3D User Interface Techniques for Selection and Manipulation

Lecture #9: Selection and Manipulation
Spring 2024
Joseph J. LaViola Jr.

Interaction Workflow
3D Interaction Techniques

- Choosing the right input and output devices not sufficient for an effective 3D UI
- Interaction techniques: methods to accomplish a task via the interface
  - Hardware components
  - Software components: control-display mappings or transfer functions
  - Metaphors or concepts
- Universal tasks: selection and manipulation, travel, system control

Overview

- Manipulation: a fundamental task in both physical and virtual environments
- 3D manipulation task types
- Classifications of manipulation techniques
- Techniques classified by metaphor:
  - Grasping
  - Pointing
  - Surface
  - Indirect
  - Bimanual
  - Hybrid
3D Manipulation Tasks

- Broad definition: any act of physically handling objects with one or two hands
- Narrower definition: spatial rigid object manipulation (shape preserving)

Canonical Manipulation Tasks

- **Selection**: acquiring or identifying an object or subset of objects
- **Positioning**: changing object’s 3D position
- **Rotation**: changing object’s 3D orientation
- **Scaling**: uniformly changing the size of an object
3D Manipulation Tasks

Canonical Manipulation Tasks

- Task parameters

<table>
<thead>
<tr>
<th>Task</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selection</td>
<td>Distance and direction to target, target size, density of objects around the target, number of targets to be selected, target occlusion</td>
</tr>
<tr>
<td>Positioning</td>
<td>Distance and direction to initial position, distance and direction to target position, translation distance, required precision of positioning</td>
</tr>
<tr>
<td>Rotation</td>
<td>Distance to target, initial orientation, final orientation, amount of rotation, required precision of rotation</td>
</tr>
</tbody>
</table>

Application-Specific Manipulation Tasks

- Canonical tasks can fail to capture important task properties for real applications
- Ex: positioning a medical probe relative to virtual models of internal organs in a VR medical training application
- Techniques must capture and replicate minute details of such manipulation tasks
3D Manipulation Tasks

Manipulation Techniques and Input Devices

- Number of control dimensions
- Integration of control dimensions
  - Multiple integrated DOFs typically best for 3D manipulation
- Force vs. position control
  - Position control preferred for manipulation
  - Force control more suitable for controlling rates

- Device shape
  - Generic vs. task-specific
- Device placement/grasp
  - Power grip
  - Precision grip
    - Use fingers
    - Reduce clutching
Classifications for 3D Manipulation

- Isomorphic (realistic) vs. non-isomorphic (magic)
- Task decomposition
- Metaphor

Grasping Metaphors

**Hand-Based Grasping**

- Simple virtual hand
- Go-Go

\[
\bar{r}_p = F(r_s) = \begin{cases} 
  r_s & \text{if } r_s \leq D \\
  r_s + \alpha(r_s - D)^2 & \text{otherwise}
\end{cases}
\]

where \( r_s = \text{length of } \hat{R}_s \)

\( r_s = \text{length of } \hat{R}_s \)

\( D, \alpha \) are constants
Grasping Metaphors

**Finger-Based Grasping**

- Rigid-body fingers
- Soft-body fingers
- God fingers

Rigid-body fingers (Borst and Indugla 2005)

- Need to track the hands and fingers (e.g., bend sensing glove or 3D depth camera)
- Map hand and finger positions to virtual hand and fingers
- Physics-based interactions
  - use virtual torsional and linear spring dampers
  - dynamically influence mapping between real and virtual hands
- Can be “sticky” – difficult to precisely release objects
- Sticky object problem can be reduced with better heuristic-based release functions
Grasping Metaphors

Soft-Body Fingers (Jacobs and Froehlich 2011)

- Use deformable representations for virtual fingers
- Lattice shape matching algorithm
  - deform the pads of virtual fingers to dynamically adapt to shapes of grasped objects
  - when real fingers initially collide with virtual objects, virtual finger pads deform slightly
  - when real fingers penetrate inner space of virtual objects, more points of collision produced for virtual fingers
- Implicit friction model compared to rigid body model

God Fingers (Talvas et al. 2013)

- god object – a virtual point that adheres to rigid body physics and never penetrates virtual objects (remains on their surface)
  - force direction can be easily calculated
- Goal is to use god-objects for finger grasping and manipulation
  - compute contact area about god-object point as if surface was flat
  - contact area fitted to geometry of the object based on god object force direction
  - odd deformations are prevented by using angular threshold between force directions and surface normals
Grasping Metaphors

Enhancements for Grasping Metaphors

- 3D bubble cursor
- PRISM
- Hook
- Intent-driven selection

3D Bubble Cursor (Vanacken et al. 2007)

- Semi-transparent sphere that dynamically resizes itself to encapsulate the nearest virtual object
- Designed for selecting a single object
- When sphere is too large and begins to intersect a nearby object a second semi-transparent sphere is created to encapsulate that object
Grasping Metaphors

PRISM (Frees and Kessler 2005)
- Precise and Rapid Interaction through Scaled Manipulation
- Apply scaled down motion to user’s virtual hand when the physical hand is moving below a specified speed
  - decreased control to display gain
  - increased precision
- Causes mismatch between virtual and physical hand location
  - use offset recovery mechanism based on hand speed
  - allows virtual hand to catch up to physical

Hook (Ortega 2013)
- Supports object selection of moving objects
- Observe relationship between moving objects and the hand to develop tracking heuristics
  - compute distance of hand to each virtual object
  - orders and scores targets based on increasing distance
  - close targets have scores increased, far targets have scores deceased
- When selection is made, target with highest score is selected
Grasping Metaphors

Intent-Driven Selection (Periverzov and Llies 2015)
- Use posture of virtual fingers as confidence level in object selection
- Proximity sphere is positioned within grasp of virtual hand
  - virtual fingers touch the sphere
  - anything within the sphere is selectable
- As hand closes, additional proximity spheres are made to specify a smaller subset of selectable objects until one target is selected

Pointing Metaphors
- Pointing is powerful for selection
  - Remote selection
  - Fewer DOFs to control
  - Less hand movement required
- Pointing is poor for positioning
- Design variables:
  - How pointing direction is defined
  - Type of selection calculation
Pointing Metaphors

Vector-Based Pointing Techniques

- Ray-casting
- Fishing reel
- Image-plane pointing

Ray-casting

- Simple pointing technique
- Point at object with virtual ray
  - virtual line indicates direction (e.g., laser pointer)
  - size of the virtual line can vary
- Perform ray casting to select desired object
- Precision can be compromised with far away objects
Pointing Metaphors

Fishing Reel
- Additional input mechanism to control the virtual ray
- Select with ray casting and real the object back and forth using additional input (e.g., slider, gesture)

Image Plane Pointing (Pierce et al. 1997)
- Image plane techniques simplify object selection by using 2 DOF
  - select and manipulate objects with their 2D projections
  - use virtual image plane in front of user
  - simulate direct touch
- Used to manipulate orientation, not position
- Examples include Head Crusher, Lifting Palms, Sticky Finger, and Framing
Pointing Metaphors

Volume-Based Pointing Techniques

- Flashlight
- Aperture
- Sphere-casting

**Flashlight**
- Provides soft selection and does not require as much precision
- Instead of using a ray, a conic selection volume is used
- Apex of cone is at the input device
- Object does not have to be entirely within the cone
- Must deal with disambiguation issues
  - choose object closer to the centerline
Pointing Metaphors

**Aperture Selection (Forsberg et al. 1996)**
- Modification of flashlight technique
- User can interactively control the spread of the selection volume
- Pointing direction defined by 3D position of user’s viewpoint (tracked head location) and position of a hand sensor
- Moving hand sensor closer or farther away changes aperture

**Sphere Casting**
- Define position of predefined volume at the intersection of a vector used for pointing and the VE
- Modified version of ray casting
  - casts sphere onto nearest intersected surface
Pointing Metaphors

Enhancements for Pointing Metaphors
- Bendcast
- Depth ray
- Absolute and relative mapping

Bendcast (Riege et al. 2006)
- Pointing analog to 3D bubble cursor
- Bends the pointing vector toward object closest to the vector’s path
  - point line distance from each selectable object is calculated
  - circular arc used to provide feedback
Pointing Metaphors

**Depth Ray (Vanacken et al. 2007)**
- Used to disambiguate which object the user intends to select when pointing vector intersects multiple targets
- Uses depth marker along the ray length
- Object closest to the marker is selected
- User can control marker by moving a tracked input device back or forward

**Absolute and Relative Mapping (Kopper et al. 2010)**
- Useful in dense environments
- Provides manual control of control to display gain ratio of pointing
  - Lets users increase the effective angular width of targets
- Can give user impression of slow motion pointer
Surface Metaphors

**Surface-Based 2D Interaction Techniques**
- Dragging
- Rotating

**Surface-based 3D Interaction Techniques**
- Pinching
- Void shadows
- Balloon selection
- Corkscrew widget
- Triangle cursor
Indirect Metaphors

Indirect Control-Space Techniques
- Indirect touch
- Virtual interaction surface
- Levels-of-precision cursor
- Virtual pad

Indirect Touch (Simeone 2016)
- Touch multi-touch surface to control cursor on primary display
- With second finger touch the surface to select an object under the cursor
- Use surface-based techniques for manipulation
- Choice of absolute or relative mapping
Indirect Metaphors

Virtual Interaction Surfaces (Ohnishi et al. 2012)
- Extension of indirect touch
- Mapping of multi-touch surface to nonplanar surfaces in VE
- Allow user to manipulate objects relative to desired paths or other objects
- Supports drawing directly on complex 3D surfaces

Levels-of-Precision Cursor (Debarba et al. 2012)
- Extends indirect touch with physical 3D interactions
- Uses smartphone
  - affords multi-touch and 3D interaction using inertial sensors and gyroscopes
- Map smaller area of smartphone to larger area of primary display
- Determine orientation for pointing operations
Indirect Metaphors

Virtual Pad (Andujar and Argelaguet 2007)
- Does not require multi-touch surface
- Virtual surface within the VE is used
- Similar to image plane methods

Indirect Proxy Techniques
- World in miniature
- Voodoo Dolls
Indirect Metaphors

**World in Miniature (Stoakley et al. 1995)**
- Scale entire world down and bring within user’s reach
- Miniature hand held model of the VE (exact copy)
- Manipulating object in WIM indirectly manipulates object in the VE
- Many design decisions for implementation
  - has scaling issues

**Voodoo Dolls (Pierce et al. 1999)**
- Builds upon WIM and image plane techniques
- Seamless switching between different reference frames for manipulation
  - manipulate objects indirectly using temporary handheld copies of objects (dolls)
  - user can decide which objects to manipulate by using image plane selection (no scaling issues)
- Two handed technique
  - non-dominant hand represents a stationary reference frame
  - dominant hand defines position and orientation of object relative to stationary reference frame
  - user can pass doll from one hand to the other
Indirect Metaphors

Indirect Widget Techniques
- 3D widgets
- Virtual sphere
- Arcball

Bimanual Metaphors
- Dominant and non-dominant hands
- Symmetric vs. asymmetric
- Synchronous vs. asynchronous
- Ex: balloon selection is asymmetric (two hands have different functions) and synchronous (two hands operate at the same time)
Bimanual Metaphors

Symmetric Bimanual Techniques

- Spindle
- iSith

Spindle (Mapes and Moshell 1995)

- Two 6 DOF controllers used to define a virtual spindle that extends from one controller to another
  - center of spindle represents primary point of interaction
- Translation – move both hands in unison
- Rotation – yaw and roll by rotating hands relative to each other
- Scale – lengthen or shorten distance of hands
Bimanual Metaphors

iSith (Wyss et al. 2006)
- Intersection-based Spatial Interaction for Two Hands
- Two 6 DOF controllers define two separate rays
  - ray-casting with both hands
  - shortest line between two rays is found by crossing two vectors to find vector perpendicular to both
  - known as projected intersection point (point of interaction)

Bimanual Metaphors

Asymmetric Bimanual Techniques
- Spindle + Wheel
- Flexible pointer
Bimanual Metaphors

Spindle + Wheel (Cho and Wartell 2015)
- Extended Spindle to include rotating pitch of virtual object
- Uses virtual wheel collocated with dominant hand cursor
  - twist dominant hand for rotation

Flexible Pointer (Olwal and Feiner 2003)
- Make use of two handed pointing
- Curved ray that can point at partially occluded objects
  - implemented as quadratic Bezier spline
Hybrid Metaphors

- Aggregation of techniques
- Integration of techniques
  - HOMER
  - Scaled-world grab

HOMER (Bowman and Hodges 1997)
- Hand-centered Object Manipulation
- Extended Ray-Casting
- Select object using ray casting
- Users hand then attaches to the object
- User can then manipulate object (position and orientation) with virtual hand
Hybrid Metaphors

**Scaled World Grab (Mine et al. 1997)**
- User selects object with given selection technique
- Entire VE is scaled down around user’s virtual viewpoint
- Scaling is done so object is within user’s reach
- If center of scaling point is midway between user’s eyes, the user will be unaware of the scaling

Other Aspects of 3D Manipulation

**Nonisomorphic 3D rotation**
- Amplifying 3D rotations to increase range and decrease clutching
- Slowing down rotation to increase precision
- Absolute vs. relative mappings
  - Absolute mappings can violate *directional compliance*
  - Relative mappings do not preserve *nulling compliance*
Isomorphic vs. Non-Isomorphic Philosophies

- Human-Machine interaction
  - input device
  - display device
  - transfer function (control to display mapping)
- Isomorphic – one-to-one mapping
- Non-isomorphic – scaled linear/non-linear mapping

Non-Isomorphic 3D Spatial Rotation

- Important advantages
  - manual control constrained by human anatomy
  - more effective use of limited tracking range (i.e. vision-based tracking)
  - additional tools for fine tuning interaction techniques
- Questions
  - faster?
  - more accurate?
Rotational Space

- Rotations in 3D space are a little tricky
  - do not follow laws of Euclidian geometry
- Space of rotations is not a vector space
- Represented as a closed and curved surface
  - 4D sphere or manifold
- Quaternions provide a tool for describing this surface

Quaternions

- Four-dimensional vector \((\mathbf{v}, w)\) where \(\mathbf{v}\) is a 3D vector and \(w\) is a real number
- A quaternion of unit length can be used to represent a single rotation about a unit axis \(\hat{u}\) and angle \(\theta\) as

\[
q = (\sin(\frac{\theta}{2})\hat{u}, \cos(\frac{\theta}{2})) = e^{\frac{\theta}{2}\hat{u}}
\]
Linear 0th Order 3D Rotation

- Let \( q_c \) be the orientation of the input device and \( q_d \) be the displayed orientation then
  
  \[
  (1) \quad q_c = (\sin(\frac{\omega}{2})\hat{u}_c, \cos(\frac{\omega}{2})) = e^{\frac{\omega}{2} \hat{u}_c}
  \]
  
  \[
  (2) \quad q_d = (\sin(\frac{k\omega}{2})\hat{u}_d, \cos(\frac{k\omega}{2})) = e^{\frac{k\omega}{2} \hat{u}_d} = q_o^k
  \]

- Final equations w.r.t. identity or reference orientation \( q_o \) are
  
  \[
  (3) \quad q_q = q_c^k \quad (4) \quad q_d = (q_c q_o^{-1})^k q_o, \quad k = \text{CD gain coefficient t}
  \]

Non-Linear 0th Order 3D Rotation

- Consider
  
  \[
  (3) \quad q_d = q_c^k \quad (4) \quad q_d = (q_c q_o^{-1})^k q_o
  \]

- Let \( k \) be a non-linear function as in
  
  \[
  \omega = 2 \arccos(q_c \cdot q_o) \quad \text{or} \quad \omega = 2 \arccos(w)
  \]

  \[
  k = F(\omega) = \begin{cases} 
    1 & \text{if } \omega < \omega_o \\
    f(\omega) = 1 + c(\omega - \omega_o)^2 & \text{otherwise}
  \end{cases}
  \]

  where \( c \) is a coefficient and \( \omega_o \) is the threshold angle
Design Considerations

- Absolute mapping – taken on $i$-th cycle of the simulation loop
  \[ q_{d_i} = q_{c_i} \]

- Relative mapping – taken between the $i$-th and $(i-1)$-th cycle of the simulation loop
  \[ q_{d_i} = (q_{c_i} q_{c_{i-1}}^{-1})^k q_{d_{i-1}} \]

Absolute Non-Isomorphic Mapping

- Generally do not preserve directional compliance
- Strictly preserves nulling compliance
Relative Non-Isomorphic Mapping

- Always maintain directional compliance
- Do not generally preserve nulling compliance

Amplified Non-Linear Rotation for VE Navigation (1)

- Users expect the virtual world to exist in any direction
  - 3-walled Cave does not allow this
  - adapt expected UI to work in restricted environment
- Amplified rotation allows users to see a full 360 degrees in a 3-walled display
- A number of approaches were tested
  - important to take cybersickness into account
Amplified Non-Linear Rotation for VE Navigation (2)

- Apply a non-linear mapping function to the user’s waist orientation $\theta$ and his or her distance $d$ from the back of the Cave.
- Calculate the rotation factor using a scaled 2D Gaussian function:
  \[ \phi = f(\theta, d) = \frac{1}{\sqrt{2\pi\sigma_1}} e^{-\frac{(|\theta| - \pi - (1-d/L))^2}{2\sigma_2^2}} \]

- The new viewing angle is $\theta_{\text{new}} = \theta(1 - \phi)$

Amplified Non-Linear Rotation for VE Navigation (3)

\[ \sigma_1 = 0.57 \]
\[ \sigma_2 = 0.85 \]
\[ L = 30 \]
\[ \mu = \pi \]
Non-Linear Translation for VE Navigation (1)

- Users lean about the waist to move small to medium distances
  - users can lean and look in different directions
- Users can also lean to translate a floor-based interactive world in miniature (WIM)
  - Step WIM must be active
  - user’s gaze must be 25 degrees below horizontal

Non-Linear Translation for VE Navigation (2)

- Leaning vector $\vec{L}_R$ is the projection of the vector between the waist and the head onto the floor
  - gives direction and raw magnitude components
- Navigation speed is dependent on the user’s physical location
  - Leaning sensitivity increases close to a boundary
- Linear function $L_T = a \cdot D_{\text{min}} + b$

- Mapped velocity $v = \left\|\vec{L}_R\right\| - L_T$
Non-Linear Translation for VE Navigation (3)

- Navigation speed is also dependent on the user’s head orientation with respect to the vertical axis
  - especially useful when translating the floor-based WIM
- Mapping is done with a scaled exponential function
  \[ F = \alpha \cdot e^{-\beta |\vec{H} \cdot \vec{V}_{up}|} \]
- Final leaning velocity is
  \[ v_{final} = F \cdot v \]

Other Aspects of 3D Manipulation

**Multiple Object Selection**

- Serial selection mode
- Volume-based selection techniques
  - e.g., flashlight, aperture, sphere-casting
- Defining selection volumes
  - e.g., two-corners, lasso on image plane
- Selection-volume widget
  - e.g., PORT
Other Aspects of 3D Manipulation

**Progressive Refinement**
- Gradually reducing set of objects till only one remains
- Multiple fast selections with low precision requirements
- SQUAD
- Expand
- Double Bubble

**SQUAD (Kopper et al. 2011)**
- Sphere-casting refined by QUAD menu
  - progressive refinement for dense VEs
- User specifies initial subset of environment using sphere cast
- Selectable objects laid out in QUAD menu
- Use ray-casting to select one of the four quadrants
  - selected quadrant is laid out in four quadrants
  - repeat until one object is selected
Other Aspects of 3D Manipulation

**Expand (Cashion et al. 2012)**
- Similar to SQUAD
- User selects collection of objects
- User’s view expands this area and creates clones of the selectable objects (laid out in grid)
- User uses ray-cast to select object

**Double Bubble (Bacim 2015)**
- Both SQUAD and Expand suffer from initial selection containing large set of objects
- 3D bubble cursor is used upon initial selection
  - bubble not allowed to shrink beyond a certain size
- Objects laid out in a menu and selected using 3D bubble cursor
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.

Nonisomorphic (“magic”) techniques are useful and intuitive.
- Use pointing techniques for selection and grasping techniques for manipulation.
- Consider the use of grasp-sensitive object selection.
- Reduce degrees of freedom when possible.
- Consider the trade-off between technique design and environment design.
- There is no single best manipulation technique.
Case Studies

**VR Gaming Case Study**
- Bimanual approach:
  - Non-dominant hand defines interaction area ("flashlight")
  - Dominant hand selects/manipulates in that area ("tool")
- Playful metaphors, multiple tools
- Key concepts:
  - Progressive refinement selection techniques can help users avoid fatigue by not requiring precise interactions.
  - Basic 3D selection and manipulation techniques can be customized to fit the theme or story of a particular application.

**Mobile AR Case Study**
- Finger-based selection for infrequent use with single datasets
- Pen-based selection for frequent use or richer datasets
- Key concepts:
  - Size of selectable items: keep the size of your selectable objects or menu items as small as possible, while reflecting the limitations of your input method and the visibility (legibility) of these items.
  - Selection method: depending on the frequency of selection tasks, different input methods could be preferable. Often, there is a direct relationship between input method, selection performance and frequency, and user comfort.
Conclusion

- 3D manipulation is a foundational task in 3D UIs
- Huge design space with many competing considerations
- Consider tradeoffs in your application context carefully

Next Class

- Navigation – Travel
- Readings
  - 3DUI Book – Chapter 7