Module IN 2018

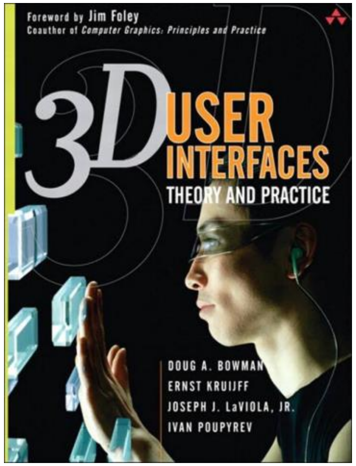
3D User Interfaces
- Dreidimensionale Nutzerschnittstellen -

Prof. Gudrun Klinker

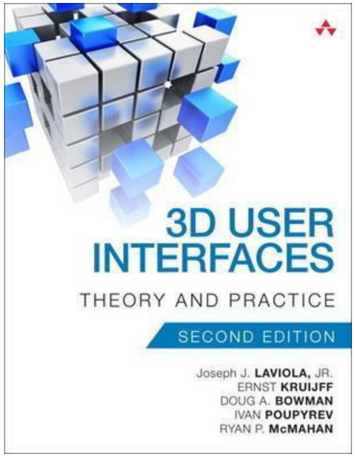
Selection and Manipulation
SS 2023

Literature

- *3D User Interfaces – Theory and Practice. 2nd edition*
  D. Bowman, E. Kruiff, J. LaViola, I. Poupyrev.
  Addison-Wesley Pearson Education, 2005.



- *3D User Interfaces – Theory and Practice. 2nd edition*
  J. LaViola, E. Kruiff, D. Bowman, I. Poupyrev, R. McMahan.
  Addison-Wesley Pearson Education, 2017.
  https://www.pearson.com/us/higher-education/program/La-Viola-3-D-User-Interfaces-Theory-and-Practice-2nd-Edition/PGM101825.html



Agenda

1. 3D Selection and Manipulation Tasks
2. Interaction Techniques and Input Devices
3. Interaction Techniques for 3D Manipulation
4. Design Guidelines

1. 3D Selection and Manipulation Tasks

→ 1.1 Canonical Selection and Manipulation Tasks

1.2 Application-Specific Selection and Manipulation Tasks

1.1 Canonical Selection and Manipulation Tasks

- Parameters associated with specific manipulation tasks
  - Application goals
  - Object sizes
  - Object shapes
  - Distance from objects to user
  - Characteristics of the physical environment
  - Physical and psychological states of the user

- Try to find repeating „basic“ tasks across many applications

1.1 Canonical Selection and Manipulation Tasks

Basic manipulation tasks:

- Selection
  (Real world: Picking an object with a hand)
- Positioning
  (Real world: Moving an object from A to B)
- Rotation
  (Real world: Rotating an object from one pose into another)
- (Scaling)

1.1 Canonical Selection and Manipulation Tasks

Parameter spaces (of individual tasks)

- Task space
- Each task parameter defines a design dimension

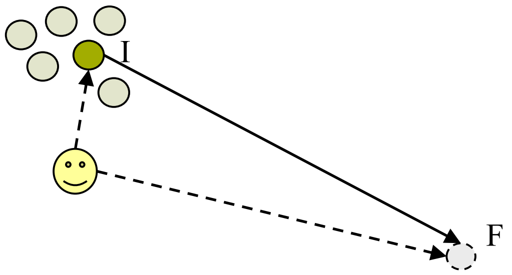
Parameters associated with specific manipulation tasks

- *Application goals*
- *Object sizes*
- *Object shapes*
- *Distance from objects to user*
- *Characteristics of the physical environment*
- *Physical and psychological states of the user*

1.1 Canonical Selection and Manipulation Tasks

Parameter spaces for **basic, recurring tasks**

- Selection
  - Distance, direction to target
  - Target size
  - 密度 Density of objects around target
  - Number of targets to be selected
  - Target occlusion
- Positioning (Google Map)
  - Distance, direction to initial position (I)
  - Distance, direction to final position (F)
  - Translation distance
  - Required precision of positioning 精度
- Rotation
  - Distance to target
  - Initial orientation
  - Final orientation
  - Amount of rotation
  - Required precision of rotation



1. 3D Selection and Manipulation Tasks

1.1 Canonical Selection and Manipulation Tasks

→ 1.2 Application-Specific Selection and Manipulation Tasks

1.2 Application-Specific Selection and Manipulation Tasks

- *Parameters associated with specific manipulation tasks*
  - *Application goals*
  - *Object sizes*
  - *Object shapes*
  - *Distance from objects to user*
  - *Characteristics of the physical environment*
  - *Physical and psychological states of the user*
- Select a representative subset of requirements for a specific task, e.g. for
  - VR medical training: positioning a medical probe in virtual patient models
  - Flight simulator: landing a plane

Agenda

1. 3D Selection and Manipulation Tasks
2. Interaction Techniques and Input Devices
3. Interaction Techniques for 3D Manipulation
4. Design Guidelines

重心：Input.

2. Interaction Techniques and Input Devices

→ 2.1 Input-Control-Mapping

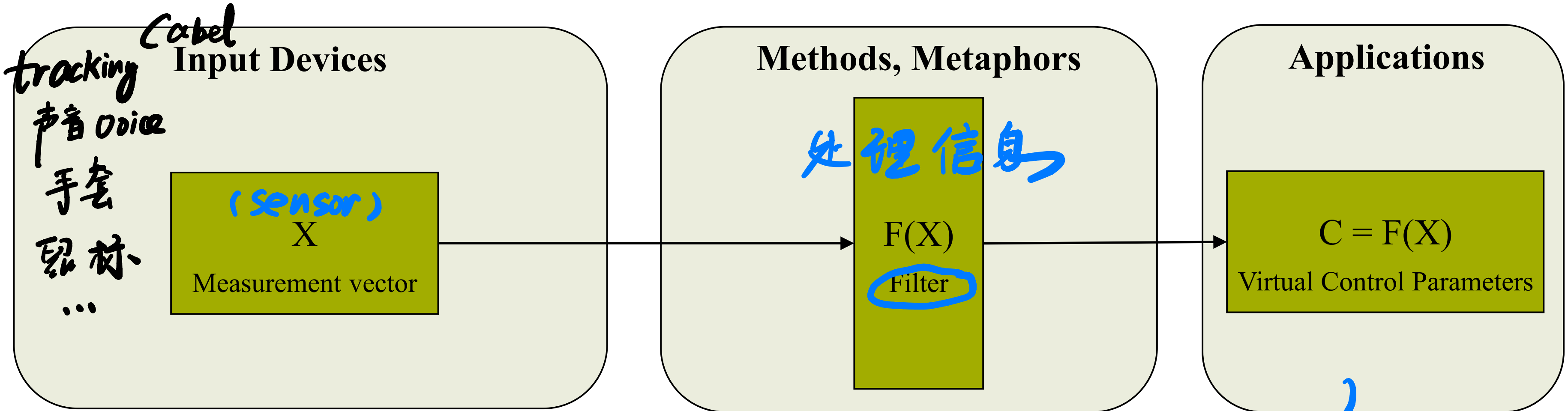
2.2 Control Dimensions and Integrated Control in 3D Manipulation

2.3 Force versus Position Control

Remarks:

- Here: restricted to input devices
- More on this also in a subsequent class on input devices

2.1 Input-Control-Mapping

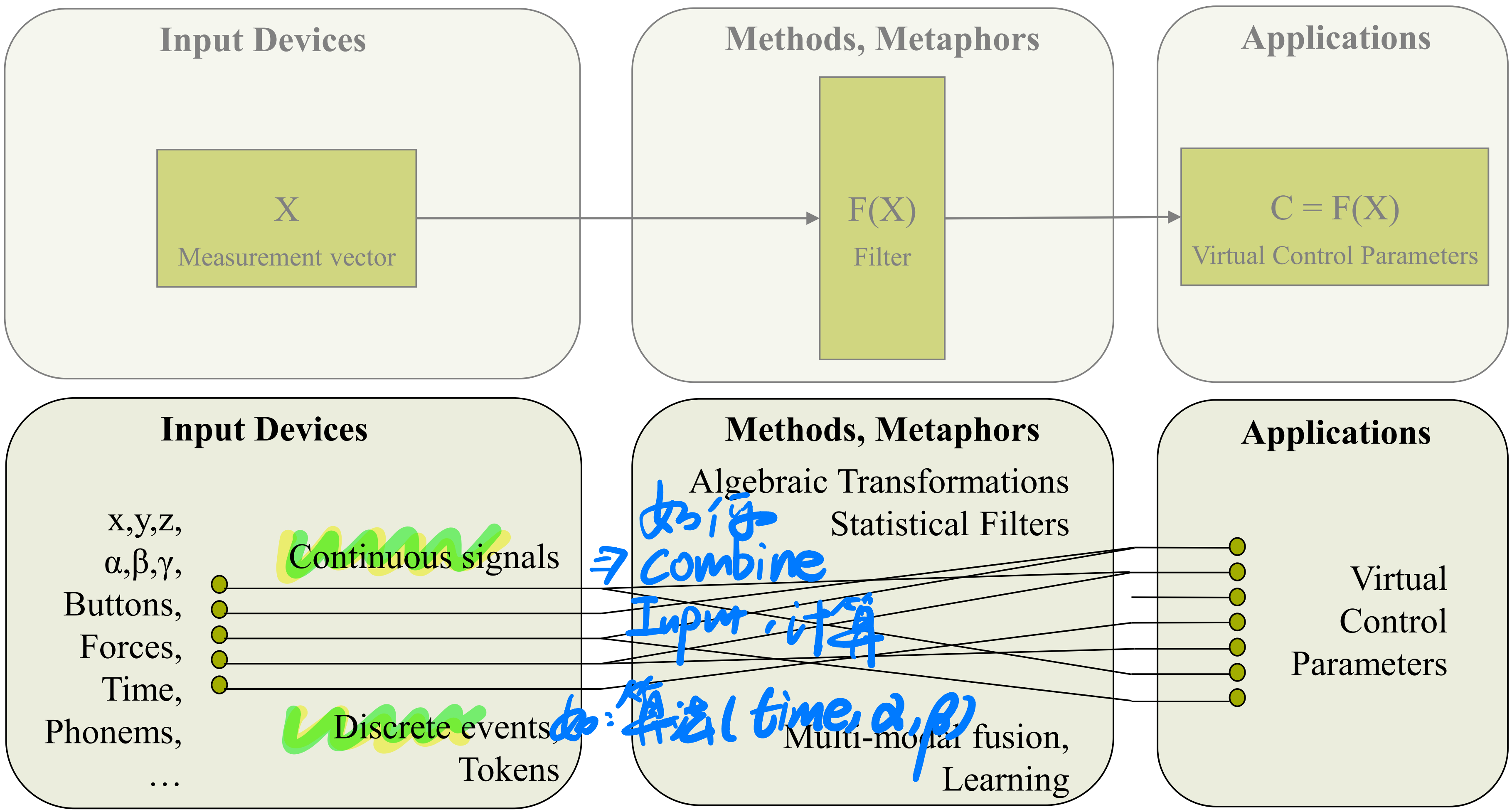


Input-Control Mapping

- Object selection, picking
- Object manipulation
- Travel, camera control

基本 action 的 mapping

2.1 Input-Control-Mapping



2. Interaction Techniques and Input Devices

2.1 Input-Control-Mapping

→ 2.2 Control Dimensions and Integrated Control in 3D Manipulation

2.3 Force versus Position Control

2.2 Control Dimensions and Integrated Control

- Key parameters
  - Number of control dimensions (DOFs) Degrees of Freedom
  - Level of integration (simultaneous control of DOFs)
- Examples
  - Desktop mouse: 2 DOFs
  - Trackers (magnetic, optical): ≤ 6 DOFs
  - Game controllers: ≥ 4 DOFs (2+2 for each hand)
- Issues
  - Cost
  - Availability
  - Ease of maintenance
  - Targeted user population

2. Interaction Techniques and Input Devices

2.1 Input-Control-Mapping

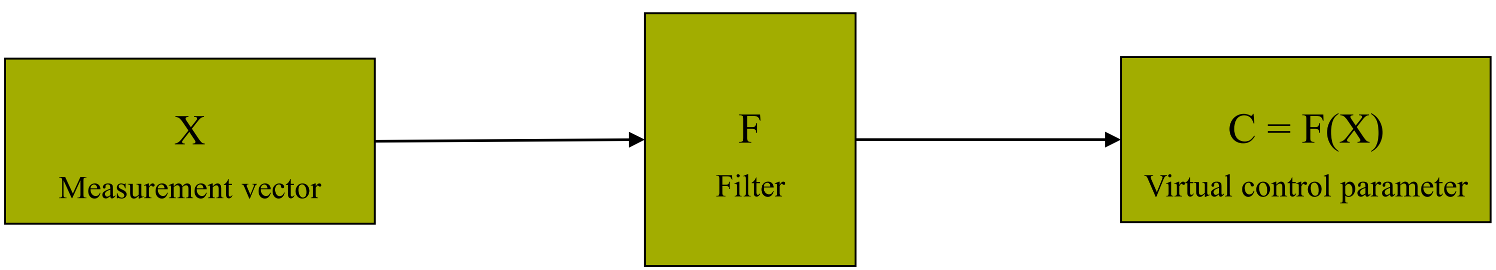
2.2 Control Dimensions and Integrated Control in 3D Manipulation

→ 2.3 Force versus Position Control

2.3 Force versus Position Control

- *Isomorphic control*:
  - Absolute or relative **position** of the human hand (mouse, trackers, etc)
- *Non-isomorphic (*either *isometric* or *isotonic) control*:
  - 力 Force applied to a device, speed of motion (**rate** of position changes) (joy stick)
    速度
  - …

- For 6-DOF manipulation tasks: Position control usually better than force control [Zhai and Milgram 93]
- Force control better at controlling rates (speed of navigation).



Agenda

1. 3D Selection and Manipulation Tasks
2. Interaction Techniques and Input Devices
3. Interaction Techniques for 3D Manipulation
4. Design Guidelines

3. Interaction Techniques for 3D Selection and Manipulation

- 3.1 Classifications of Selection and Manipulation Techniques
- 3.2 Interacting by Pointing
- 3.3 Direct Manipulation: Virtual Hand Techniques
- 3.4 World-in-Miniature
- 3.5 Combining Techniques
- 3.6 Nonisomorphic 3D Rotation
- 3.7 Desktop 3D Manipulation

3.1 Classifications

Isomorphism in selection and manipulation techniques

- *Isomorphic* view → 比例调整x、y、z 对人而言自然
  - Strict, geometrical 1-to-1 mapping (one to one mapping)
  - Pro: Very natural
  - Con:
    - Impractical due to constraints on input devices
    - Impractical due to human limitations

    stuck in physical world
- *Non-isomorphic* view:
  - „Better reality“, specifically taylored to 3D environments
  - „Magic“ virtual tools (laser beams, rubber arms, voodoo dolls)

↓

非 geometrical 的 1 to 1

3.1 Classifications

Classification

- By task decomposition
- By metaphor

Input

3.1 Classifications

按一个事件描述Input方式

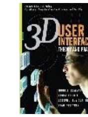
By **task decomposition** 整个Input过程

- All selection and manipulation techniques consist of the same basic components that serve similar purposes
- Interaction „building blocks“:

Example:

- selection technique
  - indication of object 选
    - occlusion
    - object touching
      - list
      - voice sel.
      - automatic
      - iconic objects
    - 所有可能
    - pointing
      - 2D
      - 3D gaze
      - 3D hand
    - indirect selection
      - 1 to 1
      - pos → pos
      - vel → pos
      - pos → vel
  - confirmation of selection 确定吗?
    - event
    - gesture
    - voice command
    - no explicit command
  - feedback
    - text/symbolic
    - aural
    - visual
    - force/tactile

Input程序结果



3.1 Classifications 更自然的学习法

By **metaphor**: Fundamental mental model of a technique

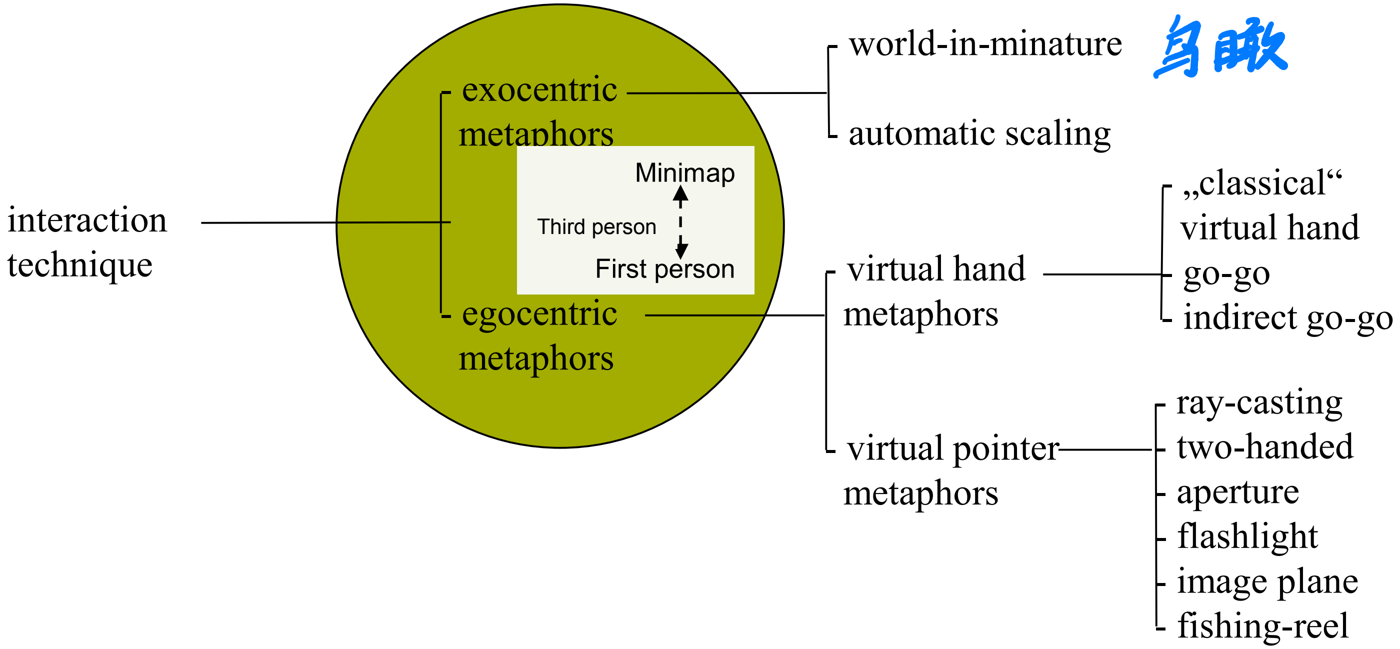
- *Affordances*: what users can do by using a technique
- *Constraints*: what users cannot do with a technique

- interaction technique
  - 向外 exocentric metaphors
    - world-in-minature
    - automatic scaling
  - 中心主义 egocentric metaphors
    - virtual hand metaphors
      - „classical“ virtual hand
      - go-go
      - indirect go-go
    - virtual pointer metaphors
      - ray-casting
      - two-handed
      - aperture
      - flashlight
      - image plane
      - fishing-reel

3.1 Classifications

By metaphor

- Exocentric metaphors: bird‘s eye view
- Egocentric metaphors: through the user‘s eyes



3. Interaction Techniques for 3D Selection and Manipulation

3.1 Classifications of Selection and Manipulation Techniques

→ 3.2 Interacting by Pointing

3.3 Direct Manipulation: Virtual Hand Techniques

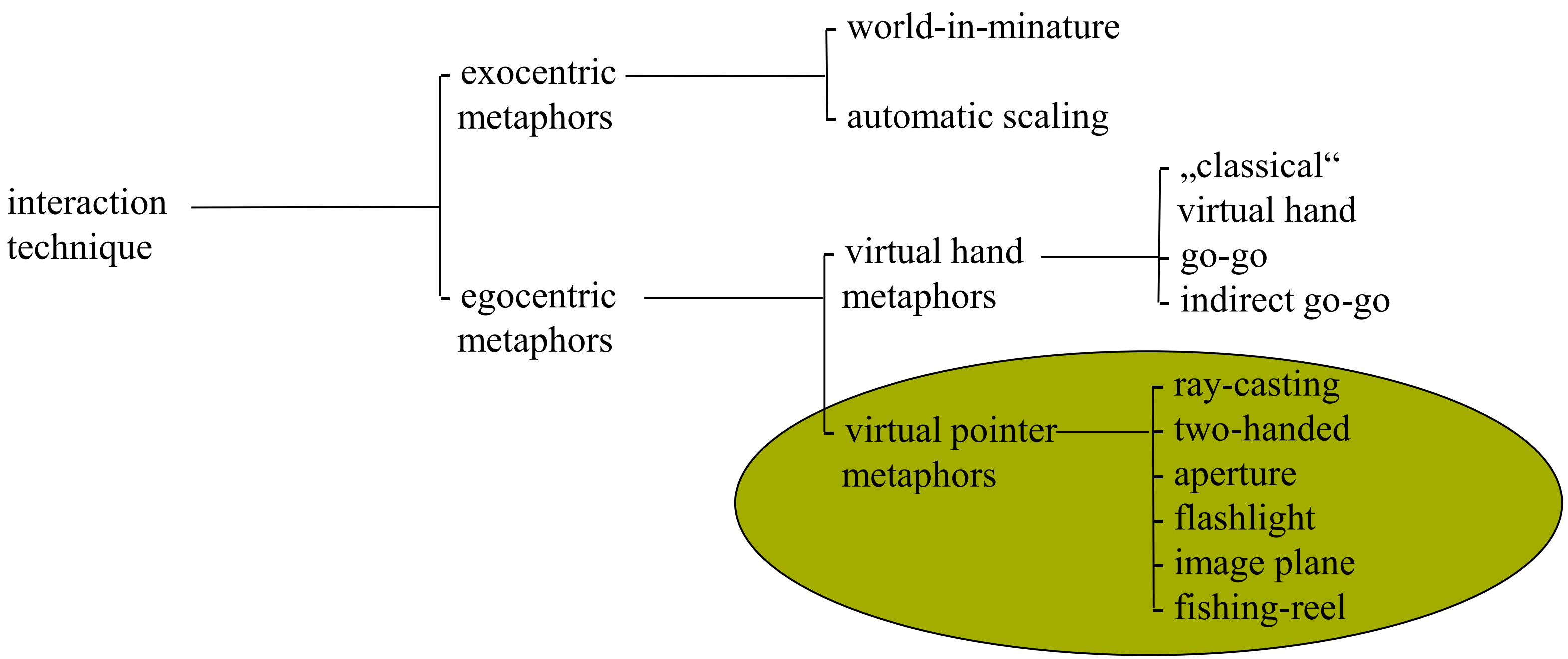
3.4 World-in-Miniature

3.5 Combining Techniques

3.6 Nonisomorphic 3D Rotation

3.7 Desktop 3D Manipulation

3.2 Interacting by Pointing



3.2 Interacting by Pointing

- Goal:
  - Select and manipulate objects **beyond** area of reach
- Sequence of actions:
  - Pointing direction = 3D vector
  - Object selection = intersection (vector, virtual object)
  - Confirmation = triggering event
- Further steps:
  - Attach selected object to end of vector (for manipulation)
- Pro:
  - Better selection performance than hand-based techniques (little physical hand movement)
- Con:
  - Very poor positioning technique (only radial movements around the user)
  - Rotations only around one axis (pointing direction)

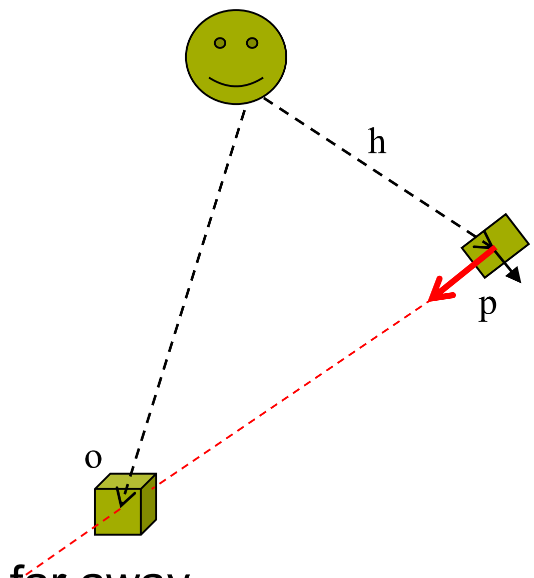
3.2 Interacting by Pointing

→ 3.2.1 Ray-casting
3.2.2 Two-handed pointing
3.2.3 Flashlight and aperture techniques
3.2.4 Image plane techniques
3.2.5 Fishing-reel techniques

3.2.1 Ray-Casting

- Hand position **h** and direction **p**
  - VEs: 6 DOF tracker
  - Desktop environment: 3D widget (WIMP)
- Object position **o**(α) = **h** + α**p**

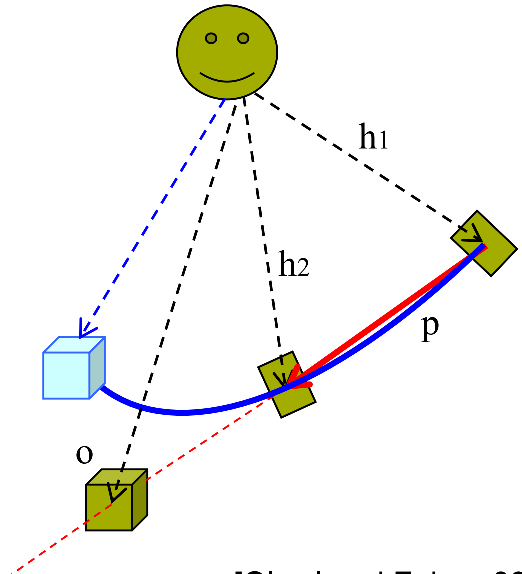
- Shape of ray
  - Short line segment (fixed length)
  - Infinite ray



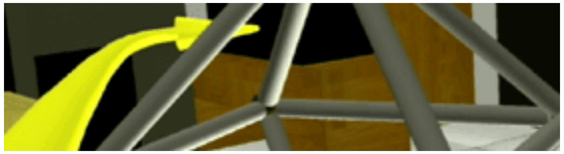
- Powerful technique, if object is not too small or too far away

3.2.2 Two-Handed Pointing

- Hand positions $\mathbf{h_1}$, $\mathbf{h_2}$
- Ray direction $\mathbf{p} = \mathbf{h_2} - \mathbf{h_1}$
- Object position $\mathbf{o}(\alpha) = \mathbf{h_1} + \alpha(\mathbf{h_2} - \mathbf{h_1})$



[Olwal and Feiner 03]

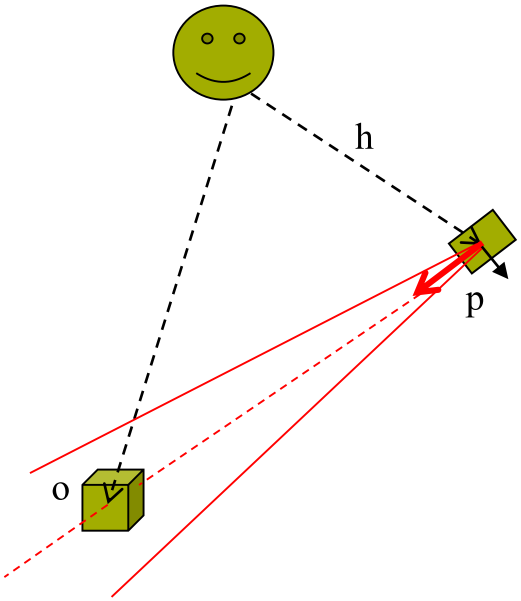


- Additional information:
  - Distance between hands = scale factor
  - Hand orientations: non-linear curve parameters
- Possible to disambiguate between several objects on ray

3.2.3 Flashlight and Aperture Techniques

Flashlight

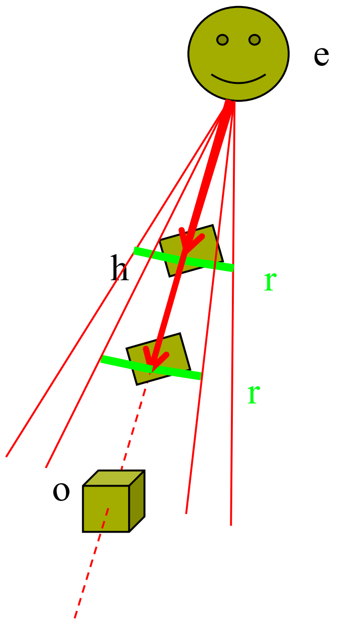
- „Soft“ selection technique
- Requires less precision of pointing
- Disambiguation between several objects inside the cone
  - Object closest to center line
  - Object closest to the device



3.2.3 Flashlight and Aperture Techniques

Aperture technique

- Interactive control of the opening angle of the cone
- Cone defined by
  - Apex position = eye position **e**
  - Cone orientation = hand-eye position **h-e**
  - Opening angle = acos (r / ||**h-e**||)
- Object position **o**(α) = **e** + α(**h** – **e**)



3.2.4 Image-Plane Techniques

- Select and manipulate 3D objects by touching and manipulating their 2D projections on a virtual plane located in front of the user
  - Sticky-finger (one finger or stylus)
  - Head-crusher (two fingers: data glove)

- Simulate direct touch
- Intuitive and easy to use for selection

3.2.5 Fishing-Reel Techniques

- Use additional input device to control the distance to virtual objects (length of virtual ray)
  - Select an object with ray casting
  - Reel it back and forth (via an additional input device)

- Separation of control between several devices
- Reduces user performance

Overview

3. Interaction Techniques for 3D Selection and Manipulation

3.1 Classifications of Selection and Manipulation Techniques

3.2 Interacting by Pointing

→ 3.3 Direct Manipulation: Virtual Hand Techniques

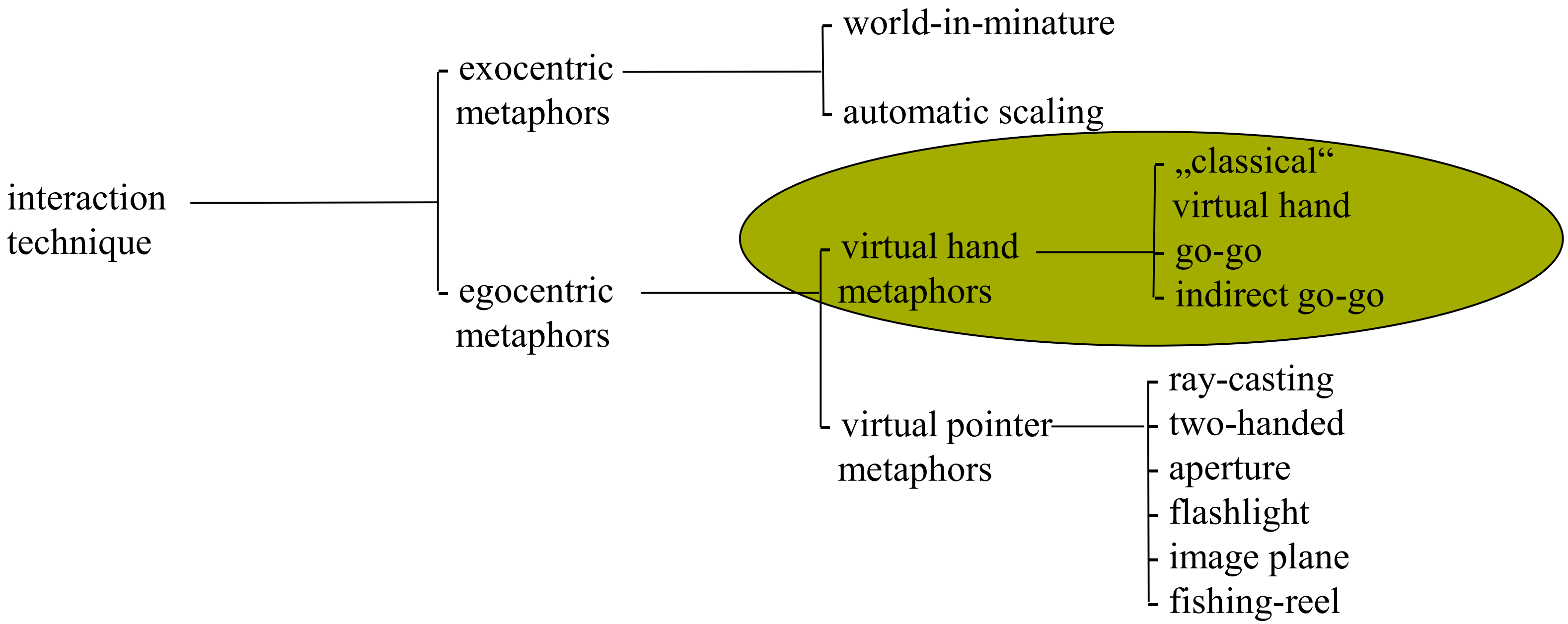
3.4 World-in-Miniature

3.5 Combining Techniques

3.6 Nonisomorphic 3D Rotation

3.7 Desktop 3D Manipulation

3.3 Virtual Hand Techniques (Isomorphic)



3.3 Virtual Hand Techniques (Isomorphic)

- Goal
  - Directly select and manipulate objects with hands
- Approaches
  - 3D cursor (e.g.: a hand icon) represents current position and orientation of user‘s hand in VE
  - Semitransparent volumetric cursors provide additional depth cue
  - Object selection =
    - Intersection of 3D cursor with virtual object
    - Triggering event (button, voice, hand gesture)
    - Object is then attached to virtual hand, can be manipulated
    - Object release via another triggering event
- Pro: Better **manipulation** performance than pointer-based techniques
    - Full 3D positioning
    - Full 3D rotations
- Con: Worse **selection** performance than pointer-based techniques
    - More physical hand movement
    - Limited reaching range

3.3 Virtual Hand Techniques

→ 3.3.1 Hand-based grasping

3.3.2 Finger-based grasping

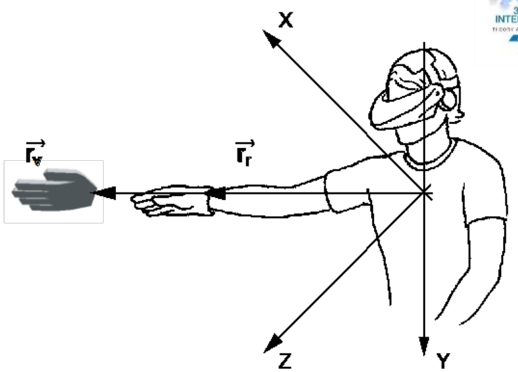
3.3.3 Enhancements for grasping metaphors

3.3.1 Hand-based Grasping

Simple virtual hand

- Direct mapping: user‘s hand motion (p_r,R_r) → virtual hand motion (p_v,R_v)
  $$p_v = \alpha p_r$$
  $$R_v = R_r$$
- *„Transfer functions“* (*„control-display gain functions“*)
- *Zero-order mappings*:
  displacement of input device = displacement of controlled element (linear mapping)
- *First-order mappings*:
  displacement of input device = change of velocity
- Pro: Intuitive due to directly simulated interaction with real objects
- Con: Only objects within the user‘s reach can be selected and manipulated

3.3.1 Hand-based Grasping

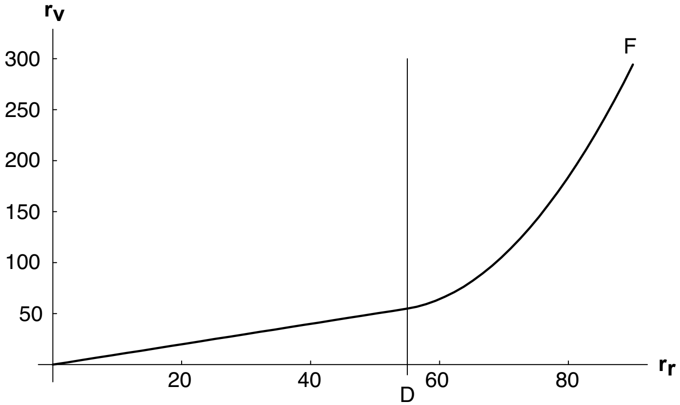


[Poupyrev et al. 96]

Go-go technique [Poupyrev et al. 1996]

- Interactively change the length of the virtual arm
- In user-centered egocentric coordinate system virtual distance r_v is non-linear function of real distance r_r

$$r_v = F(r_r) = \begin{cases} r_r & if\ r_r \le D \\ r_r + \alpha(r_r - D)^2 & otherwise \end{cases}$$



[Poupyrev et al. 96]

- Virtual hand shown at extended distance
- (For verification: small cube shown at measured distance)
- Direct, seamless 6-DOF object manipulation both at close range and at large distances
- Studies: users did not have difficulties understanding the technique

3.3 Virtual Hand Techniques

3.3.1 Hand-based grasping

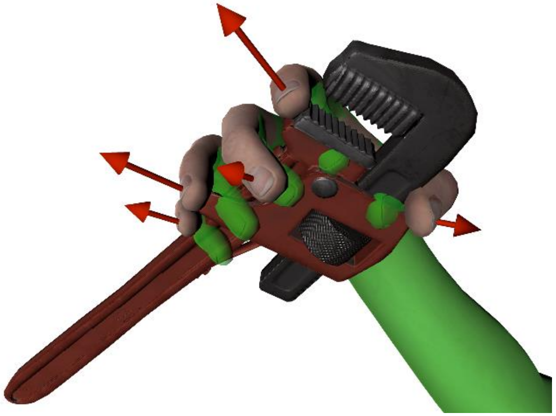
→ 3.3.2 Finger-based grasping

3.3.3 Enhancements for grasping metaphors

3.3.2 Finger-based Grasping

Different physical approaches to simulate the fingers and interactions

- Rigid-body fingers
  - Bend-sensing gloves
  - System of virtual torsional and linear spring-dampers
  - Dynamic influence on the mappings between the user‘s real hand („tracked hand“) and the virtual hand („spring hand“)
  - When the tracked hand collides with / penetrates the inner space of a virtual object, the spring dynamically prevents visual penetration
  - Produces forces and torques that enable direct interactions with the virtual object
  - Difficult to release an object due to clumsy spring-damping coefficients („sticky“)



[Borst and Indugula 05]

3.3.2 Finger-based Grasping

Different physical approaches to simulate the fingers and interactions

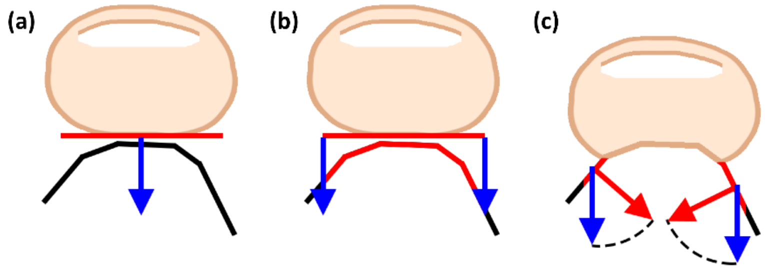
- Soft-body fingers
  - Deformable (soft-body) representations of
  - virtual fingers
  - Lattice shape-matching to dynamically adapt to the shapes of the grasped virtual objects (FastLSM algorithm)
  - Upon collision, virtual fingers deform slightly, resulting in a few points of collision
  - Upon penetration, deformation increases resulting in many points of collision

[Jacobs and Fröhlich 2011]

3.3.2 Finger-based Grasping

Different physical approaches to simulate the fingers and interactions

- God fingers
  - God object: Virtual point that adheres to rigid-body physics
  - Never penetrates virtual objects, but remains at their surfaces
  - Attach god objects to tracked fingers
  - Calculate direction of force upon penetration



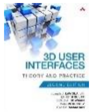
[Talvas et al 13]

3.3 Virtual Hand Techniques

3.3.1 Hand-based grasping

3.3.2 Finger-based grasping

→ 3.3.3 Enhancements for grasping metaphors



3.3.3 Enhancements for Grasping Metaphors

- 3D bubble cursor [Grossmann & Balakrishnan 2005]
  - Area cursor that dynamically changes its radius to always touch the closest object
- PRISM [Frees & Kessler 2007]
  - General enhancement for grasping metaphors, using scaled-down motion to the user‘s virtual hand: no motion, scaled motion, 1:1 motion, offset recovery
- Hook (for moving targets)
  - Make the user follow a target in an attempt to overtake it
  - Offer several targets, see which one the user chooses
- Intent-driven selection
  - Use the posture of the virtual fingers as an indication of the user‘s level of confidence in selecting an object.
  - Proximity sphere

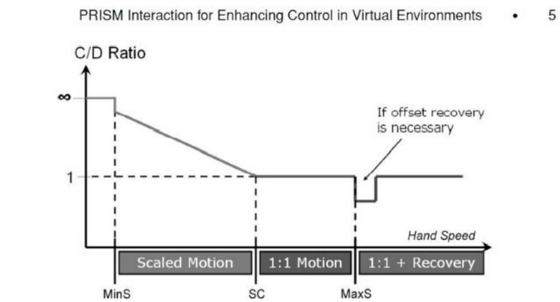
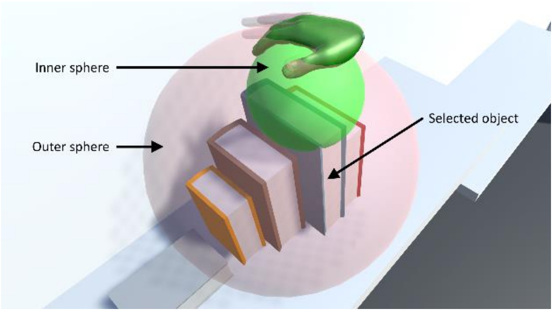


Fig. 1. Simplified interface diagram showing how PRISM uses Hand Speed to adjust CD.



[Periverzov and Ilies 15]

3. Interaction Techniques for 3D Selection and Manipulation

3.1 Classifications of Selection and Manipulation Techniques

3.2 Interacting by Pointing

3.3 Direct Manipulation: Virtual Hand Techniques

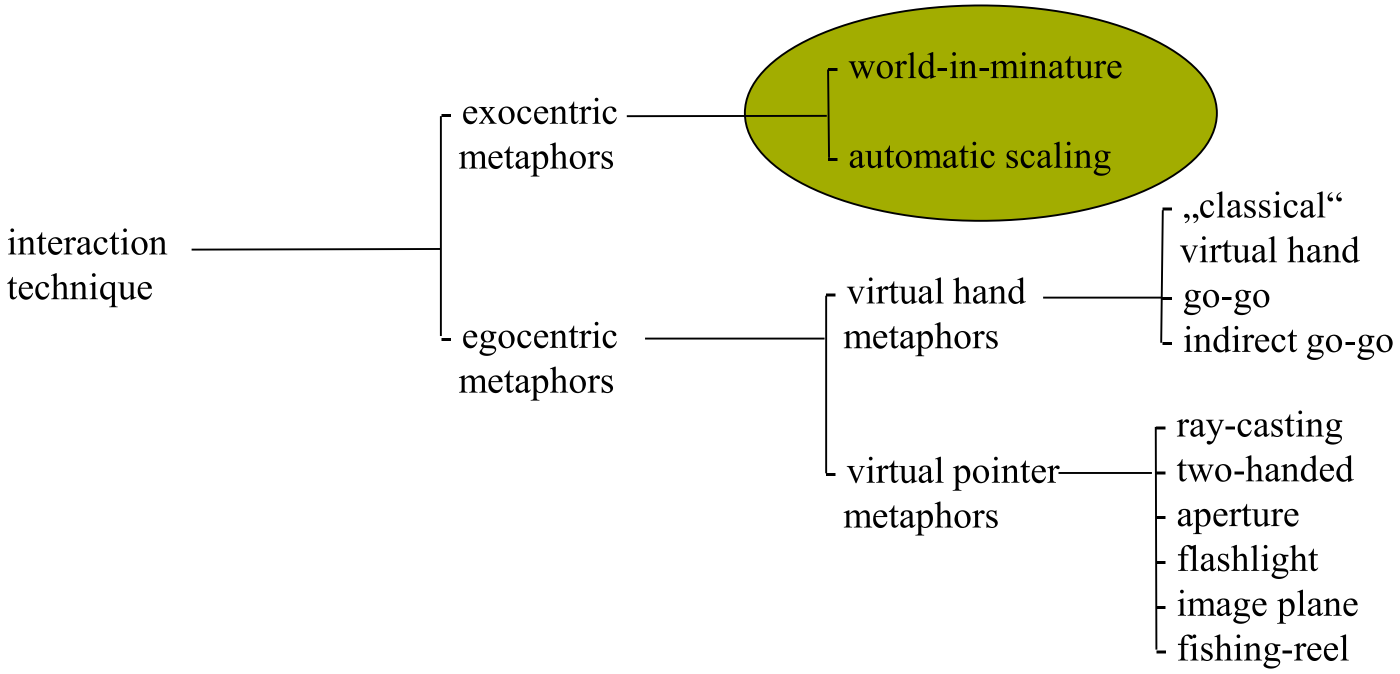
→ 3.4 World-in-Miniature

3.5 Combining Techniques

3.6 Nonisomorphic 3D Rotation

3.7 Desktop 3D Manipulation

3.4 World-in-Miniature (WIM)



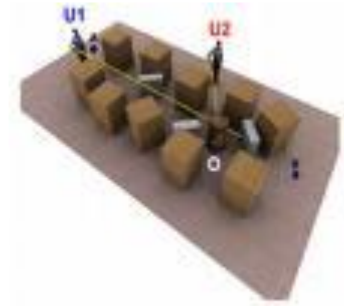
3.4 World-in-Miniature (WIM)

- Miniature (down-scaled) handheld model of the VE
- Seen from an exocentric (bird‘s eye) perspective

- Indirect manipulation of large-scale virtual objects by manipulating their down-scaled copies in WIM

- Pro
  - Easy object manipulation within and outside the user‘s reach
  - Combination of manipulation and navigation
- Con
  - Not easily scalable

- Well suited for augmented reality
- Generalization of overview maps in games



[Pinho et al 08] www.scielo.br

3. Interaction Techniques for 3D Selection and Manipulation

3.1 Classifications of Selection and Manipulation Techniques

3.2 Interacting by Pointing

3.3 Direct Manipulation: Virtual Hand Techniques

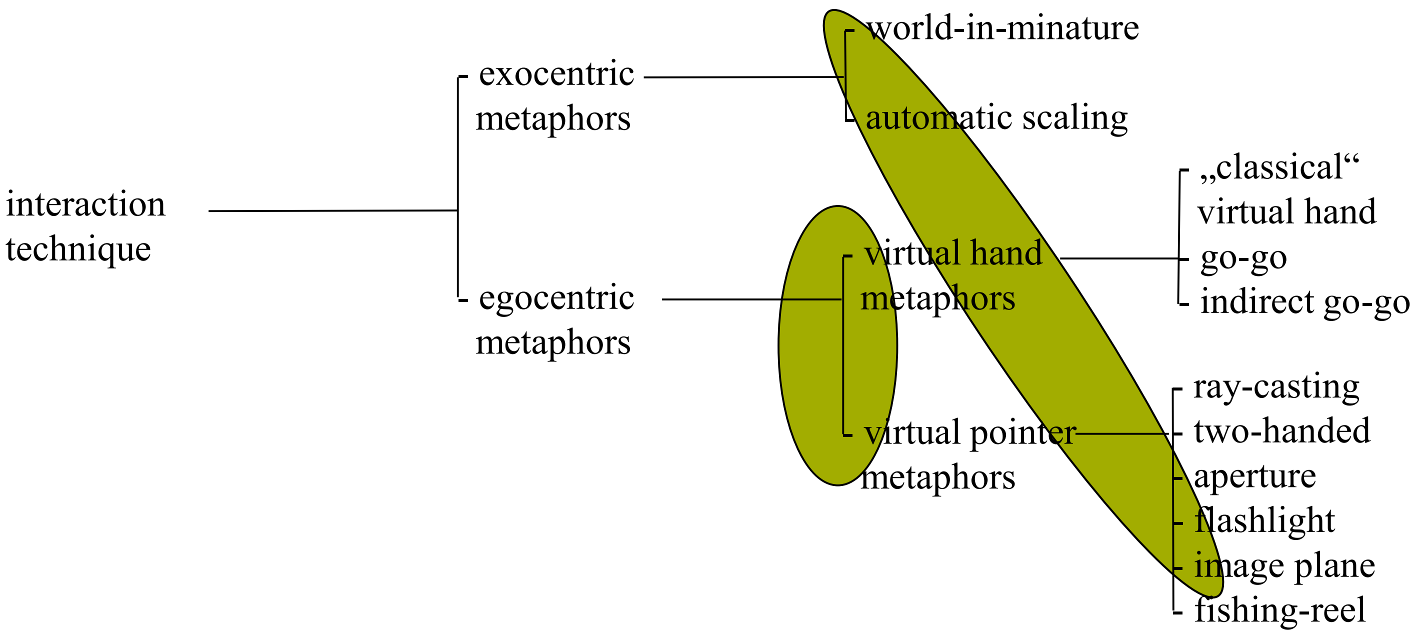
3.4 World-in-Miniature

→ 3.5 Combining Techniques

3.6 Nonisomorphic 3D Rotation

3.7 Desktop 3D Manipulation

3.5 Combining Techniques



3.5 Combining Techniques

- Aggregation of techniques
  - Users have to select from a „bag of tricks“
- Integration of techniques
  - Interface switches transparently between interaction techniques depending on the *task context*
  - Task sequence
    - Selection
    - Manipulation

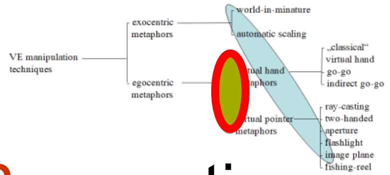
3.5 Combining Techniques

→ 3.5.1 HOMER

3.5.2 Scaled-world grab

3.5.3 Voodoo dolls

3.5.1 HOMER



Hand-centered Object Manipulation Extending Ray-casting

- Select an object (pointer-based ray-casting)
- Manipulate object with a virtual hand

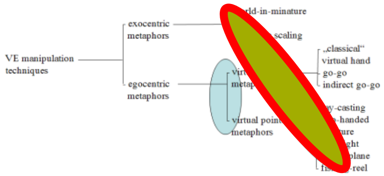
- Distance to virtual object at selection time determines scale factor α_h

$$r_v = \alpha_h \, r_r$$

$$\alpha_h = \frac{D_{virtual\ object}}{D_{real\ hand}}$$

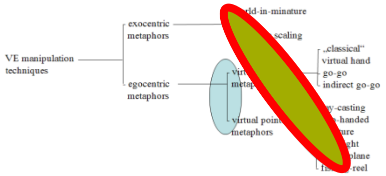
- Pro
  – Users can easily pull an object closer
- Con
  – It is difficult to push an object very much further away

3.5.2 Scaled-World Grab

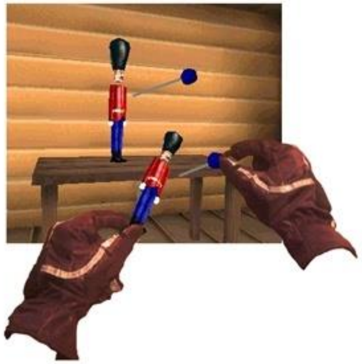


- Select an object (image-plane technique)
- Manipulate object
  - Down-scale entire VE around user‘s virtual viewpoint rather than scaling up user‘s hand motion
- Pro
  - Performs well for operations at a distance
- Con
  - May not be effective when user wants to „push away“ a close object

3.5.3 Voodoo Dolls



- Two-handed interaction
- Image-plane concept for selection (pinch gloves)
  Selected object → voodoo doll
- WIM-concept for manipulation
  Manipulate objects indirectly using temporary, miniature, handheld copies of objects („dolls“)
  - Doll in dominant hand: definition of position and orientation
  - Doll in non-dominant hand: stationary reference frame



[Pierce and Pausch 99]

3. Interaction Techniques for 3D Selection and Manipulation

3.1 Classifications of Selection and Manipulation Techniques

3.2 Interacting by Pointing

3.3 Direct Manipulation: Virtual Hand Techniques

3.4 World-in-Miniature

3.5 Combining Techniques

→ 3.6 Nonisomorphic 3D Rotation

3.7 Desktop 3D Manipulation

3.6 Nonisomorphic 3D Rotation

- Large ranges of virtual rotations controlled by small ranges of real rotations
  - Efficient use of limited rotational freedom in the real world (by scaling them down)
    - Suitable for users with disabilities
    - Minimizes need for *clutching*
  - Useful for highly accurate control of minimal rotations (by scaling them up)
    - Useful in tele-operation applications (robotic surgery)

3. Interaction Techniques for 3D Manipulation

3.1 Classifications of Selection and Manipulation Techniques
3.2 Interacting by Pointing
3.3 Direct Manipulation: Virtual Hand Techniques
3.4 World-in-Miniature
3.5 Combining Techniques
3.6 Nonisomorphic 3D Rotation
→ 3.7 Desktop 3D Manipulation

3.7 Desktop 3D Manipulation

- 6 DOF interaction devices are very expensive
- Many applications have to use cheap devices: WIMP-based tools
- Problems:
  - Mouse has only 2 DOF
- Solution:
  - Separation of DOFs
  - User switches between controlling different subsets of DOFs (potentially using both mouse buttons and arrow keys on keyboard)

3.7 Desktop 3D Manipulation

→ 3.7.1 2D interface controls
3.7.2 Surface metaphors
3.7.3 Indirect metaphors
3.7.4 3D widgets and handles
3.7.5 Virtual sphere
3.7.6 ARCBALL technique

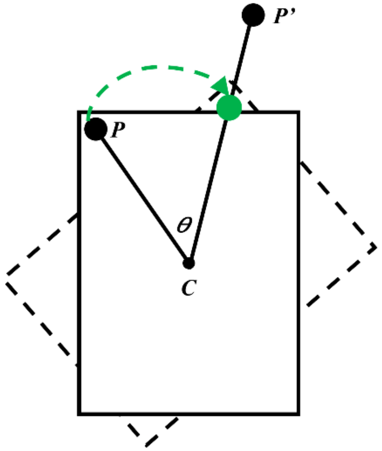
3.7.1 2D Interaction Controls

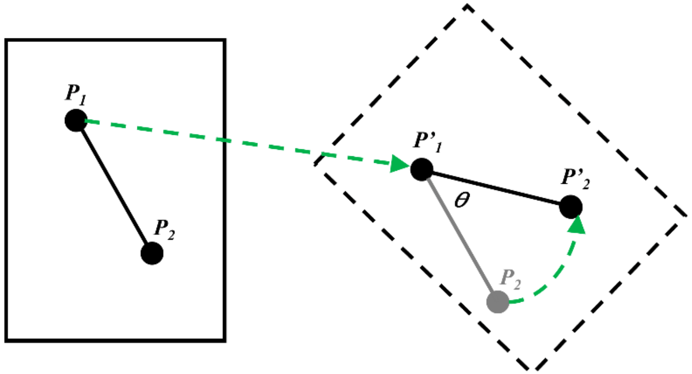
- Keyboard input:
  directly type in values for position and orientation
  - Precise definition of object pose
  - But: usually easier for users to specify object poses iteratively
- Separation of DOFs:
  - Use several, joint orthographic views of a scene
  - 2 DOFs per view, linked cursors
  - Alternative to direct mouse control: a separate slider for every dimension

3.7.2 Surface Metaphors

Surface-based 2D Interaction Techniques

- Direct touch (tablets, smartphones, surfaces)
  - Dragging: 1 finger → $P(x,y)$
  - Rotating: 2 fingers → $P_1(x,y)$, $P_2(\theta)$ around P_1
  - Rescaling (pinching) → scale ~ P_2-P_1

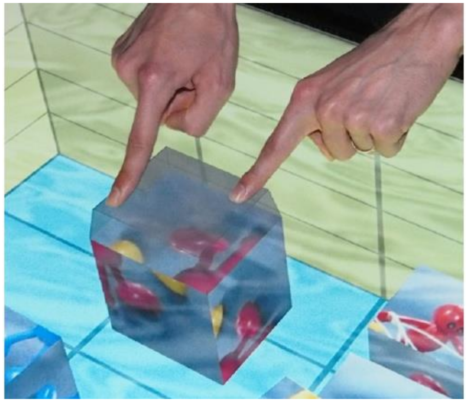


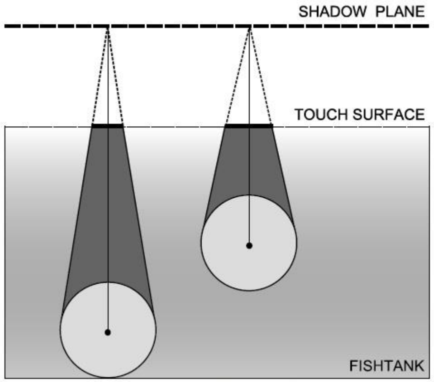


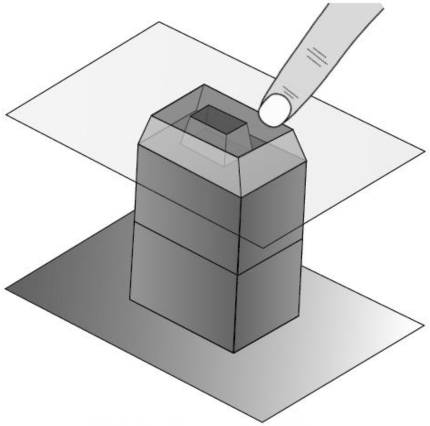
3.7.2 Surface Metaphors

Surface-based 3D Interaction Techniques

- Stereo displays
- Control depth relative to the surface
  - Pinching
    - Changing 2D scale
    - Changing 3D depth
  - Void shadows
    Deal with stacked (occluded) objects by presenting virtual shadows





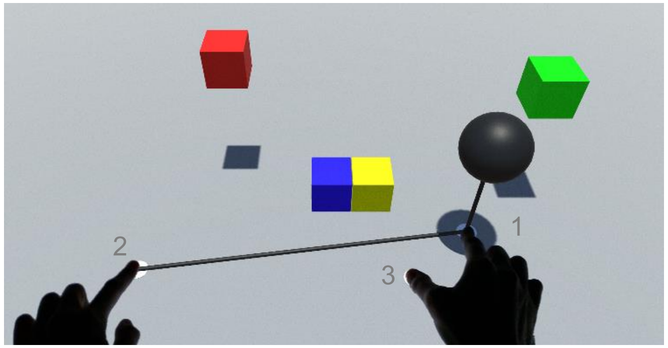


[Giesler et al 2014]

3.7.2 Surface Metaphors

Surface-based 3D Interaction Techniques

- Stereo displays
- Control depth relative to the surface
  - Balloon selection
    - Finger 1: anchor point P_1 = pos(x,y)
    - Finger 2: height $z = \left\|\overline{\overline{P_1 P_2}}\right\|$
      orientation of $\overline{\overline{P_1 P_2}}$
    - Finger 3: balloon size $s = \left\|\overline{\overline{P_1 P_3}}\right\|$
  - Significantly faster than keyboard-based 3 DOF positioning
  - Significantly lower error rate than simple virtual hand

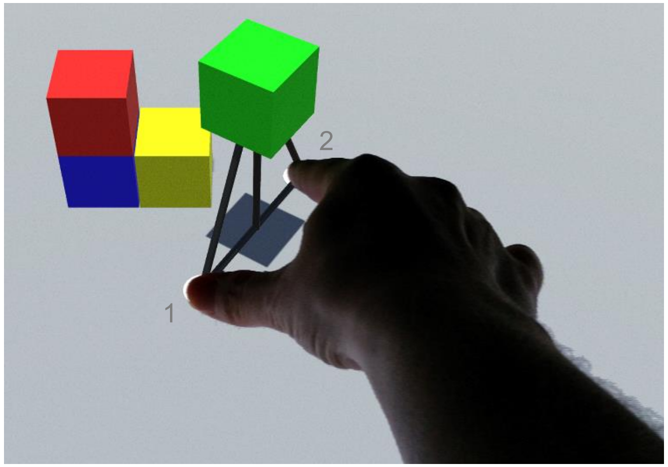


[Benko and Feiner 2007, image by McMahon]

3.7.2 Surface Metaphors

Surface-based 3D Interaction Techniques

- Stereo displays
- Control depth relative to the surface
  - Triangle cursor
    - Finger 1: P_1, Finger 2: P_2
    - anchor point = midpoint (P_1,P_2)
    - Height $z = \left\|\overline{P_1 P_2}\right\|$ (dist = 0 → z = 0)
    - Rotation = orientation of $\overline{P_1 P_2}$
  - Faster completion times than balloon selection
  - Fewer positional errors
  - Fewer orientational errors



[Strothoff et al 2011
image by McMahon]

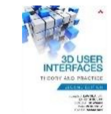
3.7.3 Indirect Metaphors

Manipulate virtual objects without directly interacting with them.

Benefits for

- Handling remote objects
- Avoiding occlusion
- Constraining manipulation by reducing DOFs

→ Indirect control-space
separate control from display



3.7.3 Indirect Metaphors

Indirect control-space

- Separate control from display
- Users can interact in a physical space distinct from the apparent location of the virtual environment (“unclutched”)

3.7.3 Indirect Metaphors

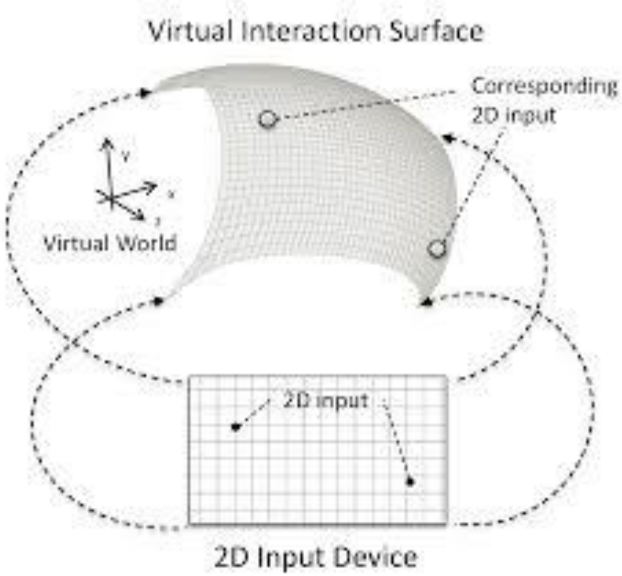
Indirect control-space

- Indirect touch
  - Use a multi-touch surface separate from the primary display
    - Select virtual object:
      - Finger 1: touch the multi-touch display to move a 3D cursor on the primary display
      - Finger 2: confirm selection of object under the cursor
    - Manipulate virtual object:
      - Conventional surface-based interaction

3.7.3 Indirect Metaphors

Indirect control-space

- Virtual interaction surface
  indirect touch for non-planar surfaces within the virtual environment
  - Mapping: 2D interaction (surface)
    → 3D interpretation (world)
    (aka texture mapping)

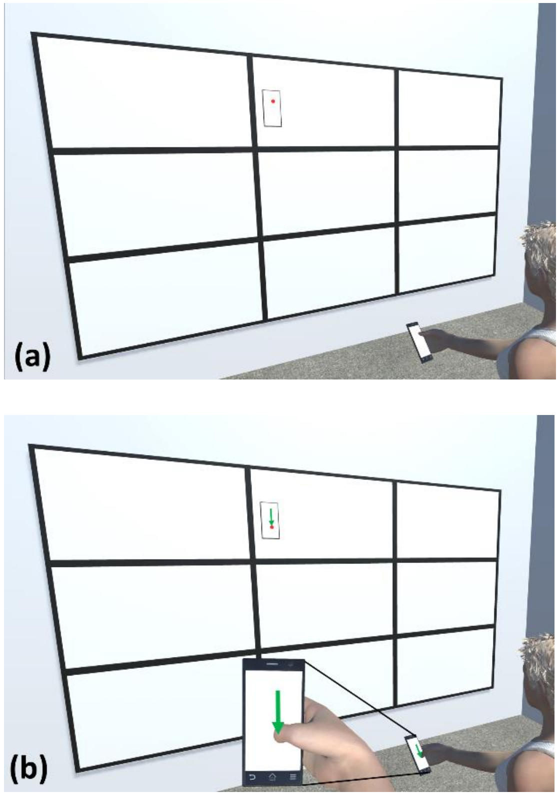


[Ohnishi et al 2012]

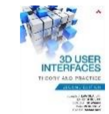
3.7.3 Indirect Metaphors

Indirect control-space

- Levels-of-precision cursor
  - Do not use a **fixed** multi-touch display; rather: use a **mobile** smartphone
  - Thus 3D interaction (motion) + 2D interaction (touch)
  - Staggered approach:
    - 3D motion for coarse selection
    - 2D touch-based motion for fine tuning



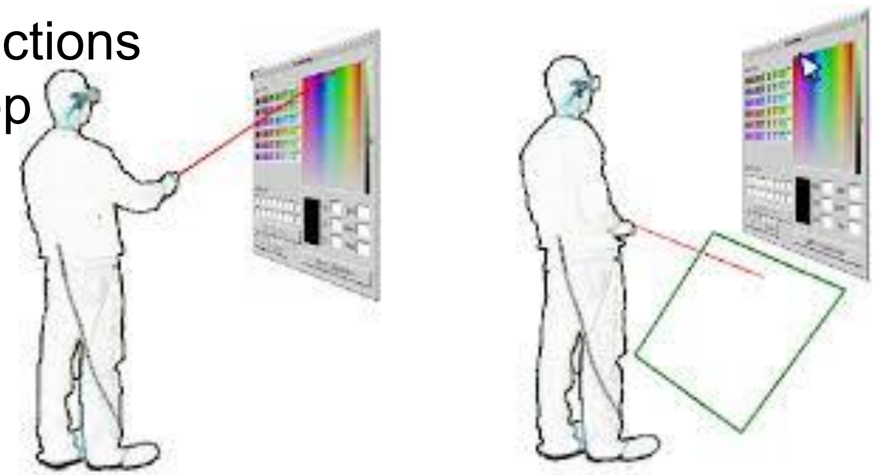
[Debarba 2012]



3.7.3 Indirect Metaphors

Indirect control-space

- Virtual pad
  - Interaction 2D adapted menus
  - Without a secondary multi-touch surface (use a nearby "virtual pad" instead)
    - User can customize interactions independently of 3D UI app



[Andujar and Argulaguet 2007]

3.7.4 3D Widgets and Handles

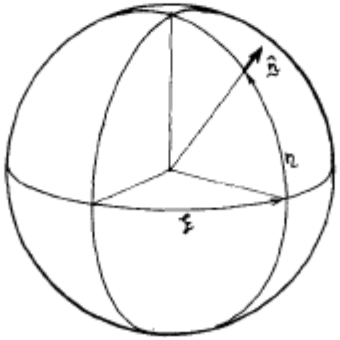
- Put controls directly into the 3D scene with the objects to be manipulated
- When an object is selected, a number of graphical manipulation widgets become visible
- Each widget is responsible for a small subset of DOFs

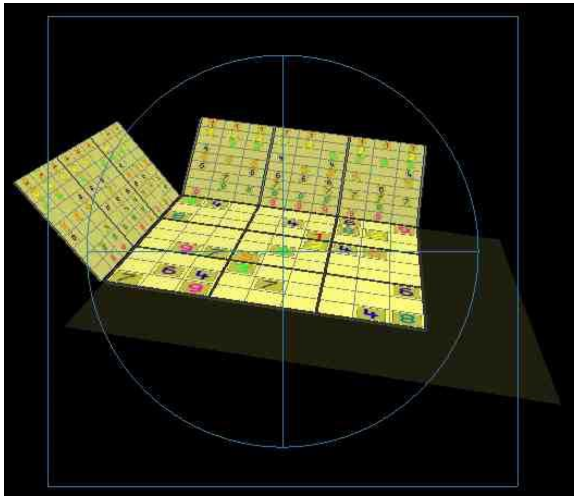
Widgets = visual manipulation constraints

- Pro
  - Easy and seamless transition between different manipulation sequences
- Con
  - Visual clutter

3.7.5 Virtual Sphere

- Several similarly integrated methods of using 2-DOF mouse movement
  - Virtual sphere
  - Rolling ball
  - Virtual trackball





- General approach
  - Enclose all rotations in a unit sphere, positioned around object center
  - Object rotations = rotations of unit sphere (grab it with your interaction devices at surface point P; pull P to a new location P´).
- Steps
  - Determine axis of rotation (perpendicular to mouse motion)
  - Calculate angle of rotation

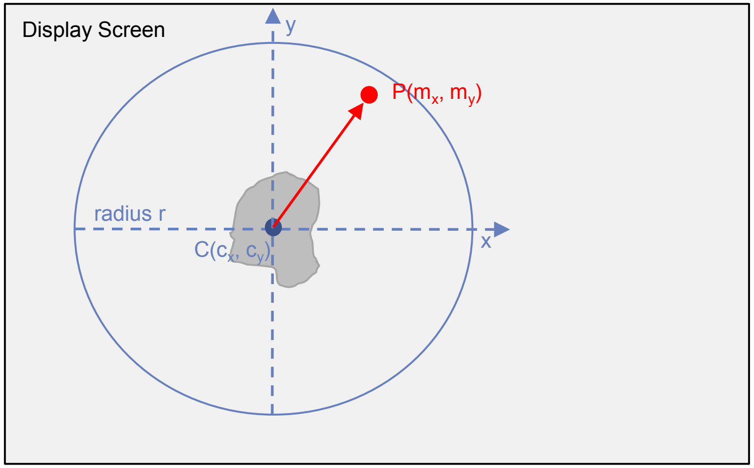
3.7.5 Virtual Sphere

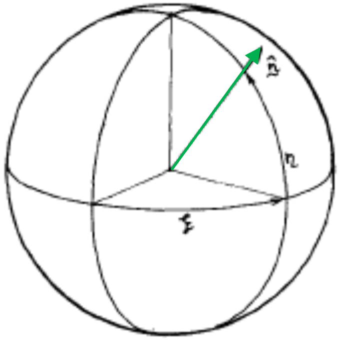
Gaussian Sphere surrounding an object

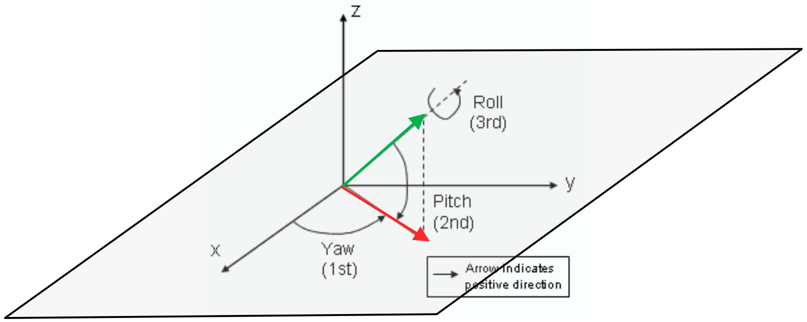
- centered at position $C(c_x,c_y)$
- with radius r
- mouse position $P(m_x,m_y)$

Interpret vector $C(c_x,c_y)P(m_x,m_y)$

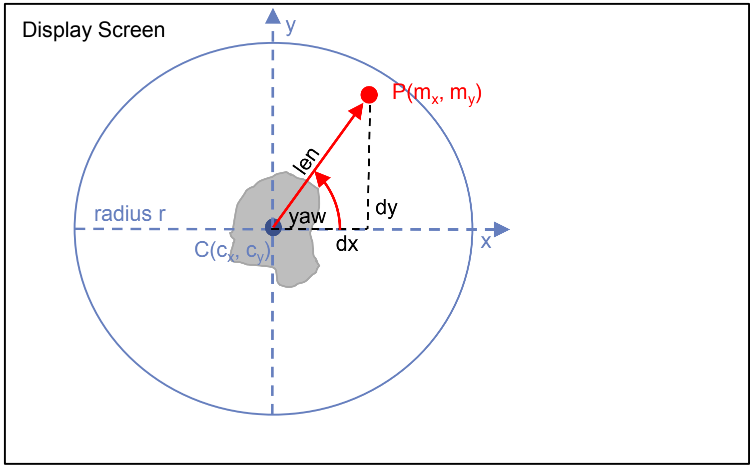
as a 3D vector „sticking out of the screen“







3.7.5 Virtual Sphere

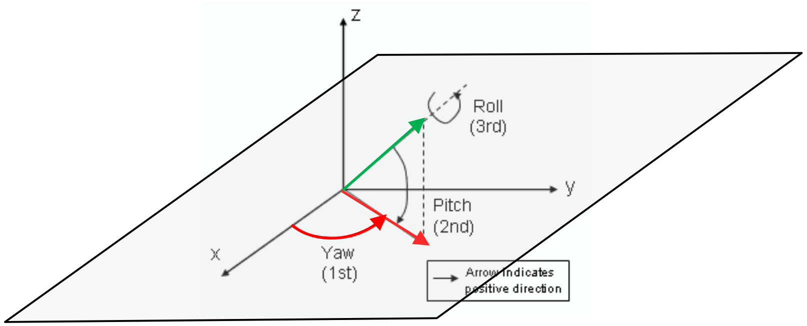


Gaussian Sphere surrounding an object

- centered at position $C(c_x,c_y)$
- with radius r
- mouse position $P(m_x,m_y)$

Interpret vector $C(c_x,c_y)P(m_x,m_y)$
as a 3D vector „sticking out of the screen“

$$yaw = \tan^{-1}(dy/dx)$$



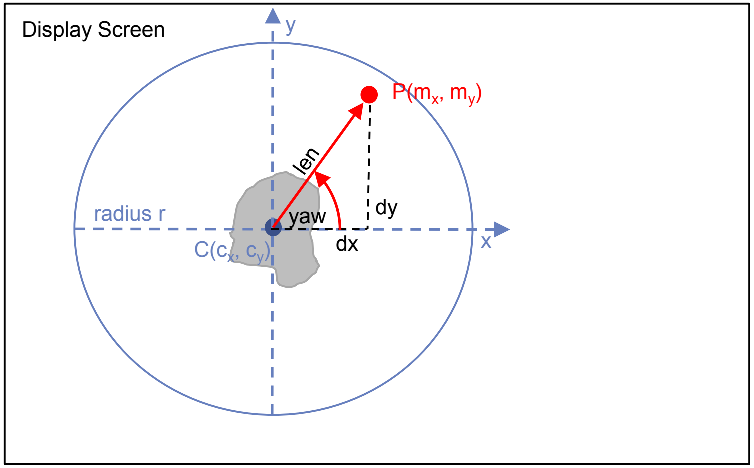
3.7.5 Virtual Sphere

Gaussian Sphere surrounding an object

- centered at position $C(c_x,c_y)$
- with radius r
- mouse position $P(m_x,m_y)$

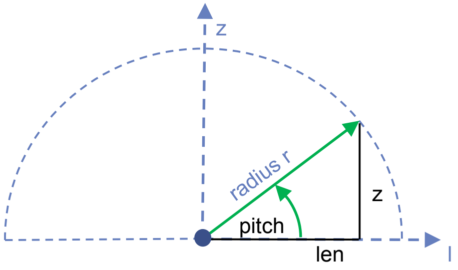
Interpret vector $C(c_x,c_y)P(m_x,m_y)$

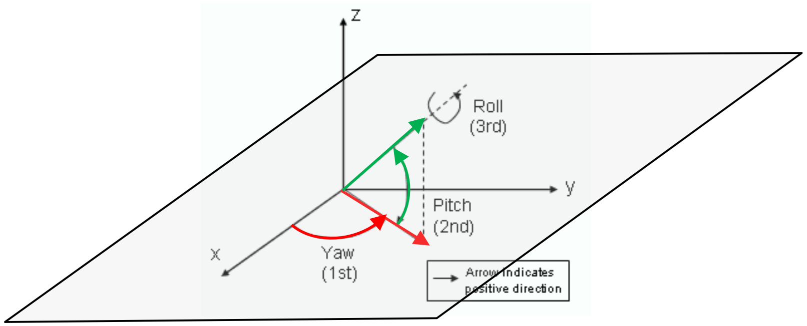
as a 3D vector „sticking out of the screen“



$$yaw = \tan^{-1}(dy/dx)$$
$$pitch = \cos^{-1}(len/r) \quad (in\ unit\ sphere)$$

$$z = \sin pitch$$
$$x = \cos yaw$$
$$y = \sin yaw$$





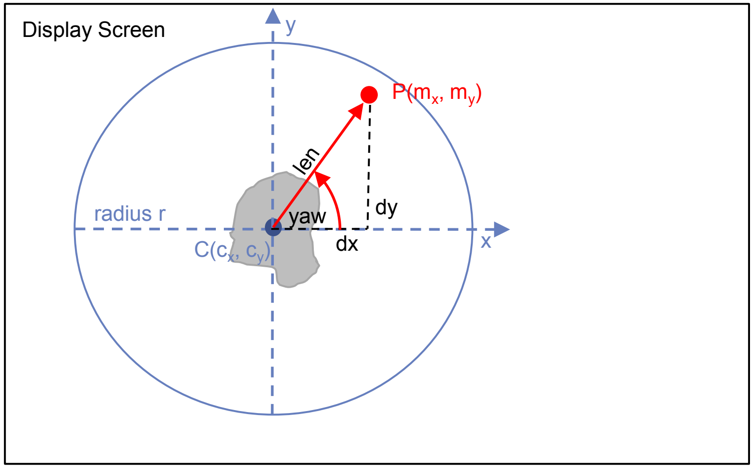
3.7.5 Virtual Sphere

Gaussian Sphere surrounding an object

- centered at position $C(c_x,c_y)$
- with radius r
- mouse position $P(m_x,m_y)$

Interpret vector $C(c_x,c_y)P(m_x,m_y)$

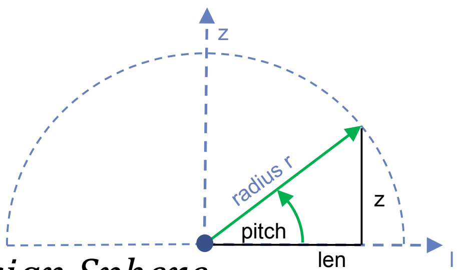
as a 3D vector „sticking out of the screen"



$$yaw = \tan^{-1}(dy/dx)$$
$$pitch = \cos^{-1}(len/r) \quad (in\ unit\ sphere)$$

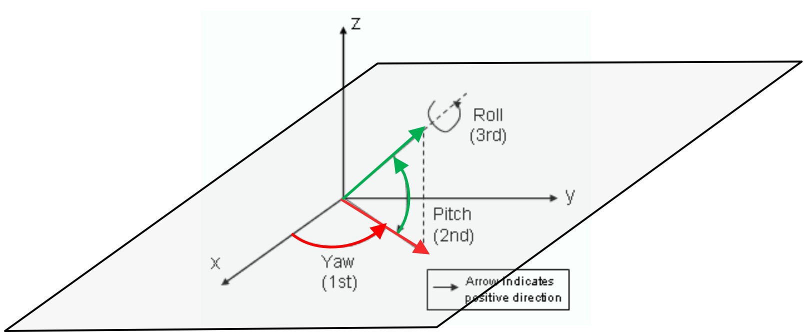
Rescale

$$z = \sin pitch \quad * r$$
$$x = \cos yaw \quad * r$$
$$y = \sin yaw \quad * r$$



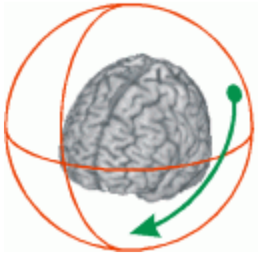
$len\ restricted\ to\ Gaussian\ Sphere$

$len \leq r$!!!



3.7.6 ARCBALL Technique

- Use spherical geometry to interpret 2D mouse movements as 3D rotations



http://www.sph.sc.edu/comd/rorden/3d.html

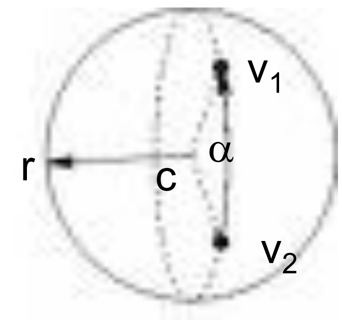
- Great circle (arc) on the sphere = rotation around the axis perpendicular to the circle
- Point on sphere = family of rotations (all great circles intersecting at this point)
- Rotation = 4D unit quaternion

http://rainwarrior.thenoos.net/dragon/arcball.html

3.7.6 ARCBALL Technique

ARCBALL technique

- 2 Vectors: v_1, v_2
- Rotation on a great circle
  (circle going through the sphere center c)

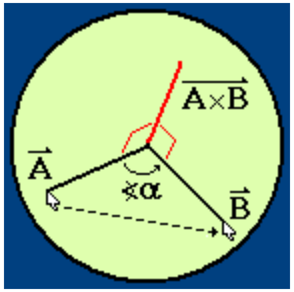


- Rotation axis: cross product

$$\text{rotation } axis\ r = v_1 \times v_2$$

- Rotation angle: scalar product

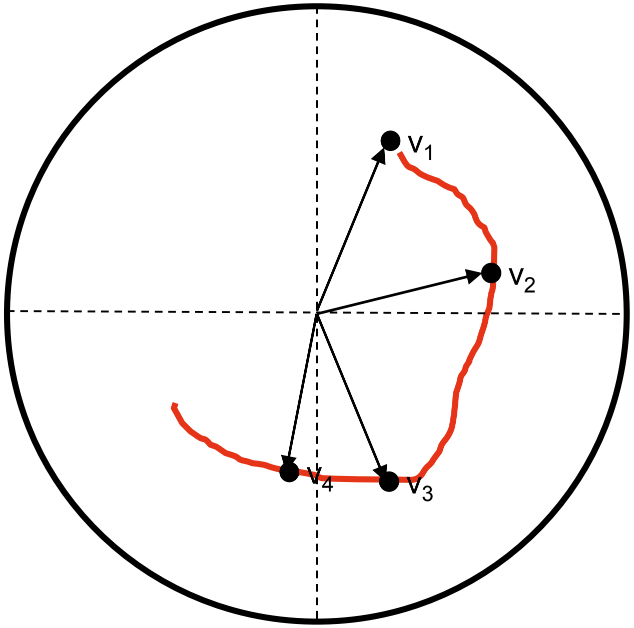
$$\text{rotation } angle\ \alpha = v_1 \cdot v_2$$



3.7.6 ARCBALL Technique

Interpretation of consecutive vector positions

- n Vectors: $v_1, v_2, v_3, \ldots v_n$



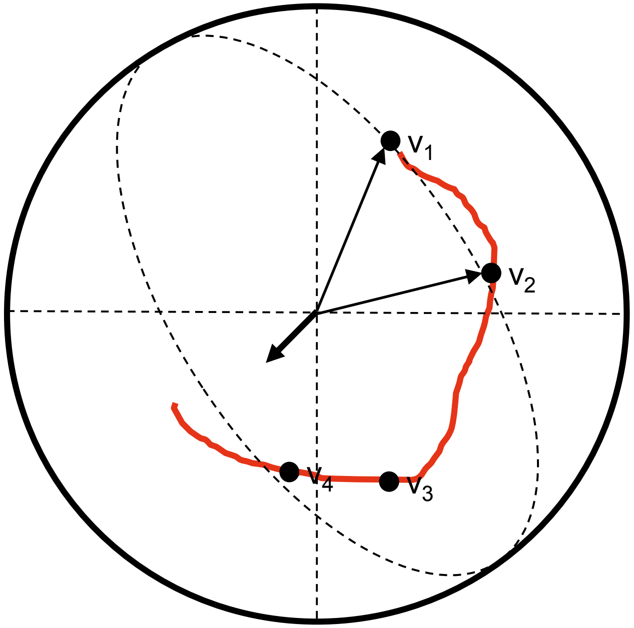
3.7.6 ARCBALL Technique

Interpretation of consecutive vector positions

- n Vectors: $v_1, v_2, v_3, \ldots v_n$

- Rotate v_i
  - With respect to v_1
    - Global (absolute)

$$r = v_i \times v_1$$
$$\alpha = v_i \cdot v_1$$



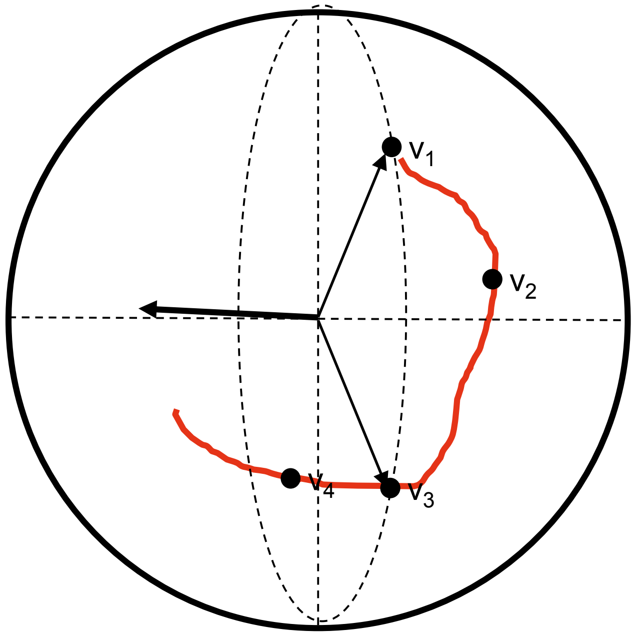
3.7.6 ARCBALL Technique

Interpretation of consecutive vector positions

- n Vectors: $v_1, v_2, v_3, \ldots v_n$

- Rotate v_i
  - With respect to v_1
    - Global (absolute)

$$r = v_i \times v_1$$
$$\alpha = v_i \cdot v_1$$



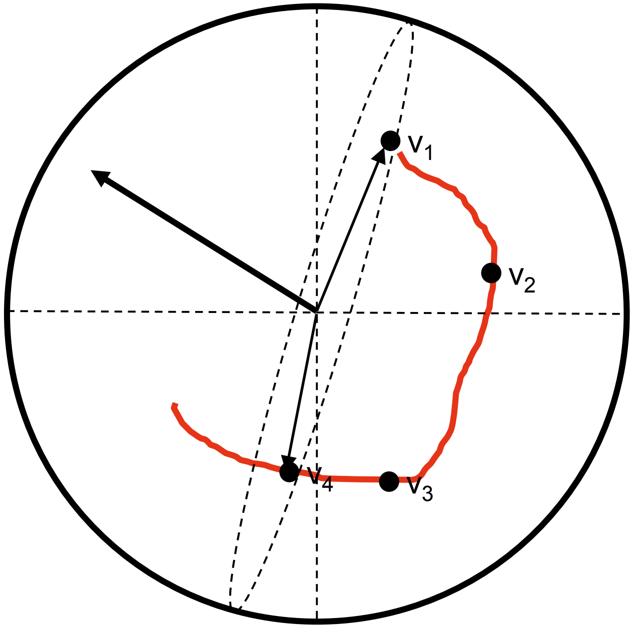
3.7.6 ARCBALL Technique

Interpretation of consecutive vector positions

- n Vectors: $v_1, v_2, v_3, \dots v_n$

- Rotate v_i
  - With respect to v_1
    - Global (absolute)

$$r = v_i \times v_1$$
$$\alpha = v_i \cdot v_1$$



3.7.6 ARCBALL Technique

Interpretation of consecutive vector positions

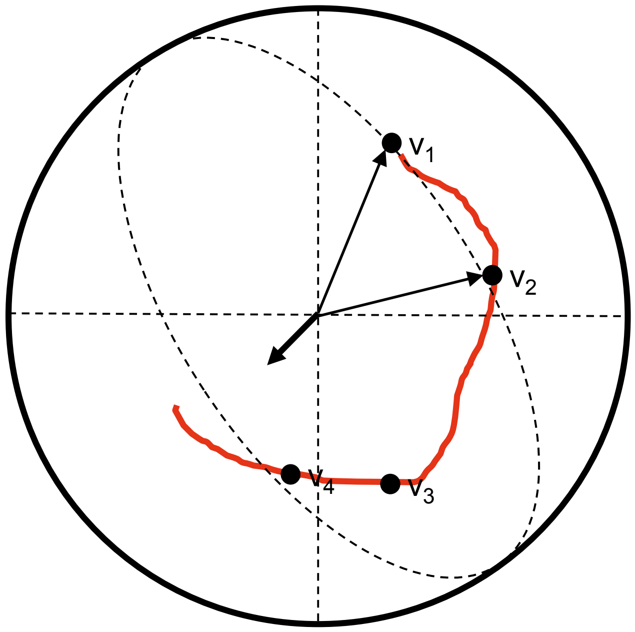
- n Vectors: $v_1, v_2, v_3, \ldots v_n$

- Rotate v_i
  - With respect to v_1
    - Global (absolute)

$$r = v_i \times v_1$$
$$\alpha = v_i \cdot v_1$$

  - With respect to v_{i-1}
    - Local (incrementally)

$$r = v_i \times v_{i-1}$$
$$\alpha = v_i \cdot v_{i-1}$$



3.7.6 ARCBALL Technique

Interpretation of consecutive vector positions

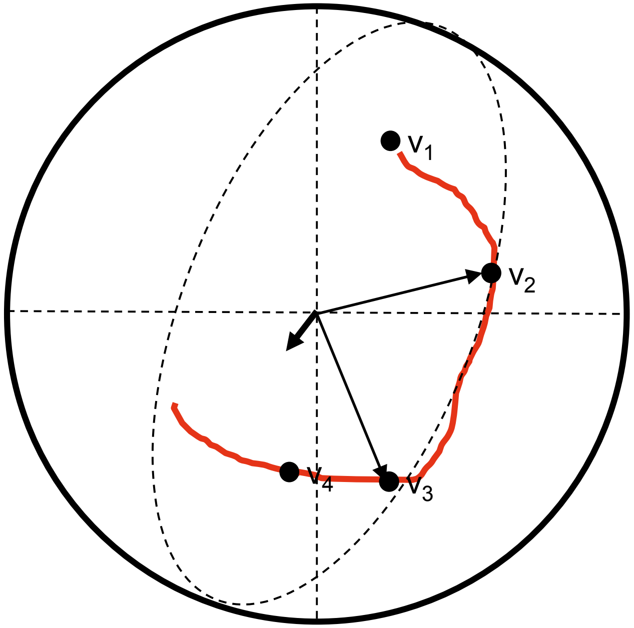
- n Vectors: $v_1, v_2, v_3, \ldots v_n$

- Rotate v_i
  - With respect to v_1
    - Global (absolute)

$$r = v_i \times v_1$$
$$\alpha = v_i \cdot v_1$$

  - With respect to v_{i-1}
    - Local (incrementally)

$$r = v_i \times v_{i-1}$$
$$\alpha = v_i \cdot v_{i-1}$$



3.7.6 ARCBALL Technique

Interpretation of consecutive vector positions

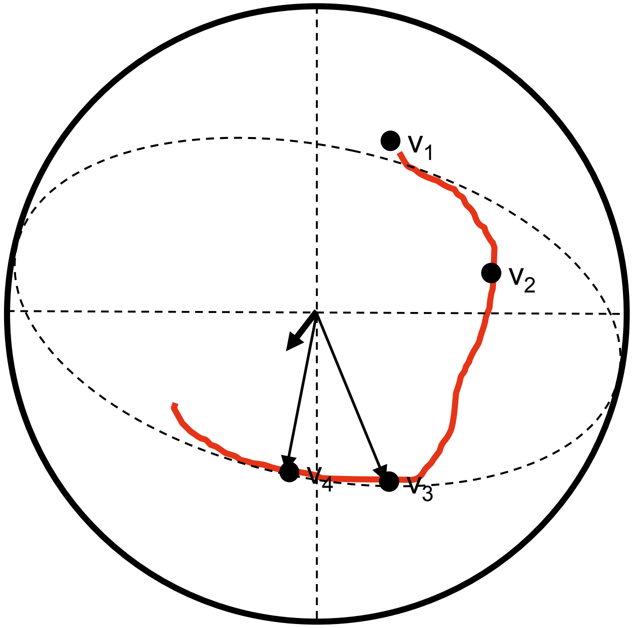
- n Vectors: $v_1, v_2, v_3, \ldots v_n$

- Rotate v_i
  - With respect to v_1
    - Global (absolute)

$$r = v_i \times v_1$$
$$\alpha = v_i \cdot v_1$$

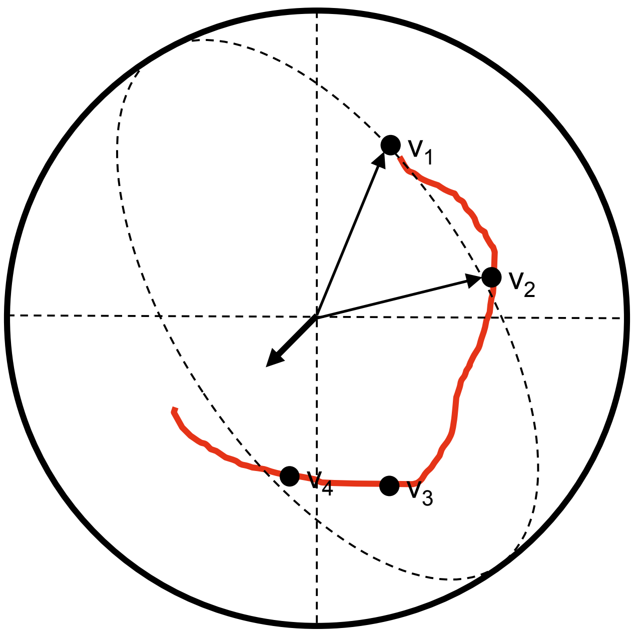
  - With respect to v_{i-1}
    - Local (incrementally)

$$r = v_i \times v_{i-1}$$
$$\alpha = v_i \cdot v_{i-1}$$

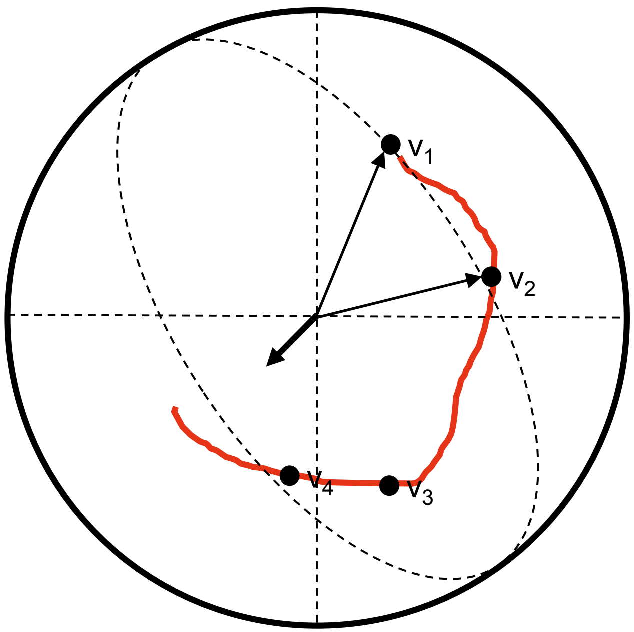


3.7.6 ARCBALL Technique

- Rotate v_i
  - with respect to v_1

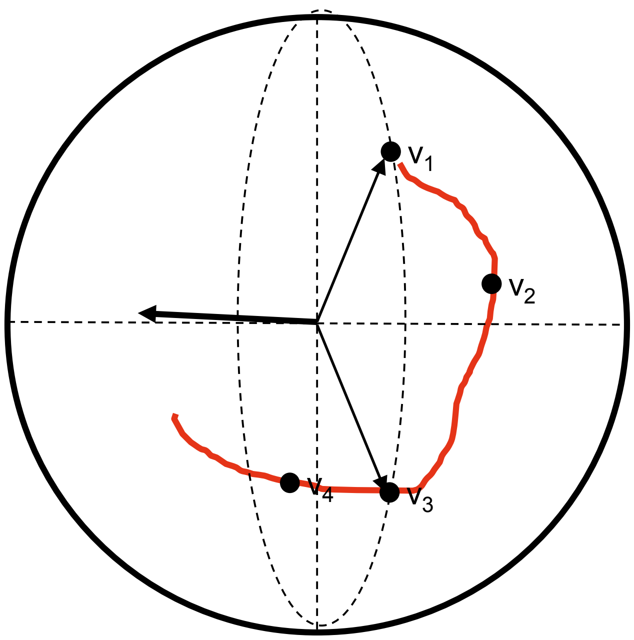


- Rotate v_i
  - With respect to v_{i-1}

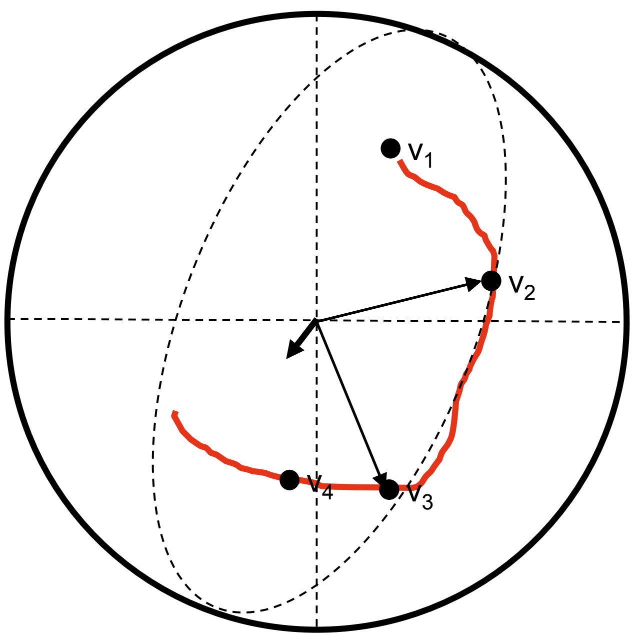


3.7.6 ARCBALL Technique

- Rotate v_i
  - with respect to v_1

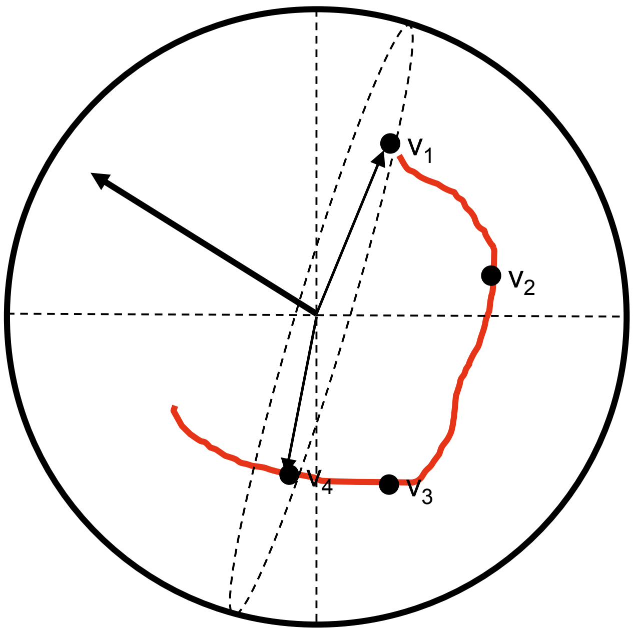


- Rotate v_i
  - With respect to v_{i-1}

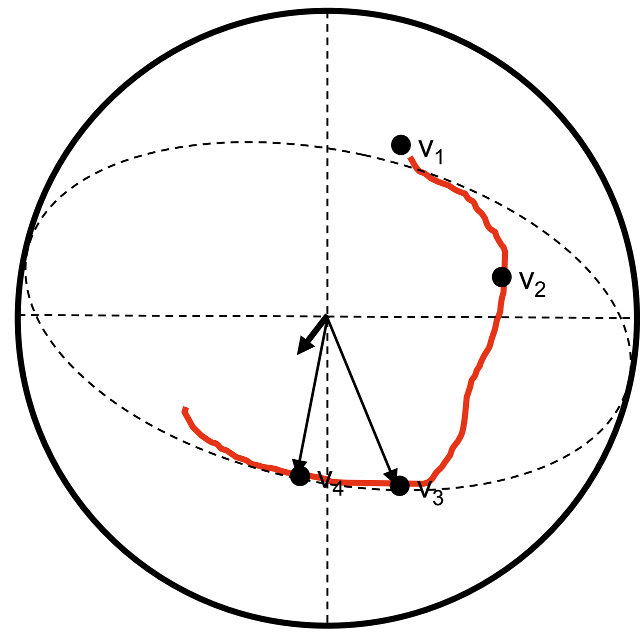


3.7.6 ARCBALL Technique

- Rotate v_i
  - with respect to v_1



- Rotate v_i
  - With respect to v_{i-1}



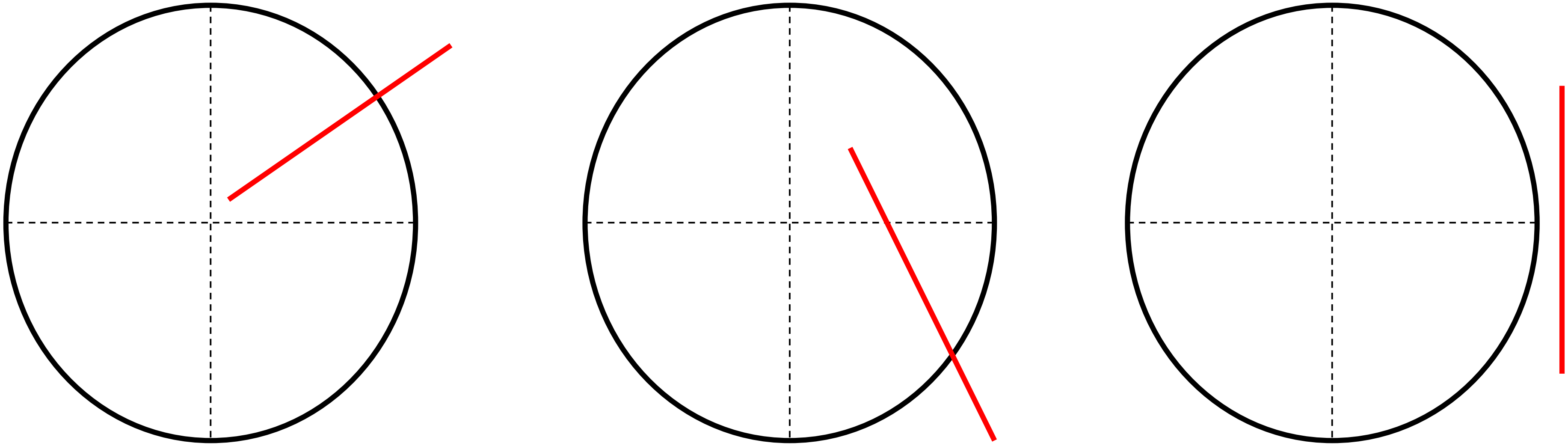
3.7.6 ARCBALL Technique

Interpretation of consecutive vector positions

- n Vectors: $v_1, v_2, v_3, \ldots v_n$

Further parameter choices:

- Fixed motion $r = v_1 \times v_2, \quad \alpha = \cdot\, v_1 v_2$
- Position (x,y), size of Gaussian Sphere r



Agenda

1. 3D Selection and Manipulation Tasks
2. Interaction Techniques and Input Devices
3. Interaction Techniques for 3D Manipulation
4. Design Guidelines

4. Design Guidelines

- Use existing manipulation techniques unless a large amount of benefit might be derived from designing a new, application-specific technique
- Use task analysis when choosing a 3D manipulation technique
- Match the interaction technique to the device
- Use techniques that can help to reduce clutching
- Non-isomorphic techniques are useful and intuitive
- Use pointing techniques for selection and virtual hand techniques for manipulation

4. Design Guidelines

- Use grasp-sensitive object selection
- Reduce degrees of freedom where possible
- Consider the tradeoff between technique design and environment design
- There is no single best manipulation technique

References

- [Olwal and Feiner 03]: A. Olwal and S. Feiner: The flexible pointer: An interaction technique for selection in augmented and virtual reality. Proc. UIST’03, 81-82.
- [Poupyrev et al 96]: I. Poupyrev, M. Billinghurst, S. Weghorst, and T. Ichikawa: The go-go interaction technique: non-linear mapping for direct manipulation in VR. Proc. UIST’96, 79-80.
- [Borst and Indugula 05]: C.W. Borst and A.P. Indugula: Realistic virtual grasping IEEE VR’05, 91-98.
- [Talvas et al 13]: A. Talvas, M. Marchal and M. Lécuyer: The god-finger method for improving 3D interaction with virtual objects through simulation of contact area. 3DUI 2013, 111-114.
- [Periverzov and Ilies 15]: F. Periverzov and H.T. Ilies: The intent driven selection method for natural user interfaces. 3DUI 2015, 121-128.
- [Pinho et al 08]: M.S. Pinho, D.A. Bowman, and C.M.D.S. Freitas: Cooperative object manipulation in collaborative virtual enviornments. J. Braz. Computer Science Society 14(2): 54-67, 2008.
- [Pierce and Pausch 99]: J.S. Pierce, B.C. Stearns and R.F. Pausch: Voodoo dolls: seamless interaction at multiple scales in virtual environments, SI3D 1999: 141-145.
- [Giesler et al 14]: A. Giesler, d. Valkov and K.H. Hinrichs: Void shadows: multi-touch interaction with stereoscopic objects on the tabletop. SUI 2014: 104-112.
- [Benko and Feiner 07]: H. Benko and S.K. Feiner: Balloon Selection: A multi-finger technique for accurate low-fatigue 3D selection. 3DUI’07: 111-117.

References

- [Strothoff et al 15]: S. Strothoff, W. Stuerzlinger and K.H. Hinrichs: Pins ‘n’ Touches: An interface for tagging and editing complex groups. ITS’15: 191-200.
- [Ohnishi et al 12]: T. Ohnishi, N. Katzakis, K. Kiyokawa and H. Takemura: Virtual Interaction Surface: Decoupling of Interaction and View Dimensions for Flexible Indirect 3D Interaction. 3DUI’12, 113-116.
- [Debarba et al 12]: H. Debarba, L. Nedel and A. Maciel: LOP-Cursor: Fast and Precise Interaction with Tiled Displays using One Hand and Levels of Precision. 3DUI’12, 125-132.
- [Andujar and Argelaguet 07]: C. Andjuar and F. Argelaguet: Virtual Pads: Decoupling Motor Space and VisualSpace for Flexible Manipulation of 2D Windows within VEs. 3DUI’07, 99-106.

Thank you!