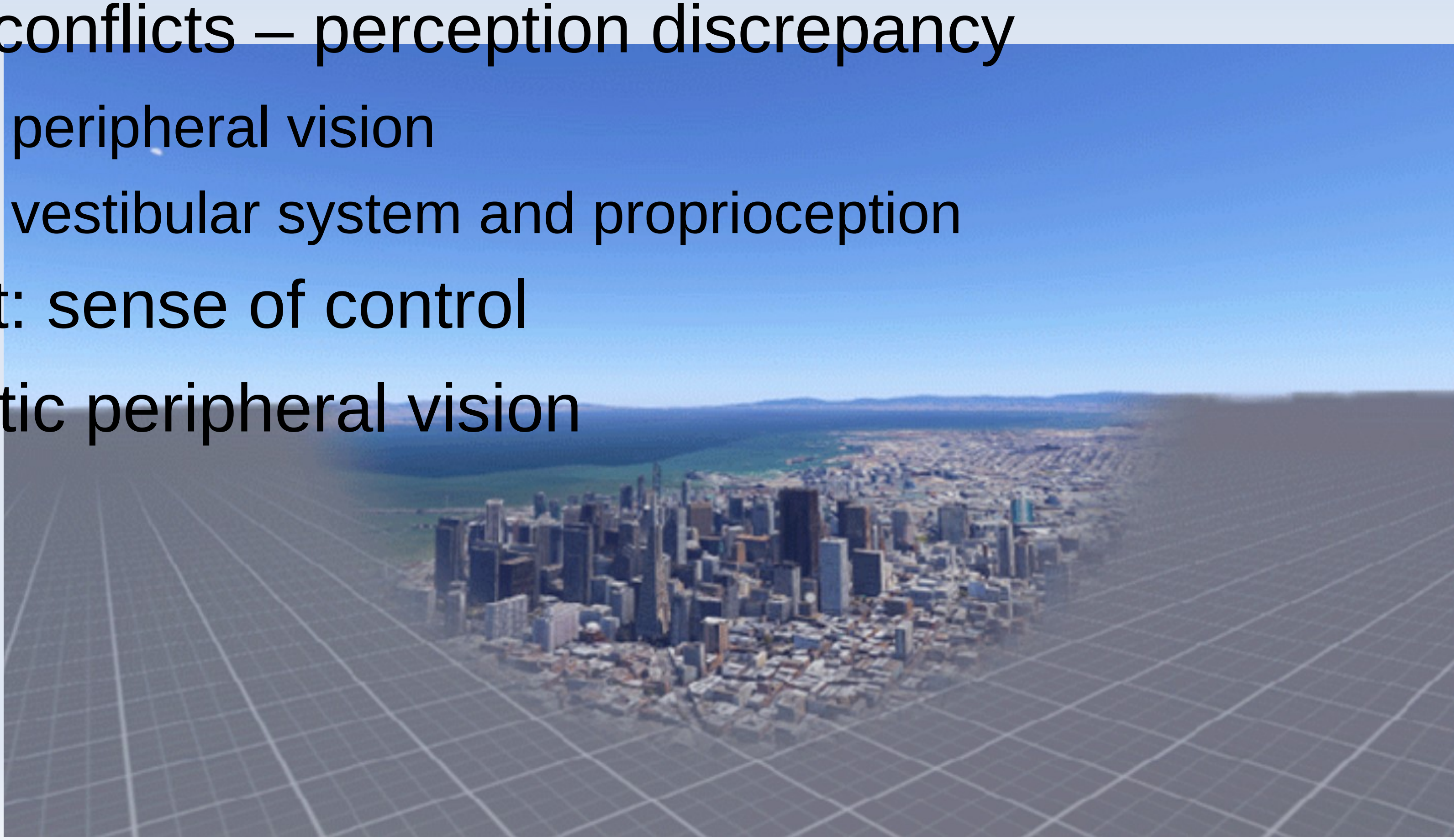


- Navigation moves the scene
  - Create a graphics origin offset
  - move objects relative the display – apply transform
  - move the display relative objects – apply inverse



Main Issue with Explicit Navigation

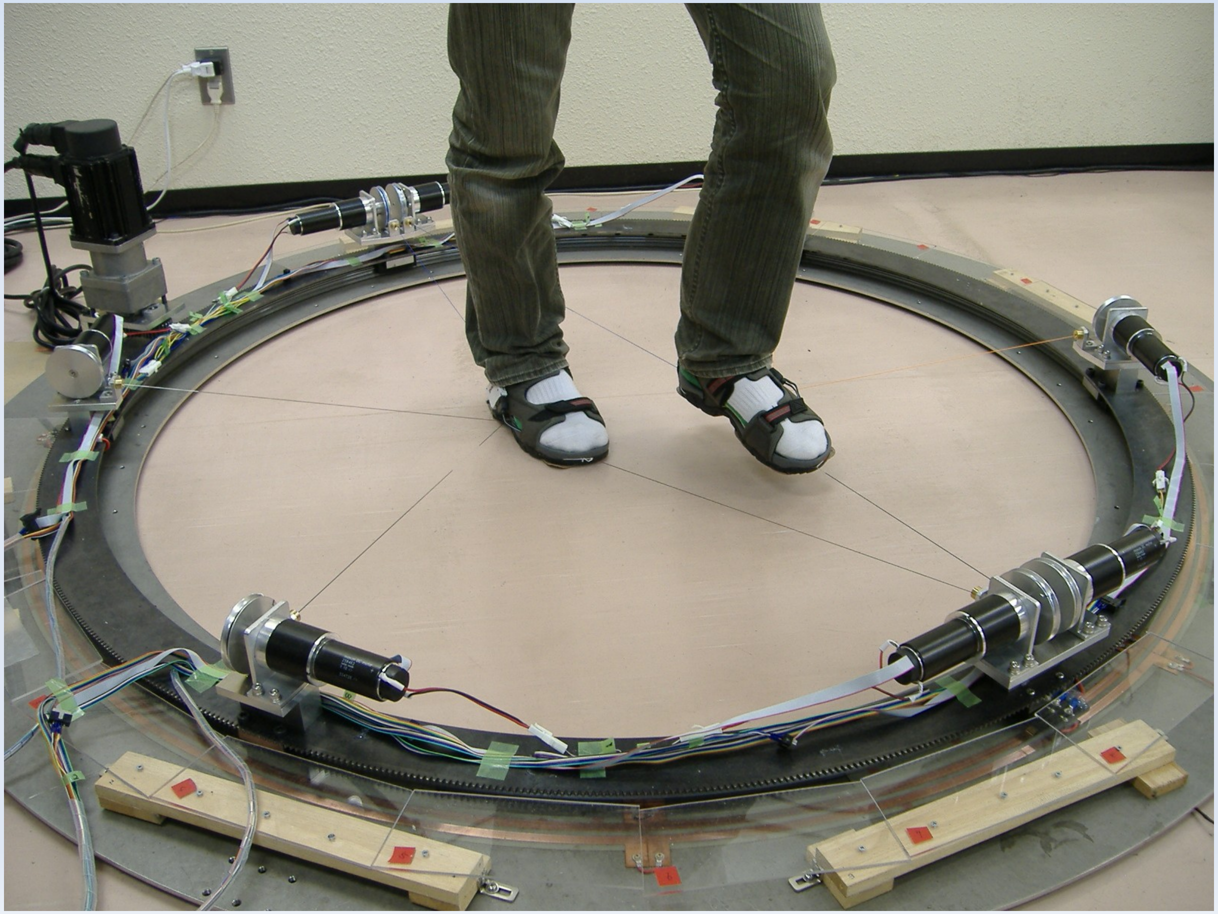
- Vection
  - Sense of movement when body is stationary
- → Cybersickness!
  - Sensory conflicts – perception discrepancy
    - peripheral vision
    - vestibular system and proprioception
  - Important: sense of control
  - Trick: static peripheral vision



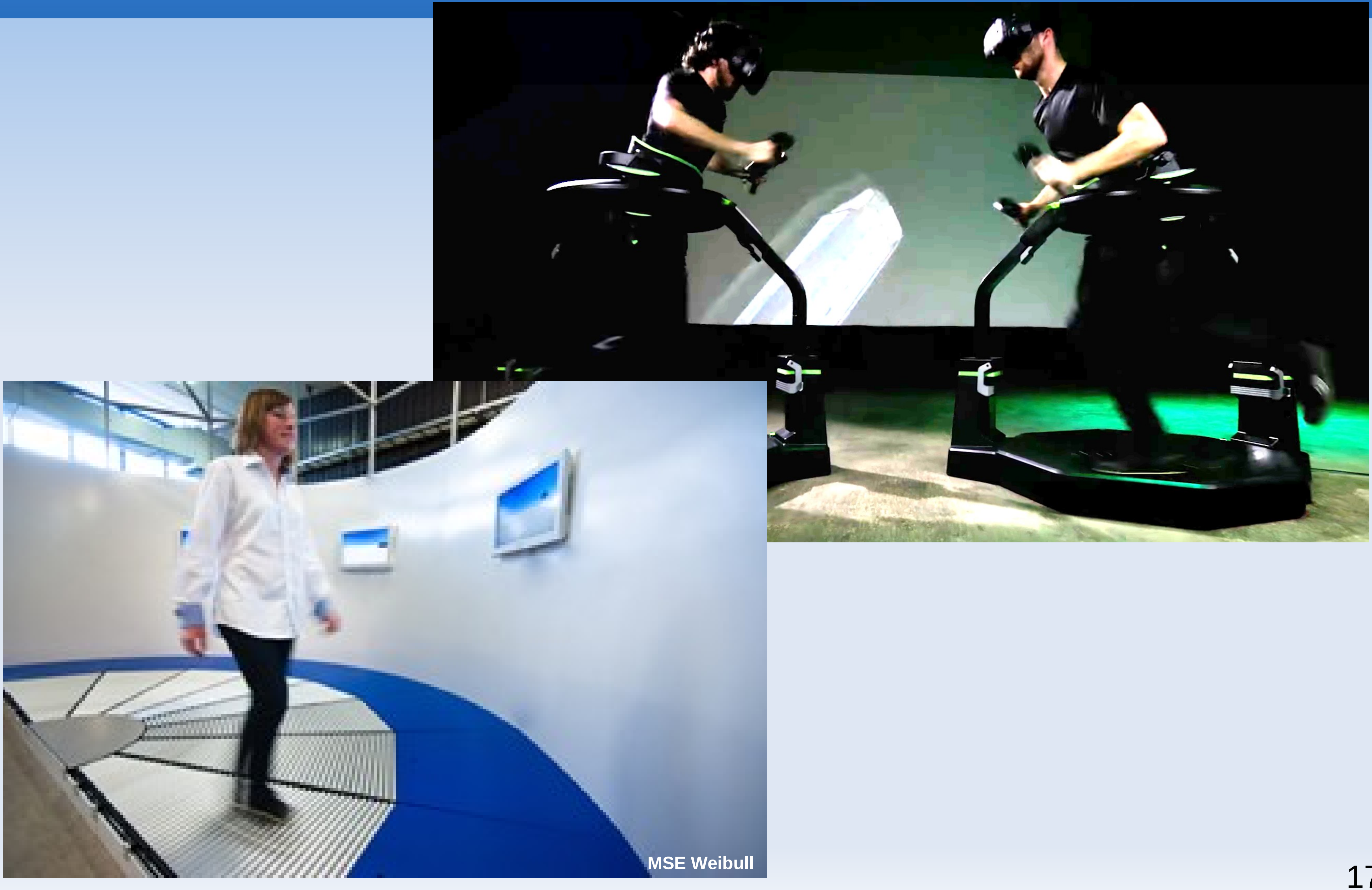
Locomotion Interfaces

- Navigate by stationary motion
  - Treadmill,
  - simulate ground – toroidal topology

Locomotion Interfaces



Locomotion Interfaces

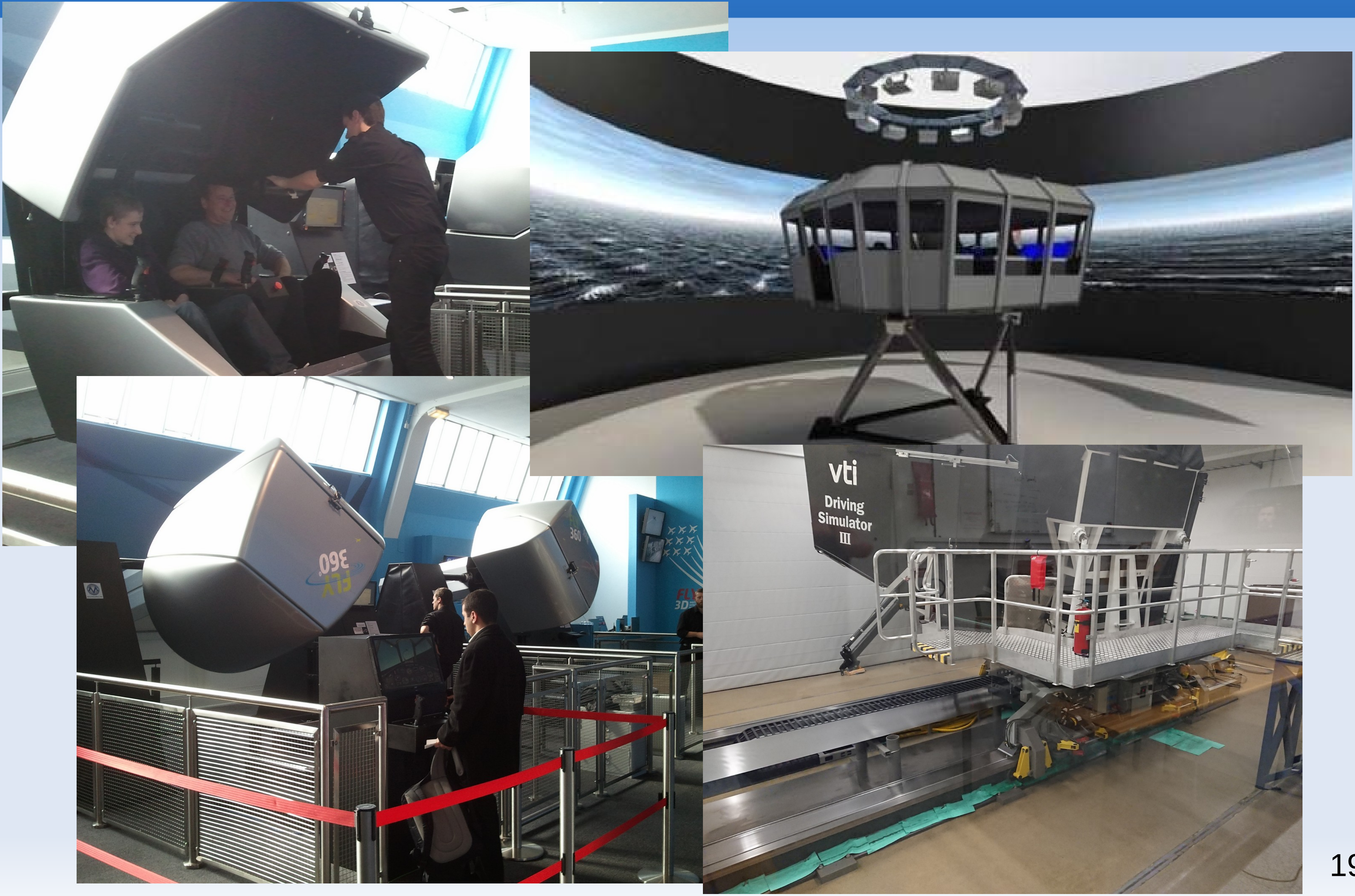


MSE Weibull

Locomotion Interfaces

- Characteristics
    - need to be fast, light and exact!
  - balance issues
    - due to vection – motion / momentum discepancy
    - due to mechanics momentum
    - due to system inaccuracy
  - unrealistic
    - simulation of hard flat surfaces
    - no stairs, mud/snow/gravel, texture, unevenness

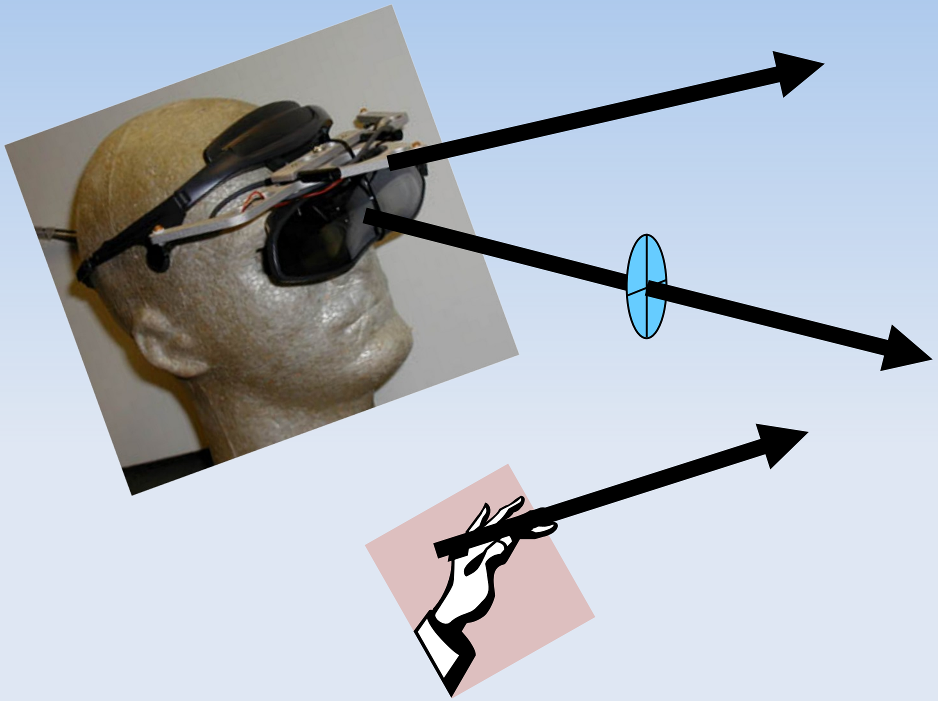
Motorized Platforms



Explicit Navigation

- Explicitly control the navigation
  - Applying some metaphor
  - Select where to go/fly/walk
- Typical metaphors
  - Walking/flying
  - Goal driven navigation
  - Object driven navigation

Fly/Walk Direction Control



- Gaze direction
  - Natural feeling
  - Can't watch scene go by
  - Common in FPS
- Pointing mode
  - Point at desired direction

$$\vec{p}\,' = \vec{p} + \mathrm{t}_{\Delta} v \hat{u} = \vec{p} + \mathrm{t}_{\Delta} v R \hat{z}$$

  - 5 DoF device tracking required
- Crosshair mode
  - Eye/hand line defines direction

$$\vec{p}\,' = \vec{p} + t_{\Delta} v \frac{x_{hand} - x_{head}}{\left\| x_{hand} - x_{head} \right\|}$$

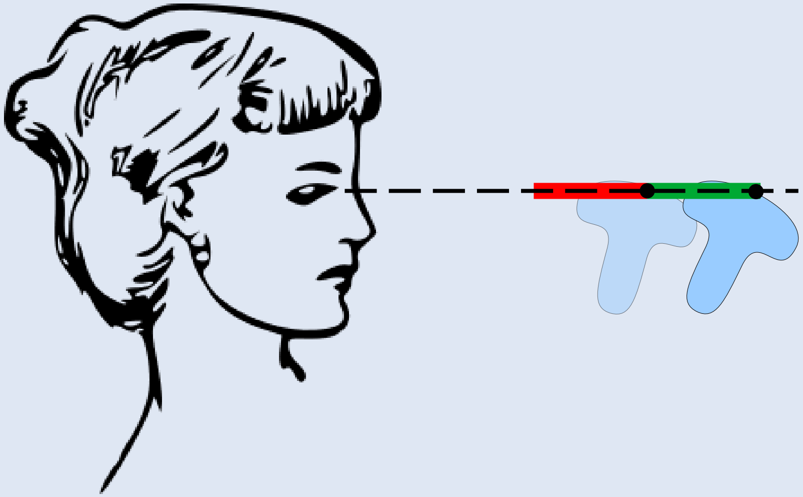
Speed Control

- Explicitly controlled speed
  - Physical controls (joystick)
  - Gestures
    - arm swinging, feet stamping – walk-in-place
  - Hand movements
- Access to a range of speeds
  - What range
    - defined by natural walk/running speed
    - defined by size of the world
    - defined by the distance to closest object

Hand Controlled Speed

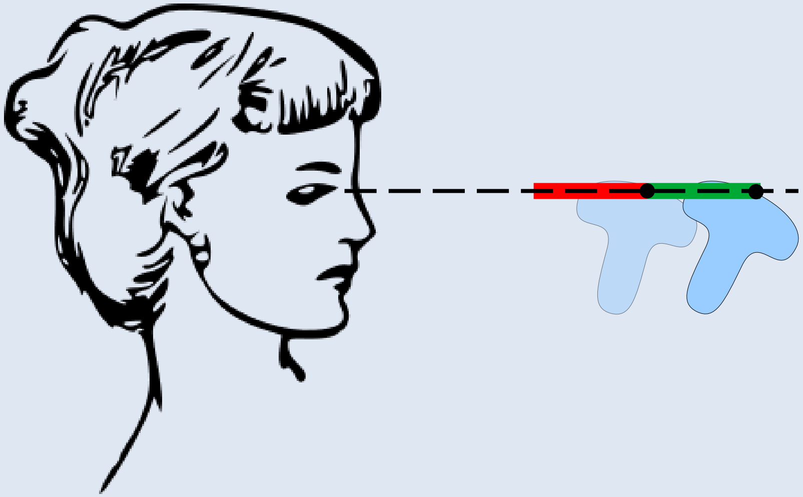
- Distance
  - Control velocity with head-to-hand distance
  - Use initial head-to-hand distance as zero velocity

- Used with crosshair control
  - Intuitive
  - Natural mapping
  - Limited range



Controlled Acceleration

- Acceleration allows for larger speed range
  - Push forward to accelerate
  - Backwards to decelerate
  - Harder to control
- When we release navigation button...
  - Immediate stop?
  - Soft deceleration?
  - Continue flying?



Problems w/ Hand-related Control

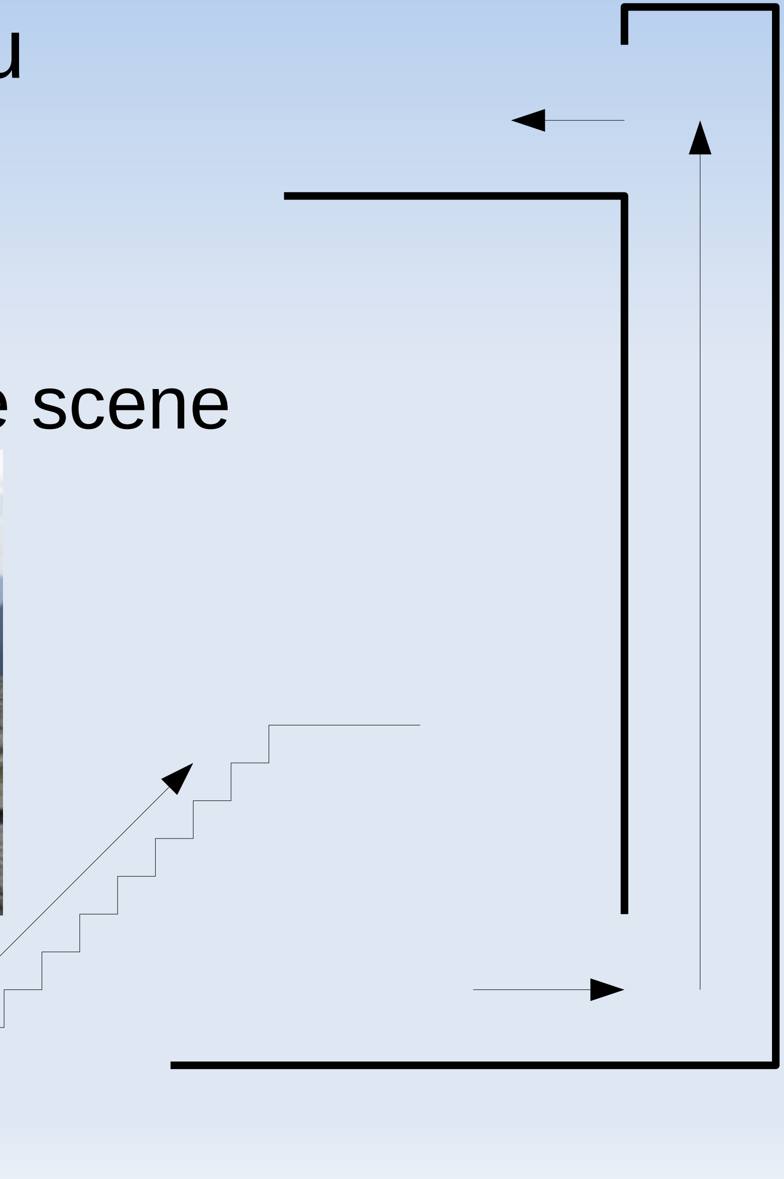
- Fatigue
  - Large and high hand movements
  - Long periods of use
  - Non-temporary use requires other methods
- Must be easy to turn on/off
  - Ordinary interaction causes motion
  - Buttons, gestures, etc

Goal Driven Navigation

- Fly-to (teleport or motion)
  - Click on pre-defined ”bookmarks”
  - Magic telescope
  - Wand-based
- Automatic navigation
  - Pre-selected ”good” path
- Virtual map
  - Select position you want to be at
  - Type of widget

Object Driven Navigation

- Objects that transports you
  - Lifts, stairways, teleporters
  - Instant or soft motion
  - Naturally integrated into the scene

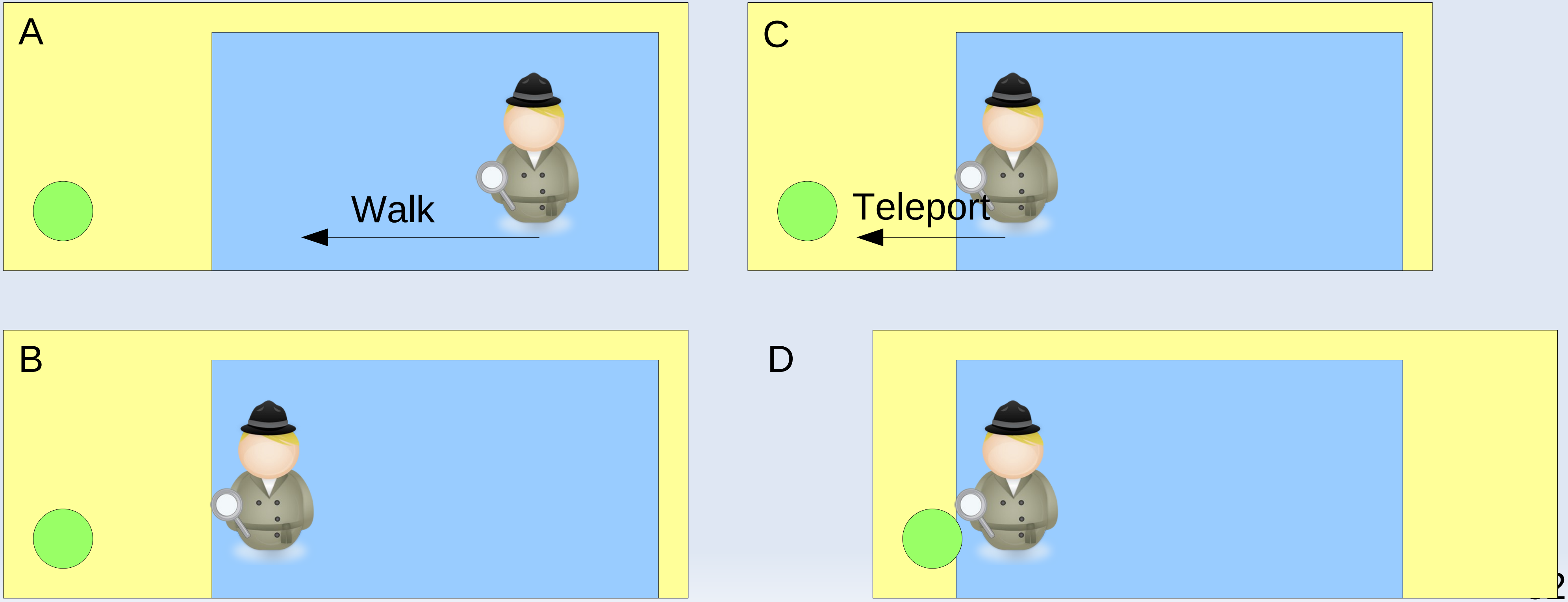


Goal and Object Driven Navigation

- Challenges and issues
  - Need to know where people want to go
    - Virtual architecture
  - Poor perceived control
    - May lead to cyber sickness
    - Provide as much control as possible
    - Use soft or instant motions
  - Possible disorientation
    - Especially with teleporters
    - Provide navigational cues
    - Use translation, avoid rotation

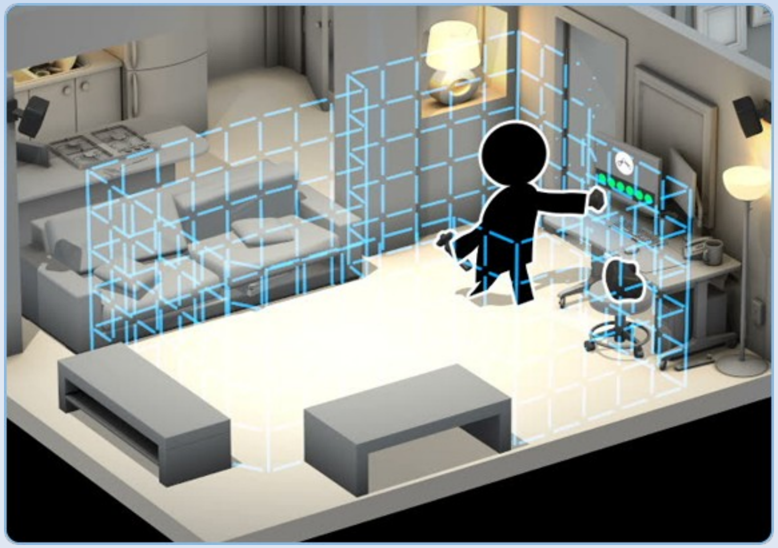
Compound Navigation

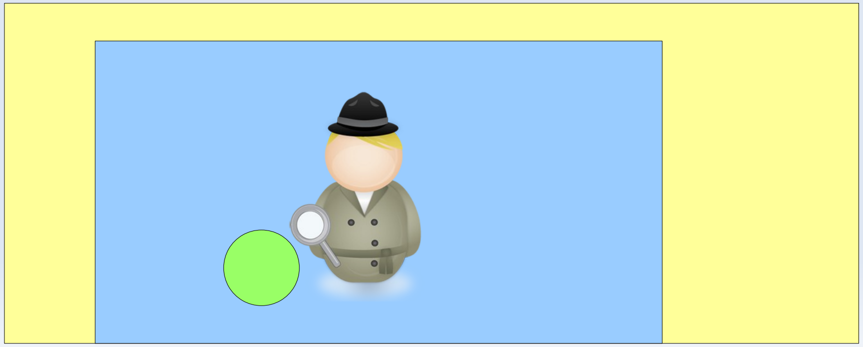
- Combination of navigation techniques
  - Local navigation – implicit
  - Larger distances – explicit
  - Help the user stay inside their workspace



Compound Navigation

- Explicit information (affects immersion)
  - show virtual boundary or obstacle
    - Steam VR Chaperone
  - show real obstacle (camera)
  - vibrotactile feedback
  - fade out virtual vision
- Implicit guidance (difficult)
  - Put virtual objects in the way
  - Move things in a smart way





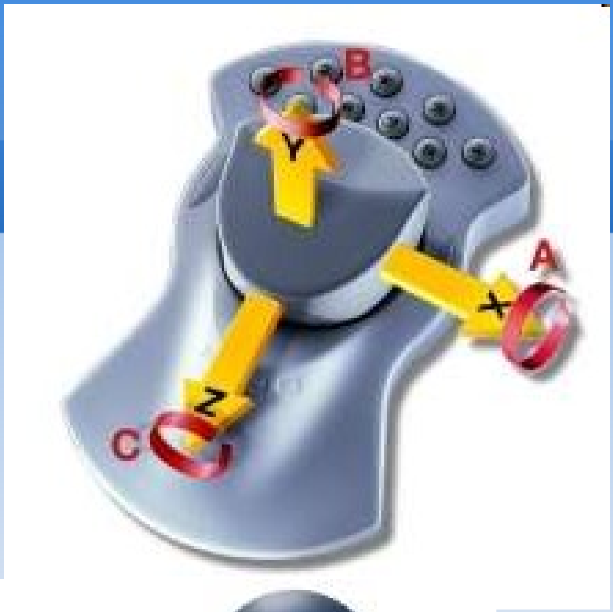
Navigation Summary

- Mix of different navigational metaphors
  - Physical navigation in local space
  - Explicit navigation over larger distances
    - Teleporters and maps
    - Important to make it natural, intuitive
- Adapt to hardware, users, tasks, situation
  - Game, CAD, surgery, etc
  - Available workspace, controls, haptics, etc
  - Long or short time use, complex tasks, etc

Common Input Devices

- General purpose devices
  - 3D Mouse
  - Wand / stylus / controller ← the thing!
  - Glove / suite
- Then there are specialized devices/equipment
  - Joystick or wheel
  - ”Real” controls

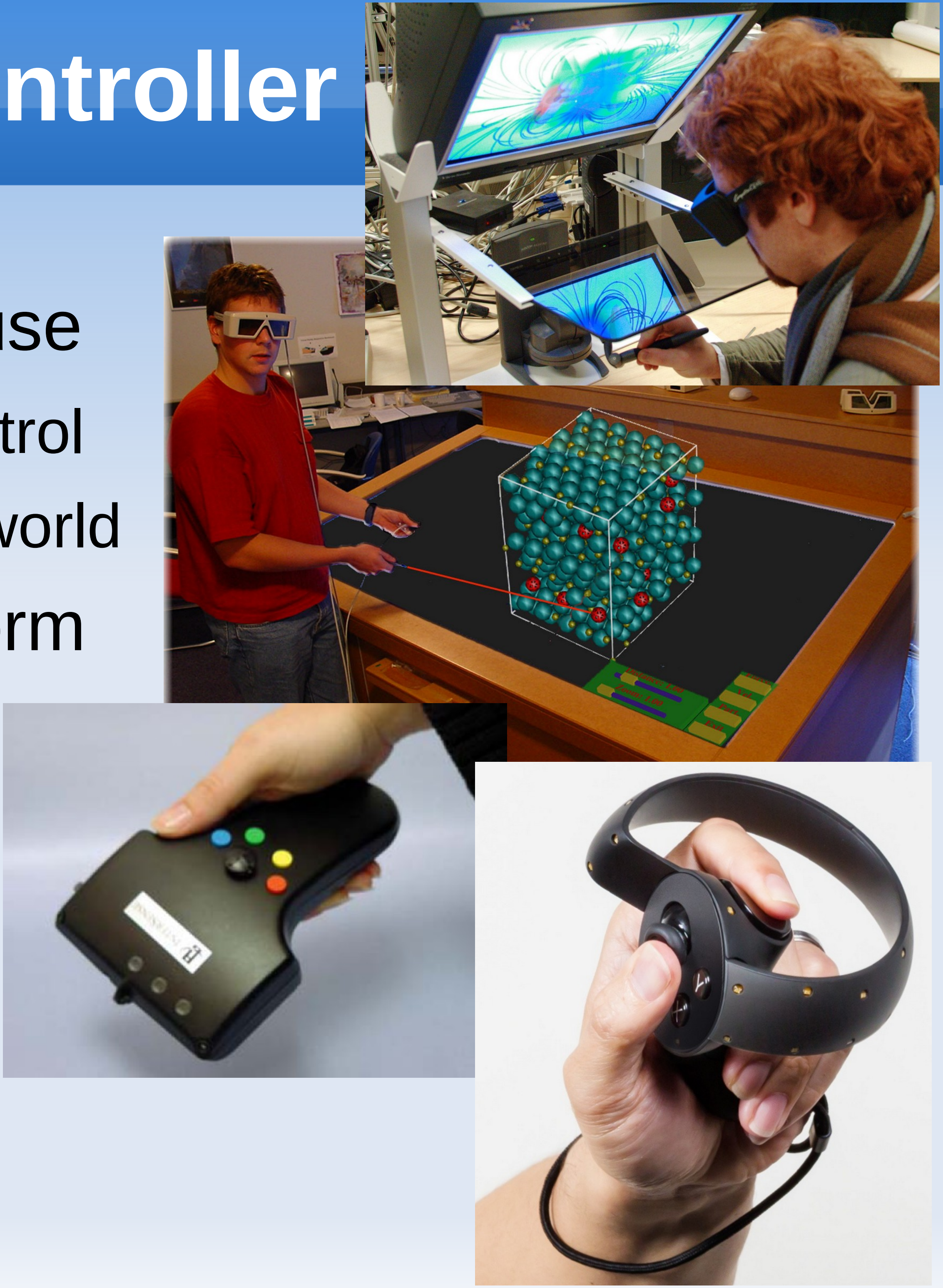
Mouse



- Magellan SpaceMouse
  - Force input, not position
  - Control velocity not position
- Interaction
  - Mouse pointer
  - Control object transform
  - Navigate
- 6 DoF velocity control is hard to use
  - Reduce DoF or use major axis

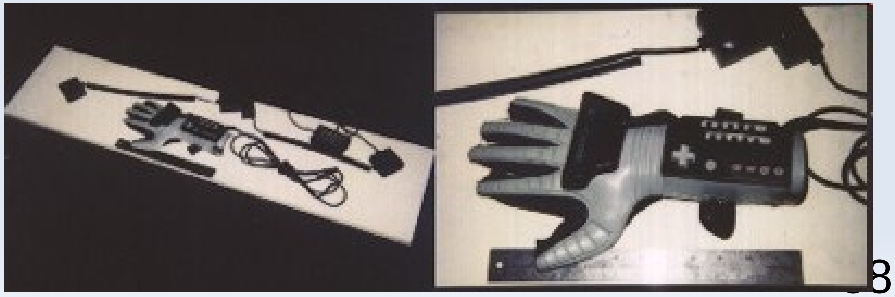
Wand/Stylus/Controller

- Extension of 2D mouse
  - Real 3D position control
  - Co-located with VR world
- 3–6 DoF in usable form
  - Hybrid tracking
  - Mechanical tracking
- Interaction
  - Pointing device
  - Touch, pick, grab
  - Buttons



Glove

- Hand position tracking
- Finger posture tracking
  - Gestures
  - Commands, e g picking
- Low demand, high price
  - Optical fibres and sensors
  - May change with haptic gloves
- Interaction
  - Select, pick, fly, menus
  - Sign language commands



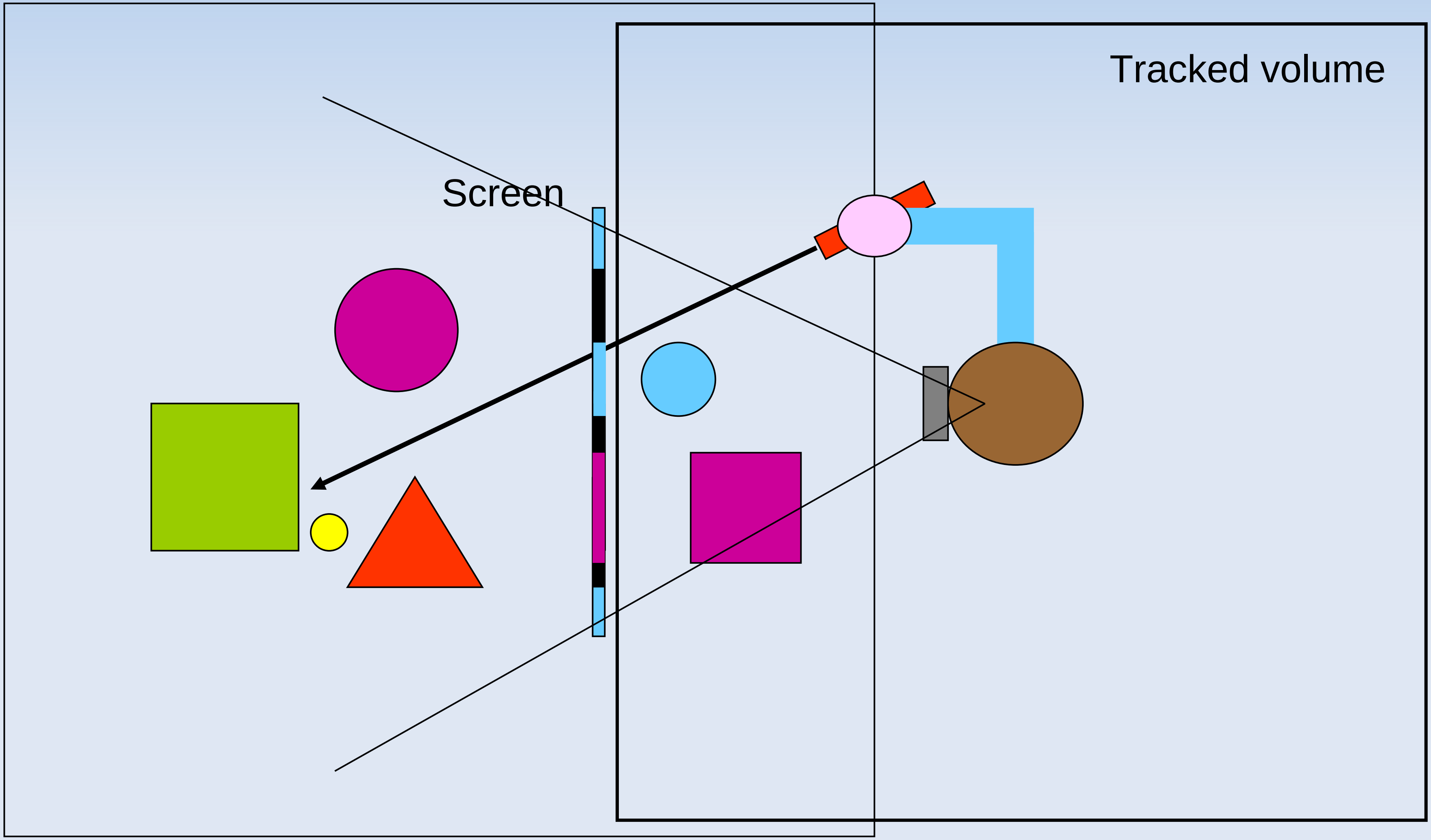
Suit

- Extension of glove
  - Trackers and flexion detectors
    - Mechanical exoskeleton
    - Magnetic, optical or ultrasonic trackers
  - Monitor body position
- Mostly for motion capture
  - Copy motion to avatar
  - Little use in interaction

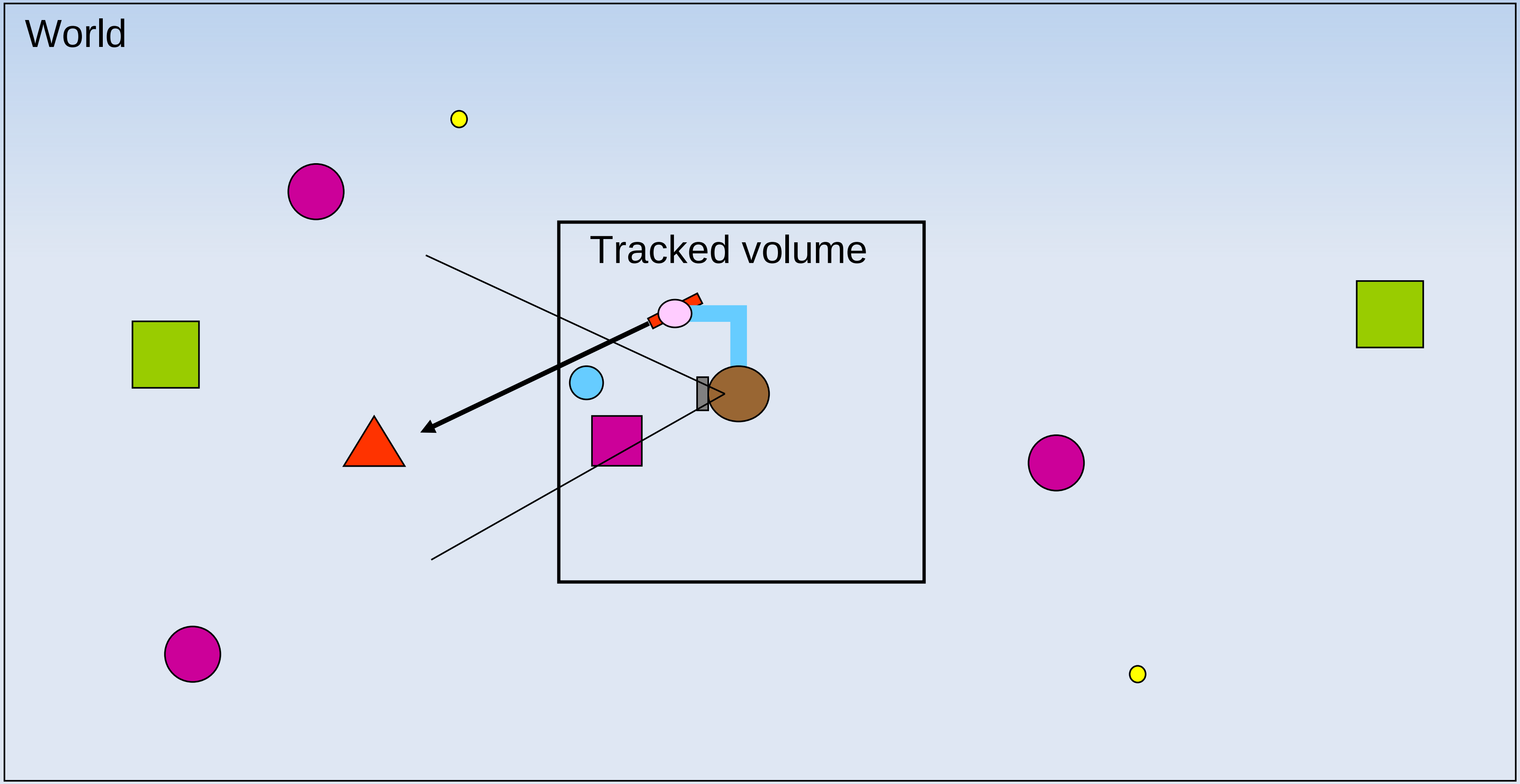
What Device to Use

- Select device
    - …based on application at hand
    - (and possibly availability)
- DoF, accuracy and precision
    - …based on the application at hand
- Feature set, ease of use and ergonomics
    - …based on the application at hand

Situation of Interaction



Situation of Interaction

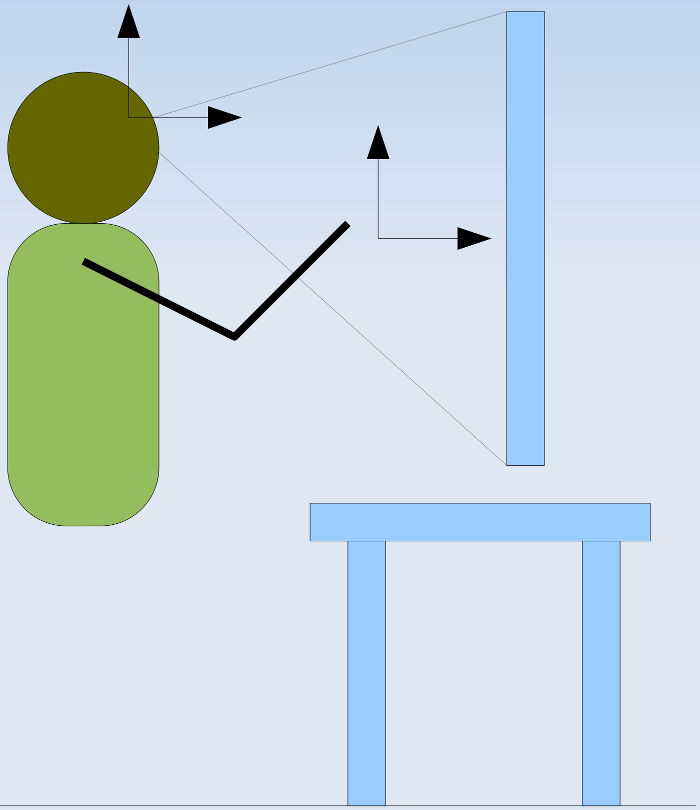


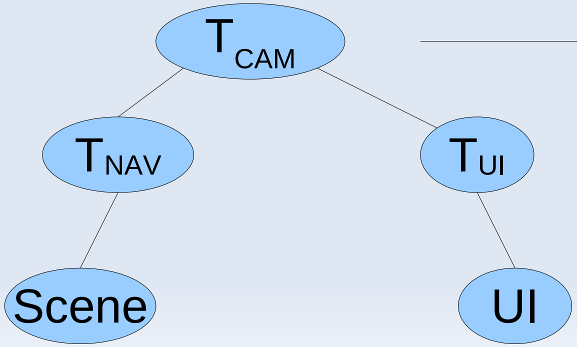
Selection Calculations

- Eye coordinates changes over time
  - … with head tracking and between eyes
  - So don't use 2D picking!
    - Not one image to work in – left or right?
    - Picking in 3D is generally not a 2D problem
- Use 3D coordinates
  - Linear algebra – closest object or intersection
  - Explicit handling of coordinates
    - Check your frame of reference!
    - Where is your wand's coordinate system?

Frames of Reference

- VR display system
  - graphics origin frame (world)
  - eye position
- Navigational coordinates
  - graphics origin offset
  - move objects relative the display
  - move the display relative objects



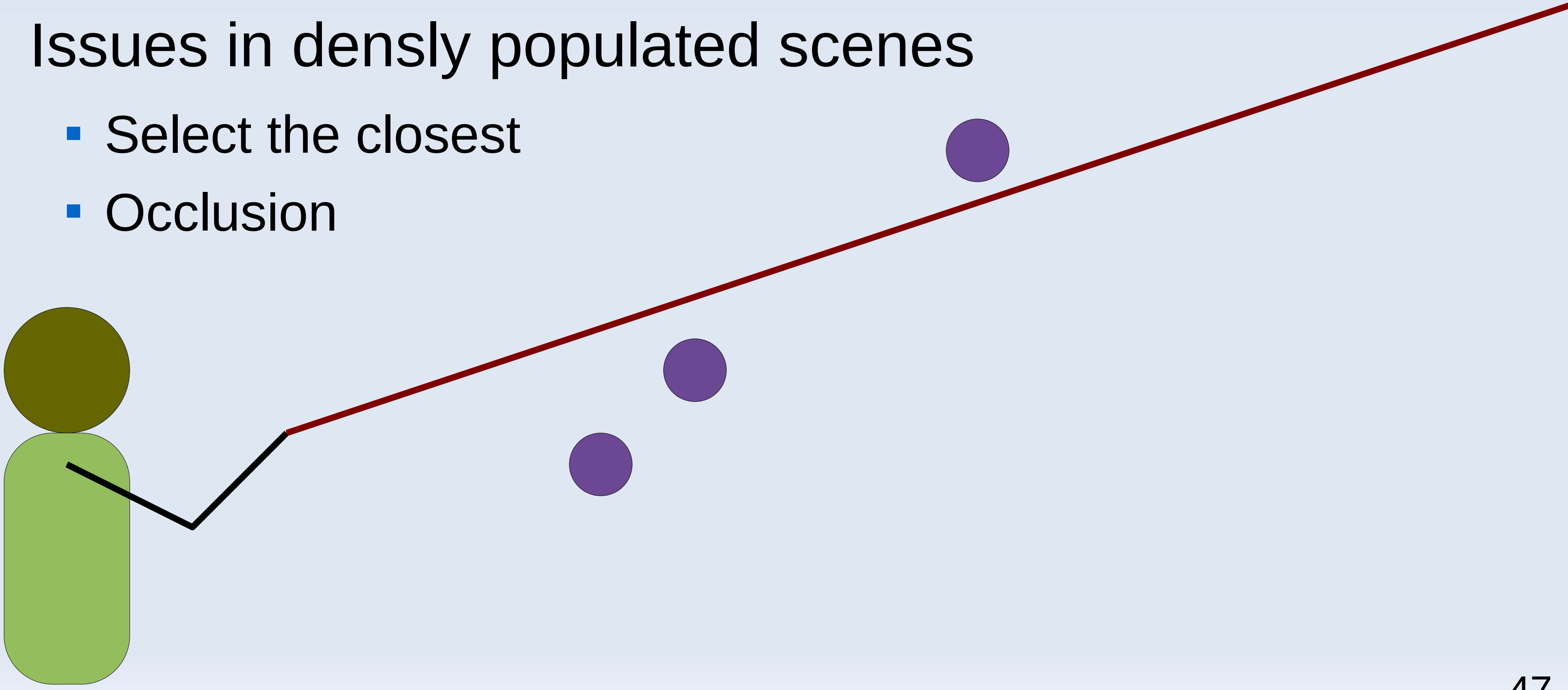


Selection

- Close – high quality hand-eye coordination
  - Touch metaphor
    - Walk to the object, reach out and touch
    - Impossible if the object is behind the screen
    - Surprisingly hard to use (no touch – need feedback!)
  - Encircle with line (typically on virtual surfaces)
  - Select with pointer/sphere
- Far away
  - Need extended reach
  - Less exact of various reasons

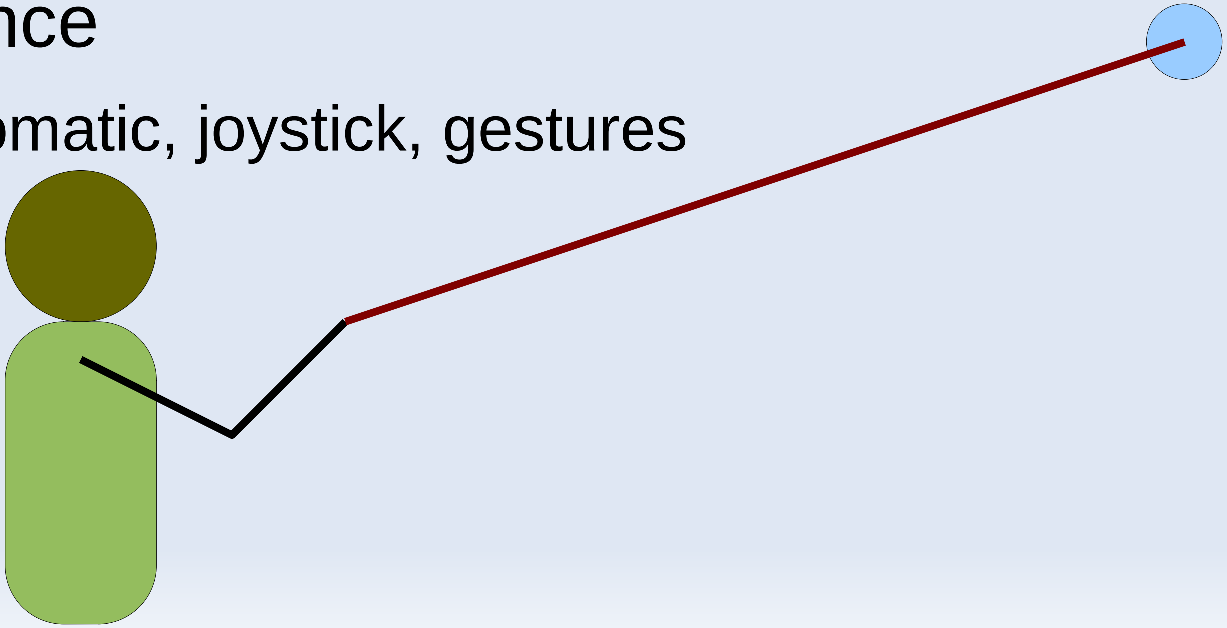
Extending Your Reach

- Pointer metaphor
  - Point at objects to select
    - High precision 6 DoF tracker required
    - Rotation w long lever = large end-point motion
  - Issues in densly populated scenes
    - Select the closest
    - Occlusion

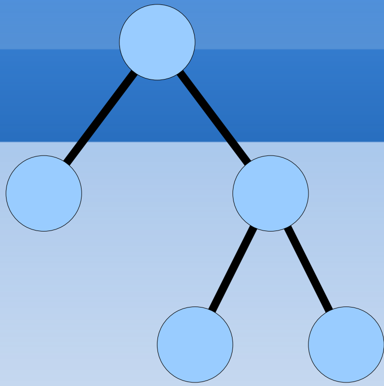


Extending Your Reach

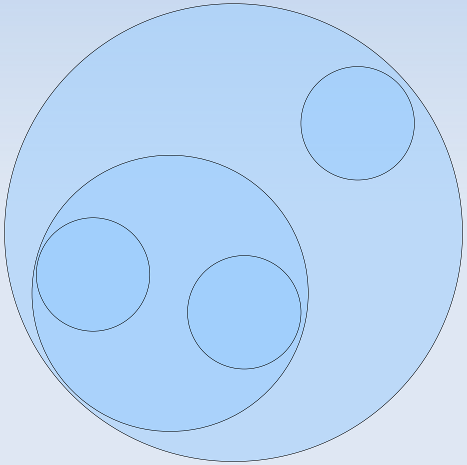
- Mouse pointer on a stick
  - Extendable stick
    - High precision 6 DoF tracker required
    - Rotation w long lever = large end-point motion
  - The tip is the active pointer
  - Control distance
    - Automatic, joystick, gestures



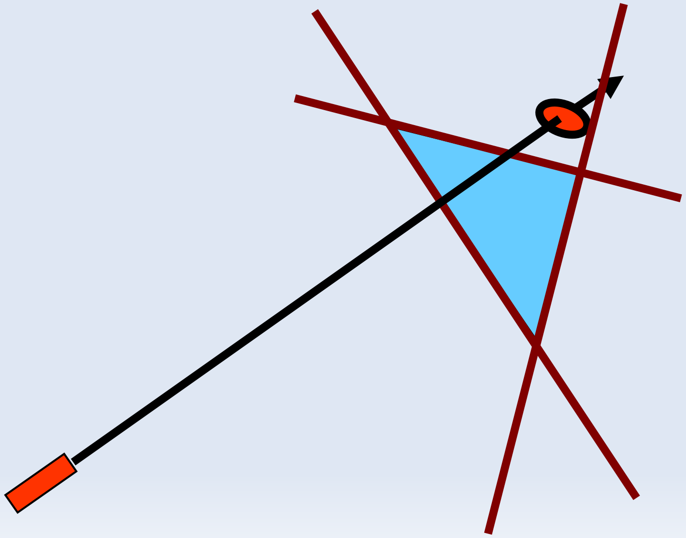
Intersection Checking



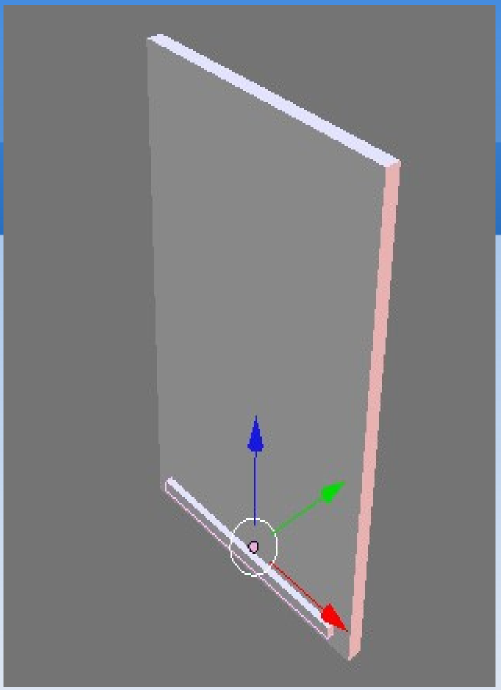
- Bounding box checking in scene
  - Check the group during traversial
  - Can any object here be intersected?
  - Use early termination!



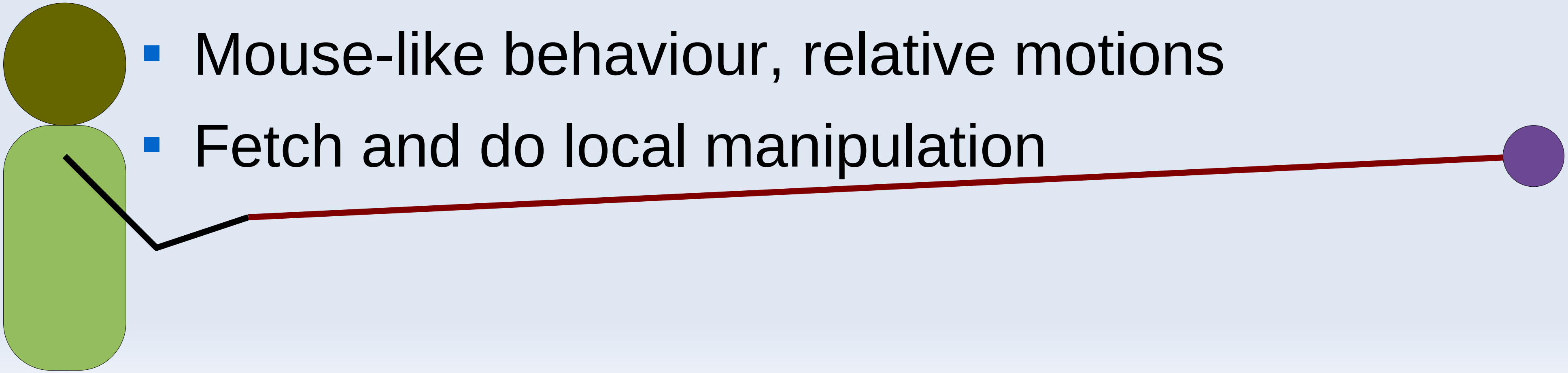
- 3D subdivision of larger objects
  - Binary space partitioning tree (BSP tree)
  - O(log n) search time complexity
- Polygonal objects
  - Ignore back faces
  - Faster if you saved the normal



Manipulation

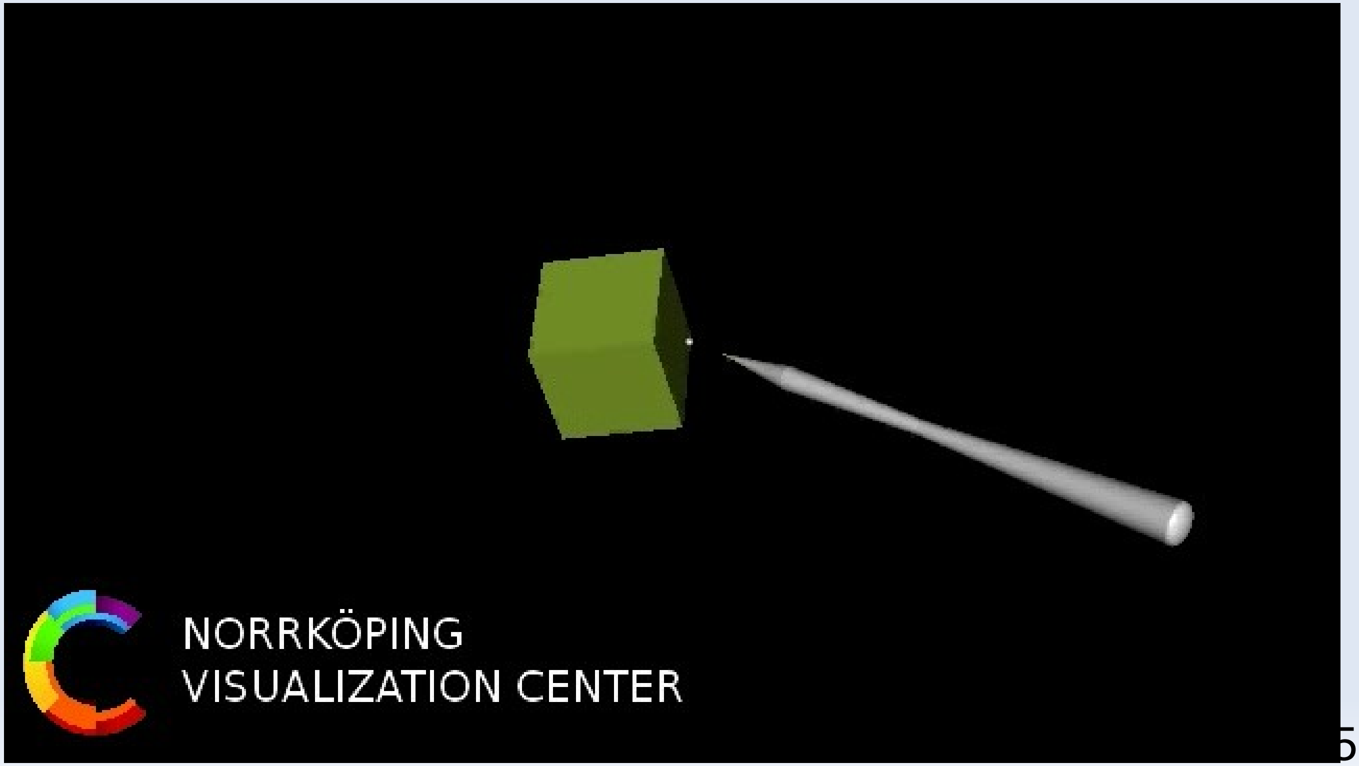


- Natural interaction
  - Pose is mostly no problem – grab
    - got 6 DoF, then use 6 DoF
  - Two hands? Drag to rescale and rotate
  - One hand? Need handles or commands
- At a distance
  - Grab is suboptimal – HOMER is better
  - Mouse-like behaviour, relative motions
  - Fetch and do local manipulation



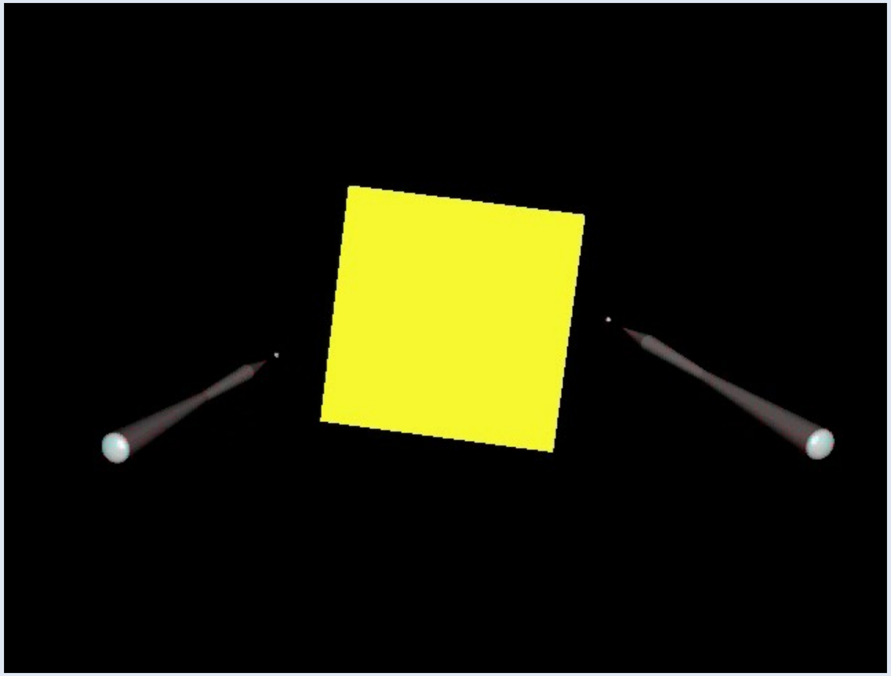
Moving, Rotating, Scaling

- One hand metaphors
  - grab, move, rotate
  - zoom/scale

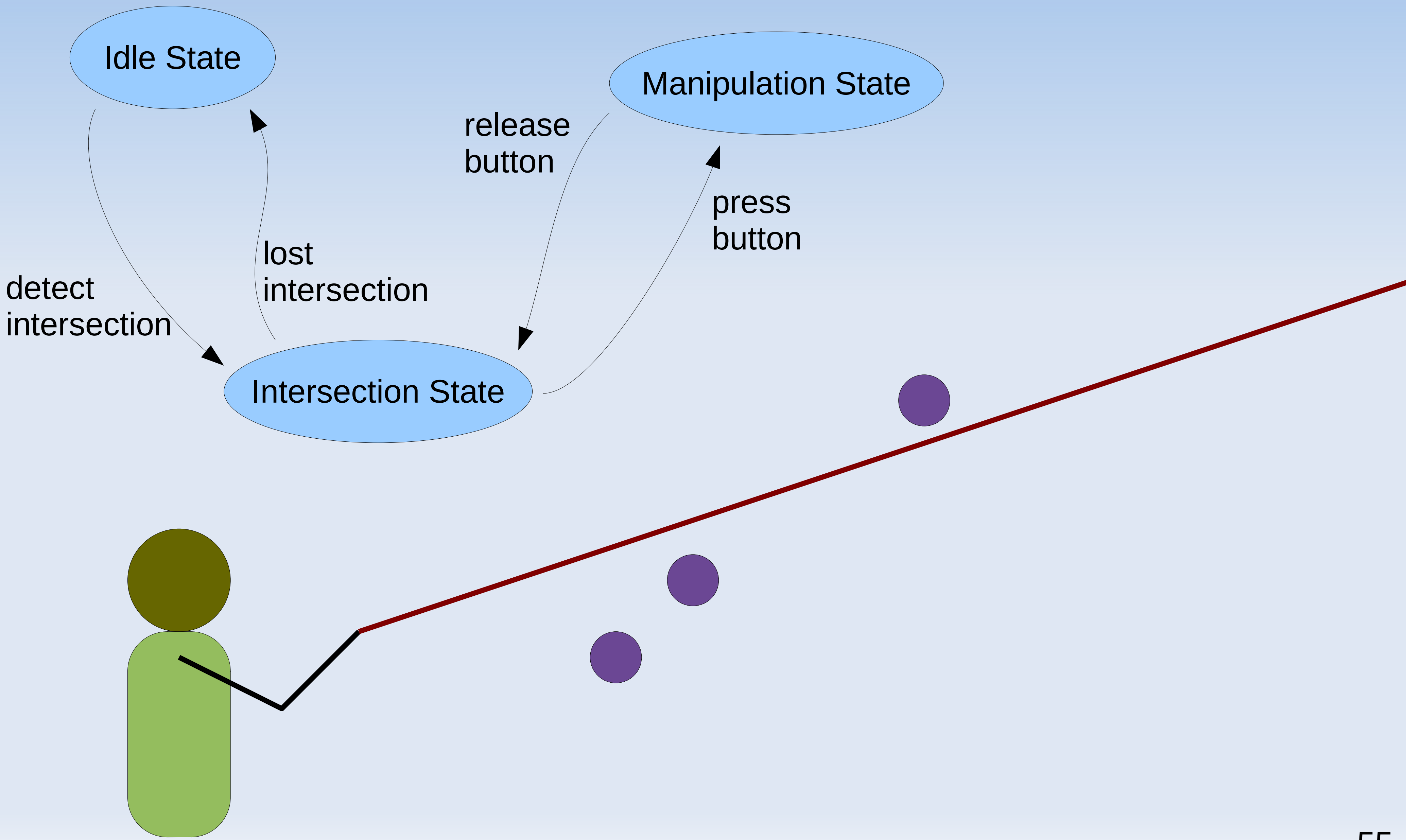


Multi-touch in 3D

- Multi point interaction extends into 3D
  - One point = 3 DoF
  - Two points = 5 DoF

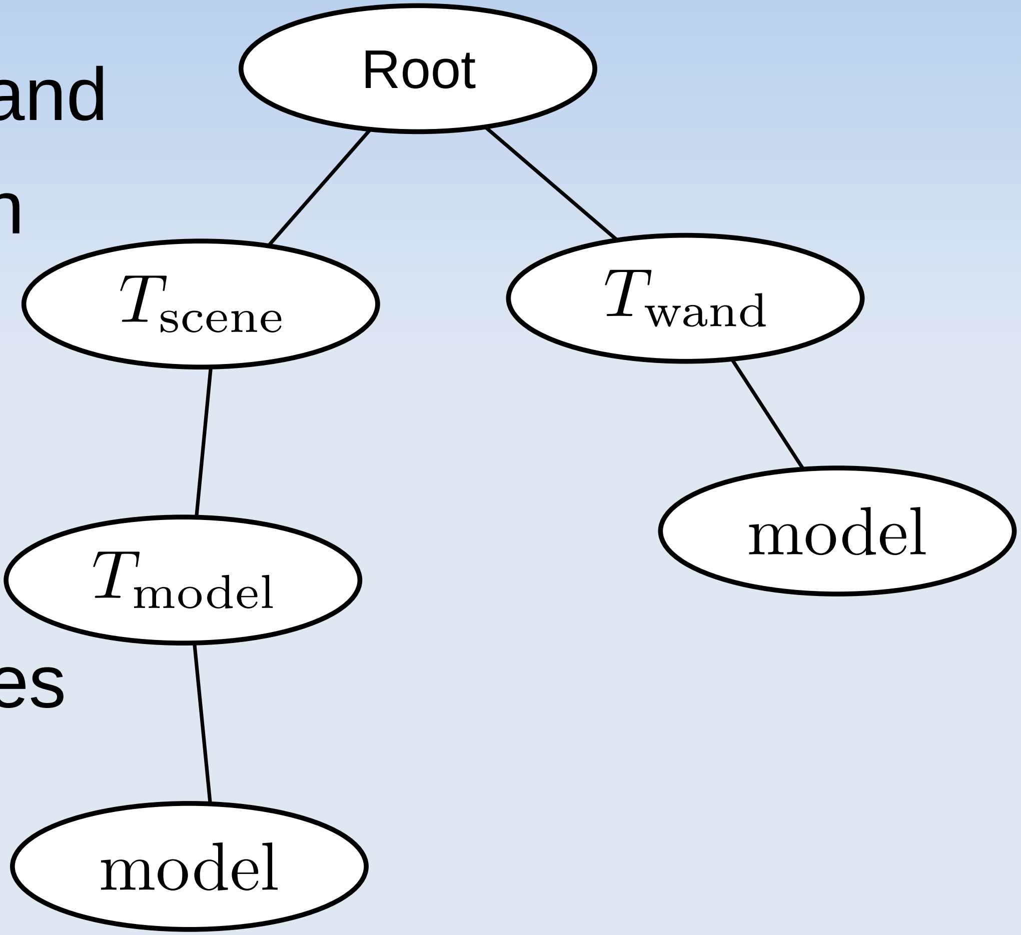


Grab States

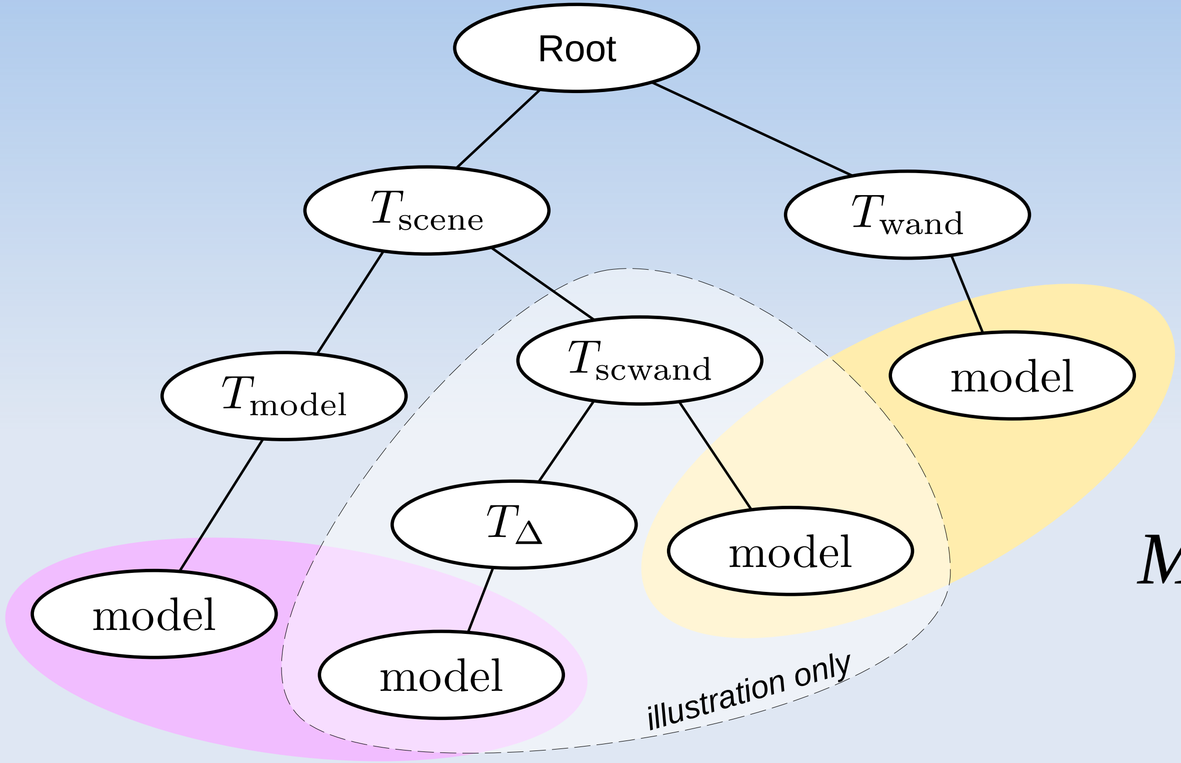

Idle State
Manipulation State
release button
press button
lost intersection
detect intersection
Intersection State

Grab Mathematics

- Manipulating model transform
  - Lock the model to the wand
  - 6 DoF pose mainpulation
  - Simple 3D algebra
- Prerequisites
  - Navigated scene
  - Wand in room coordinates



Grab Mathematics



$$M_{wand} = M_{scene} M_{scwand}$$

$$M_{scene}^{-1} M_{wand} = M_{scwand}$$

$$M_{scwand} M_{\Delta} = M_{model}$$

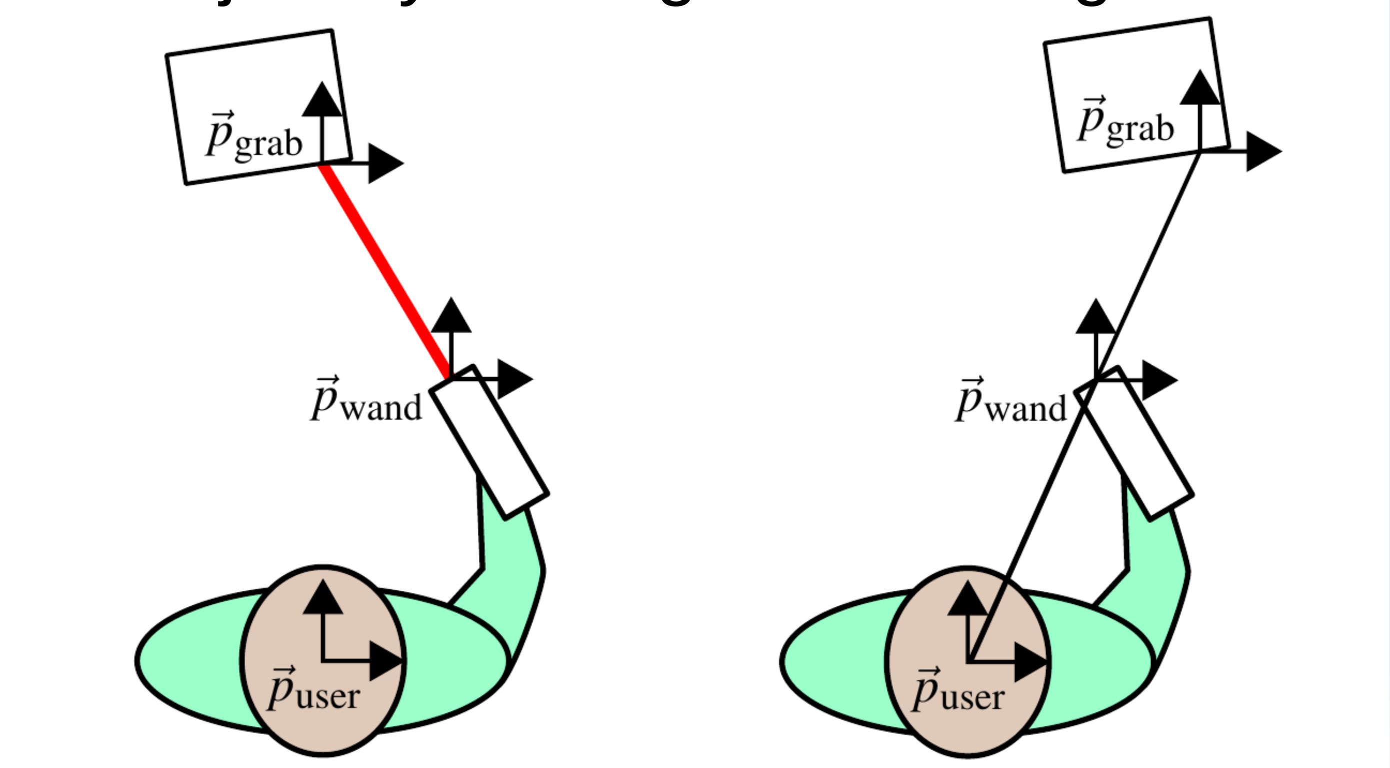
$$M_{\Delta} = M_{scwand}^{-1} M_{model}$$

Initialize upon button press: $M_{\Delta} = M_{scwand}^{-1} M_{model}$

Update during button press: $M_{scwand} M_{\Delta} = M_{model}$

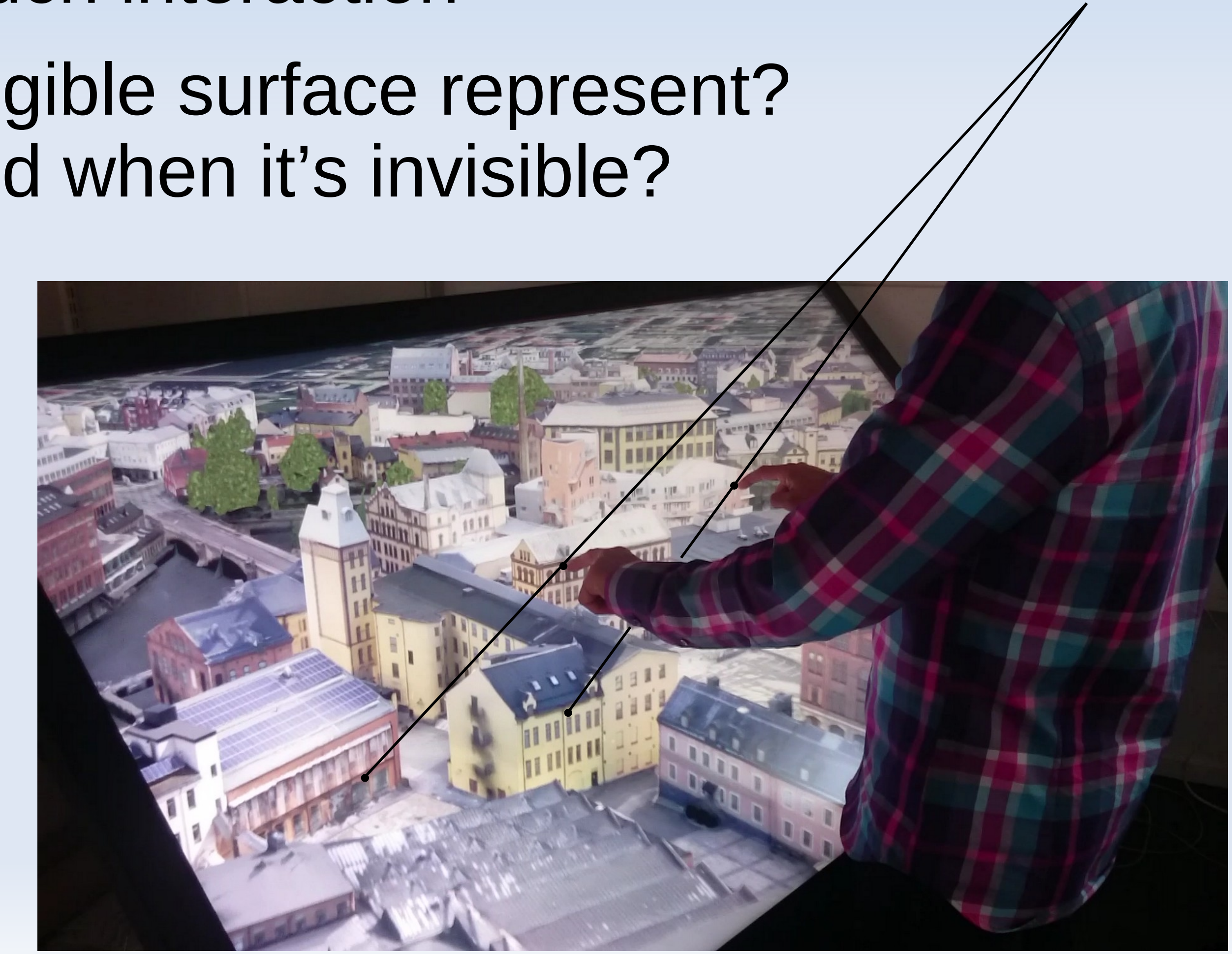
HOMER

- Hand-centered Object Manipulation Extending Ray-casting (1997)
  - Grab at a distance with pointer metaphor
  - Object locks to user-wand line
  - Move and rotate object by moving and rotating wand



Screen Space Interaction

- Use 2D interaction in 3D
  - Any 2D point is also a 3D line of sight
  - Allows for 2,5D touch interaction
  - What does the tangible surface represent? How it is perceived when it's invisible?



Interaction Summary

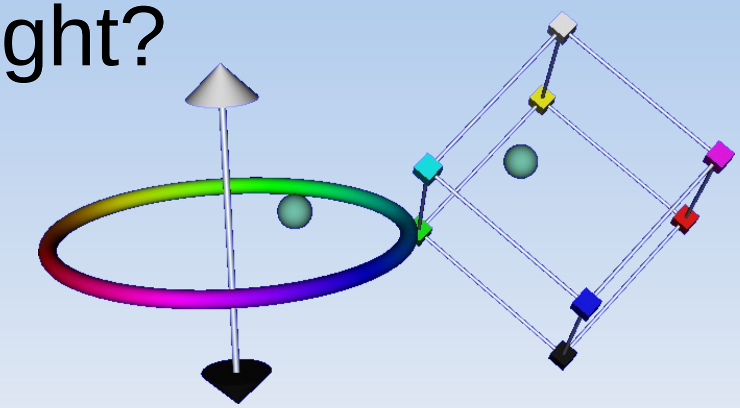
- Many available methods and metaphors
  - We have just touched a few alternatives
- Best practice depends on situation
  - Size of movements necessary
  - Size of workspace
  - Size of environment
- Fantasy sets the limits, maths is your tool!
  - Many basic metaphors
  - Don't complicate the interaction
    - make an inuitive action-reaction connection

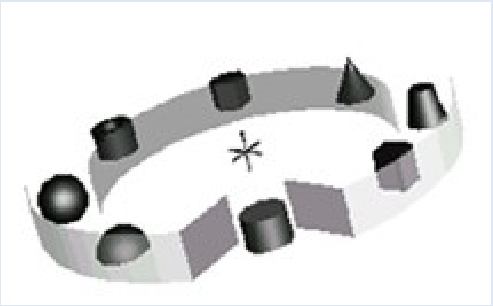
Controls

- Physical
  - In Spatial VR (RMD)
    - Simulate real situation
  - In HMD – Covered vision
    - Make them realistic, part of VR
    - Hard to interact with if you cannot see your hands
- Virtual
  - Make navigation controls natural part of VR
  - Non-haptic devices are hard to interact with

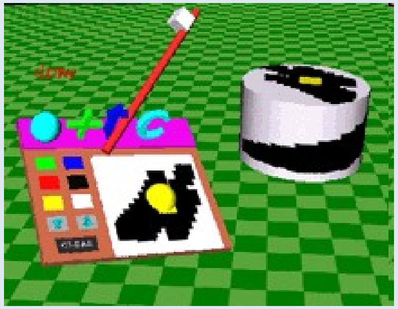
Virtual Widgets

- 3D world – more possibilities, right?
  - Any object can become a widget
  - Our interaction is typically 2D
    - Widgets typically become panels
    - Few examples of 3D shaped widgets

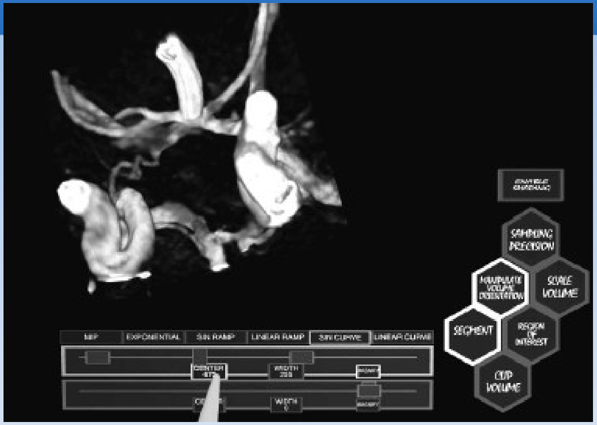




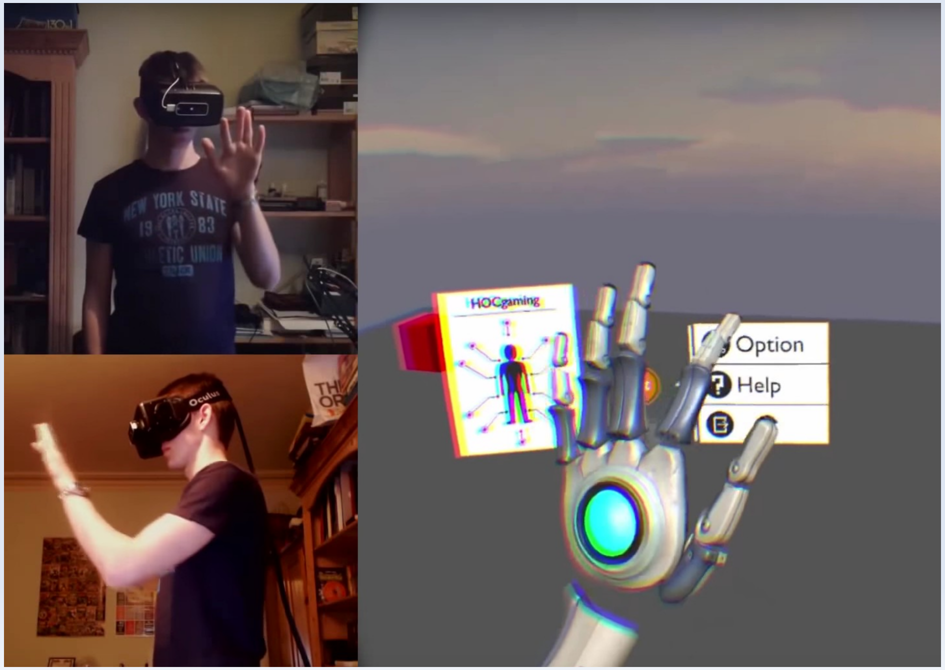
- Interaction in 3D
  - More space for many, many widgets
  - Natural (and fast) interaction with widgets
    - Gestures (Johnny Mnemonic?)
    - Picking, pointing, etc
    - Virtual controls can be hard to interact with



Potential Issues with Widgets



- Occlusion
  - Large displayed widgets block display
    - Small widgets have resolution problems
  - Context-dependent pop-ups useful
- 3D text is hard to render well
  - (no pixel alignment)
- Distance problems
  - Accomodation discrepancy
- Interaction
  - No physical feedback



Summary

- Navigation and interaction in VR is tricky
  - Many possibilities, but few good guidelines
  - Solve the problem in real 3D coords
- Lots of potential problems
  - Intuitivity, precision, fatigue
  - Complex navigation points
- Important to analyse application
  - Navigational requirements
  - Interaction requirements
  - Features in the application display