INTERACTION DESIGN AND EVALUATION

Pere-Pau Vázquez
Dept. Computer Science – UPC

INTERACTION DESIGN AND EVALUATION. SESSION 2

Pere-Pau Vázquez
Dept. Computer Science – UPC

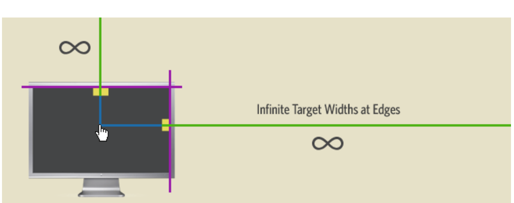
OUTLINE

- **Fitts Law in UI Design**
  - Implications
  - Applications
  - Fitts' Law in Mobile Devices
- Accelerating Target Acquisition
- Law of Crossing
- Steering Law
- Pointing Devices
- 3D Selection

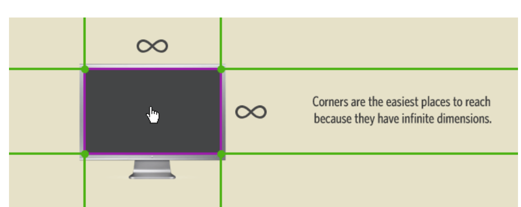
FITTS' LAW IN UI DESIGN. IMPLICATIONS

- Fitts Law accurately predicts pointing movement
  - If improvement required, it can help us modify our UI
    - Change target width:
      - Increase size for faster reach
    - Change pointer movement:
      - Increase speed

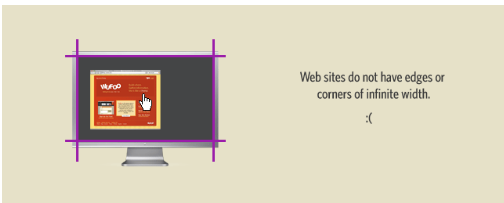
FITTS' LAW IN UI DESIGN. IMPLICATIONS



FITTS' LAW IN UI DESIGN. IMPLICATIONS



FITTS' LAW IN UI DESIGN. IMPLICATIONS



FITTS' LAW IN UI DESIGN. IMPLICATIONS

- Mac OS scrolls are faster to navigate

FITTS' LAW IN UI DESIGN. APPLICATIONS

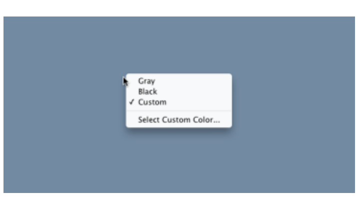
- Keep related things close
  - And opposite elements far
    - These two different buttons should be placed far away from each other

FITTS' LAW IN UI DESIGN. APPLICATIONS

- Keep related things close
  - And opposite elements far
    - Filters should be placed close to the search field

FITTS' LAW IN UI DESIGN. APPLICATIONS

- Pop-up menus reduce traveling distance

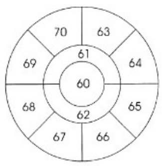


FITTS' LAW IN UI DESIGN. APPLICATIONS

- Pop-up menus. Improve two aspects:
  - Reduction of distance to travel
    - The option is close to the menu emerging place
  - Frequency-enabled may improve the time to pick an option:
    - Based on Hick-Hyman
      - Recall that users are able to point faster objects that are known
- Only used by experts!

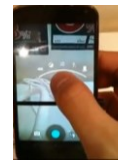
FITTS' LAW IN UI DESIGN. APPLICATIONS

- What about pie menus?



FITTS' LAW IN UI DESIGN. APPLICATIONS

- What about pie menus
  - Sort of contextual menu
    - Needs to be created on demand
    - Needs some room!
  - Should not have occlusions
    - On mobile half-pie menus better than fully circular



FITTS' LAW IN UI DESIGN. APPLICATIONS

- Pie menus difficult to design!
  - Second layer changes the size
    - And distance
  - Organizing by frequency may be a problem (learning)

FITTS' LAW IN UI DESIGN. APPLICATIONS

- Pie menus can easily be designed in a defective way

FITTS' LAW IN UI DESIGN. APPLICATIONS

- Grouping things may improve over distance

FITTS' LAW IN MOBILE DEVICES

- For mobile take into account the thumb zones
  - Consider Fitts only within the operation range of the thumb
    - Outside elements require extra effort

This Is Why the iPhone's Screen Will Always Be 3.5 Inches

Why does the iPhone have a 3.5-inch screen? Why do larger smartphones feel awkward on your hand? Dustin Curtis has an answer, and I think it is spot on:

> *Touching the upper right corner of the screen on the Galaxy S II using one hand, with its 4.27-inch screen, while you're walking down the street looking at Google Maps, is extremely difficult and frustrating. I pulled out my iPhone 4 to do a quick test, and it turns out that when you hold the iPhone in your left hand and articulate your thumb, you can reach almost exactly to the other side of the screen.*

His graphic shows this clearly. It makes total sense. And that is exactly why we would never see any larger screen iPhone. That 3.5-inch screen will be the ideal size until all humans are 7-feet tall and have hands the size of frying pans. [dcurtis]

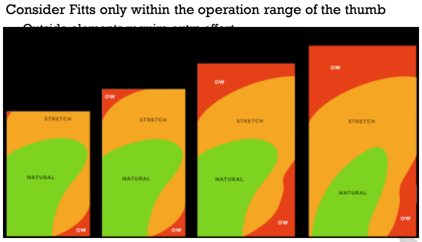
MOVING

FITTS' LAW IN MOBILE DEVICES

[redacted]

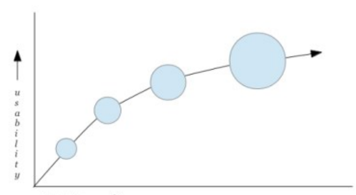
FITTS' LAW IN MOBILE DEVICES

- For mobile take into account the thumb zones
  - Consider Fitts only within the operation range of the thumb



FITTS' LAW IN MOBILE DEVICES

- Predicted usability of a button according to its size



FITTS' LAW IN MOBILE DEVICES

- Some "editing" actions must be dealt with care (send, upload, download, burn, share…):
  - Possibility of undoing (even temporarily)
    - E. g. Google's mail
  - Make item boundaries visible
  - Highlight focused items
    - May prevent accidentally return

FITTS' LAW IN MOBILE DEVICES

- Rule of the thumb
  - Make destructive/delicate tasks more difficult
    - Increasing the effort to prevent accidents
      - Buttons for non –destructive
      - Slides for destructive

OUTLINE

- *Fitts Law in UI Design*
- **Accelerating Target Acquisition**
  - Expanding Targets
  - Expanding Cursors
  - Target Moving
  - Control-Display Ratio
- Law of Crossing
- Steering Law
- Pointing Devices
- 3D Selection

ACCELERATING TARGET ACQUISITION

- Screen size is getting larger
  - Targets can be larger
    - Or not
  - Distances tend to be larger
- According to Fitts Law:
  - The larger the target the faster the access
    - We can reduce pointing times for frequent/important targets
  - The larger the distance the slower the access
    - We must balance target size and screen size
  - The faster the movement the faster the access
    - May accelerate mouse cursor speed

ACCELERATING TARGET ACQUISITION. EXPANDING TARGETS

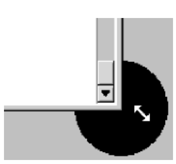
- Increase the size of targets close to the pointer
  - Pros:
    - Improve selection times
  - Needs more space:
    - Overlapping targets
    - Moving targets

ACCELERATING TARGET ACQUISITION. EXPANDING TARGETS

- Bubble targets:
  - Increase selectable region around target
    - Only when the mouse is close
    - Improves selection times
  - Issues:
    - Bubble appearing may distract users
    - Close selection points may generate several bubbles

ACCELERATING TARGET ACQUISITION. EXPANDING TARGETS

- Bubble targets:



ACCELERATING TARGET ACQUISITION. EXPANDING TARGETS

- Increase the size of targets close to the pointer
  - Implemented in Mac OSX Dock:
    - Targets resize and move
      - Increase size when getting closer and decreasing size when passed
      - Move towards the pointer and far from it

ACCELERATING TARGET ACQUISITION. EXPANDING TARGETS

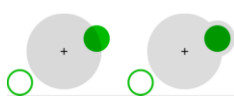
ACCELERATING TARGET ACQUISITION. EXPANDING TARGETS

- Increase the size of targets close to the pointer. Issues:
  - Moving targets reduces selectable size
    - Some users get frustrated
    - Especially on vertical (vs horizontal moves of the targets) moves
    - Target scaling when close to the pointer is sometimes confusing
  - May reduce effects if overlapping is allowed

Pere-Pau Vázquez – pere.pau@cs.upc.edu
MOVING

ACCELERATING TARGET ACQUISITION. EXPANDING CURSORS

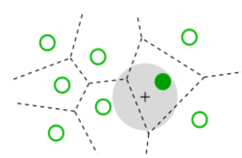
- Bubble cursor [Grossman2005]:
  - Reduction of amplitude movement
    - Cursor size increases when it is close to objectives
    - It may even grow to *absorb* closer objectives if its size does not allow it to
  - Based on position, no speed
    - In experiments Control-Display ratio fixed to 1



Pere-Pau Vázquez – pere.pau@cs.upc.edu
MOVING

ACCELERATING TARGET ACQUISITION. EXPANDING CURSORS

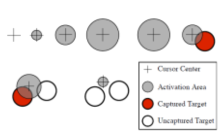
- Bubble cursor:
  - Previous determination of the area of influence of each target
  - Voronoi map of the targets
    - Once we know in which area we are, we know the closer target and the distance



Pere-Pau Vázquez – pere.pau@cs.upc.edu
MOVING

ACCELERATING TARGET ACQUISITION. EXPANDING CURSORS

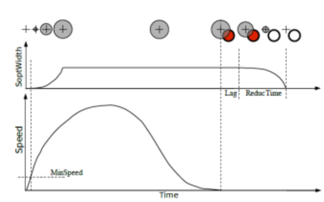
- DynaSpot [Chapuis2009]:
  - Reduction of amplitude by area cursor increase
    - Area increases according to speed and position
    - Visual cues to indicate the captured target



Pere-Pau Vázquez – pere.pau@cs.upc.edu
MOVING

ACCELERATING TARGET ACQUISITION. EXPANDING CURSORS

- DynaSpot:
  - Area change of the cursor according to speed



Pere-Pau Vázquez – pere.pau@cs.upc.edu
MOVING

ACCELERATING TARGET ACQUISITION. TARGET MOVING

- May reduce selection time
  - Reducing distance to the pointer
- Different strategies:
  - **Move targets closer** to the user
  - **Generate** targets next to the user

ACCELERATING TARGET ACQUISITION. TARGET MOVING

- Move targets to the user:
  - Mac OSX Dock
    - Though movement is relatively small
    - Studies have demonstrated no effective gain
  - Issues:
    - Difficult to correctly determine the appropriate target
    - Moving elements on screen cause spatial disorganization
      - May eliminate other benefits

ACCELERATING TARGET ACQUISITION. TARGET MOVING

- Generate targets next to the user:
  - Pop-up menus
    - Very useful, though for power users
    - Reduce pointer movement
    - Many techniques:
      - Classical menus, pie menus, semi-circular menus

ACCELERATING TARGET ACQUISITION. TARGET MOVING

- Generate targets next to the user:
  - Pie and semi-circular menus
    - Further reduction of pointer movement
      - Distance from the cursor is *constant* and very small

ACCELERATING TARGET ACQUISITION. TARGET MOVING

- Sticky targets:
  - Attract pointer
    - When the pointer is close to a selectable area
    - May reduce selection time
      - Precision not required
    - Users adapt easily

ACCELERATING TARGET ACQUISITION. CONTROL-DISPLAY RATIO

- Relation between the amplitude of movements of the user's real hand and the amplitude of movements of the virtual cursor
  - Moves in real world (physical move) mapped to moves in virtual desktop (cursor move)
  - No clear how the mapping affects perception and productivity
    - Some studies say it is not intuitive
    - Some studies say it improves some pointing tasks

Accelerating Target Acquisition. Control-Display Ratio

- Different strategies:
  - Constant
  - Dependent on mouse speed
  - Dependent on cursor position
- Interpretation according to Fitts Law:
  - Dynamic C-D ratio adaptation can be interpreted as dynamic change of physical motor space

Accelerating Target Acquisition. Control-Display Ratio

- Mac OSX and Windows both use mouse acceleration
  - When mouse moves fast, it is accelerated
    - Reducing the amplitude of movement to cover large distances
  - When mouse moves slow, it is decelerated
    - Magnifying amplitude of movement to improve precision

Outline

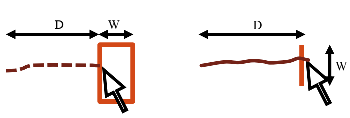
- *Fitts Law in UI Design*
- *Accelerating Target Acquisition*
- **Law of Crossing**
- Steering Law
- Pointing Devices
- 3D Selection

Law of Crossing

- Stylus or fingers naturally lead to crossing gestures
  - We are used to
  - Especially useful in tactile devices
    - Drag & drop, sketch…
- It may be investigated in the same way
  - So that we can predict both time and error rates
  - So that we can improve UI design
    - Or detect problems

Law of Crossing

- Crossing movement as compared to pointing



Law of Crossing

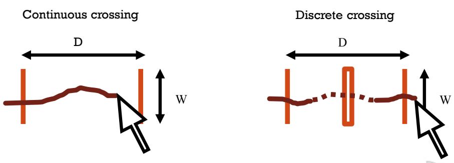
- Crossing performance across two goals [Accot99, Zhai2002]:
  - Follows the same characterization than the Fitts' Law:

$$T = a + b\log_2\left(\frac{D}{W} + 1\right)$$

    - T is the average moving time between passing the two goals.
    - *D* is the distance between the two goals
    - *W* is the width of each goal
    - *a* and *b* are constants to be determined

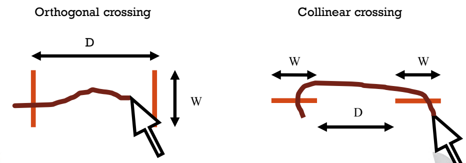
LAW OF CROSSING

- Crossing configurations:
  - Discreteness vs continuity of the movement:
    - Landing [and lifting off the stylus]



LAW OF CROSSING

- Crossing configurations:
  - Direction of the targets vs direction of the movement:
    - Targets can be orthogonal to the direction of the movement, or parallel
    - If parallel, the trace will be larger



LAW OF CROSSING

- Results of the experiments:
  - Crossing-based interfaces achieve similar (or faster) times than pointing.
  - The error rate in crossing is smaller than in pointing.
  - Discrete crossing becomes more difficult if the distance between the targets is small.
  - Crossing (especially continuous) seems superior than pointing for *ID* values > 5.

OUTLINE

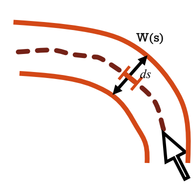
- *Fitts Law in UI Design*
- *Accelerating Target Acquisition*
- *Law of Crossing*
- **Steering Law**
- Pointing Devices
- 3D Selection

STEERING LAW

- Navigating through a constrained path is an useful operation in modern UIs
  - Navigating through nested menus
  - 3D navigation
  - Dragging elements
  - Free-hand Sketching/Drawing

STEERING LAW

- Steering: Navigating with a directional constraint:



STEERING LAW

- Navigating through a **generalized path** can be expressed as [Accot97]:
  - Movement time across the path T_s:

$$T_s = a + b\int_C \frac{ds}{W(s)}$$

  - *C* is the length of the path
  - *W(s)* is the path width at point s

STEERING LAW

- Navigating through a **generalized path** can be expressed as [Accot97]:
  - Movement time across the path T_s follows Fitts' expression:

$$T_s = a + bID_s$$

  - Where ID_s is:

$$ID_s = \int_C \frac{ds}{W(s)}$$

STEERING LAW

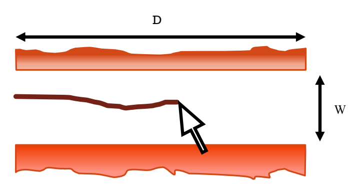
- Steering through a path
  - Interpreted as an infinite sum of goal-crossing tasks
  - Time to cross goal of width *W* at distance *D* follows Fitts' Law equations
  - Crossing *N* consecutive goals gives an ID_N of:

$$ID_N = N\log_2\left(\frac{D}{NW}+1\right)$$

  - Taking N to infinite yields ID_s

STEERING LAW

- Steering through a **straight path**:



STEERING LAW

- Time to navigate through a **straight path** (tunnel) T_p[Accot97]:

$$T_P = a + b\frac{D}{W}$$

  - D is the length of the path/tunnel
  - W is the width of the path/tunnel
- Applying Fitts formatting:

$$T_P = a + bID_P \qquad ID_P = \frac{D}{W}$$

  - Which also applies to circular paths of constant width

STEERING LAW

- Results [Accot97, Zhai2004] show that the steering law is applicable to different configurations:
  - Different path shapes: cone, spiral, straight
  - Works with different devices
  - Can be used to analyse navigation through nested menus, compare menu designs…

STEERING LAW

- Results [Accot97, Zhai2004] show that the steering law is applicable to different configurations:
  - Works for more complex interactions such as locomotion in a VR setup
    - Straight paths, circular paths…

OUTLINE

- *Fitts Law in UI Design*
- *Accelerating Target Acquisition*
- *Law of Crossing*
- *Steering Law*
- **Pointing Devices**
- 3D Selection

POINTING DEVICES

- Direct-control devices: Work directly on the surface of the screen
- Indirect-control devices: Work away from the surface

POINTING DEVICES

- Direct-control devices:
  - Old
    - Lightpen worked back in 1976
  - May produce fatigue:
    - Moving the lightpen on the screen required much effort
      - Should have a surface to rest the arm

POINTING DEVICES

- Direct-control devices. Lightpen

POINTING DEVICES

- Direct-control devices. Issues:
  - Imprecision in pointing. Many factors:
    - Quality of the screen
      - Capacitive screens less precise than resistive
    - Size of the pointer
      - *Fat* and *not-so-fat* fingers

POINTING DEVICES

- Direct-control devices. Issues:
  - Land-on strategy:
    - Select on clicking point
    - Faster feedback
    - Prone to errors
  - Lift-off strategy:
    - Initial click creates cursor, dragging used for precision pointing, lift-off selects
    - More time consuming

POINTING DEVICES

- Direct-control devices. Advantages:
  - Touch screens can be designed with no moving parts
    - Durable
    - Only device that has survived Walt Disney's theme parks
  - Multi-touch allows for complex data entry or manipulation
    - Pinch to zoom

POINTING DEVICES

- Direct-control devices. Other issues:
  - Pens may be more suitable for some tasks
    - Reduce occlusion
    - Familiar to users
    - But require to be picked up and put down
  - Fingers are less precise than wrist-based movement

POINTING DEVICES

- Indirect-control devices. Issues:
  - Alleviate hand fatigue
  - Eliminate screen occlusion
  - Mouse is the clear king
    - Cost-effective
    - Precise
    - Hand has a surface to rest on
    - Buttons easy to press
    - Long movements require to pick up mouse and replace
      - May be improved using accelerated moves

POINTING DEVICES

- Indirect-control devices. Examples:
  - Mouse, trackball, joystick, touchpad, graphics tablets…

POINTING DEVICES

- Pointing devices. Performance depends on two main factors:
  - Speed
  - Accuracy
- Some tasks may be best suited for some devices
  - Scrolling with a mouse wheel

OUTLINE

- *Fitts Law in UI Design*
- *Accelerating Target Acquisition*
- *Law of Crossing*
- *Law of Steering*
- *Pointing Devices*
- **3D Selection**

3D SELECTION

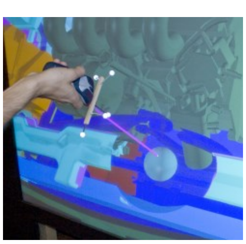
- 3D interfaces can make several tasks easier than classical 2D systems
  - Even better than reality?
- **3D selection:** selection task in a 3D immersive environment

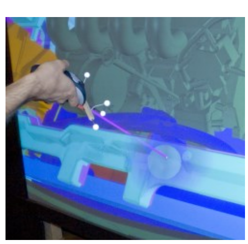
3D SELECTION

- Definitions
  - **3D interaction**
    - HC Interaction where user's tasks are carried out in a 3D spatial context
    - Using 3D or 2D input devices with direct mappings to 3D
  - **3D user interface**
    - A User Interface that involves 3D interaction.
  - **3D interaction technique**
    - Technique designed for solving a task
      - Involves the use of hardware and software

3D SELECTION

3D SELECTION





3D SELECTION

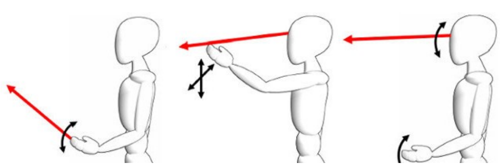
- Selection techniques:
  - Hand extension techniques or 3D point cursors
    - A 3D point in space is represented as a mapping of the user's hand position.
  - Ray-based techniques
    - Use the hand position and some element to indicate orientation
    - A ray is generated a ray in space and is used as a pointer
      - Also called aperture-based selection techniques or ray cursors

3D SELECTION

3D SELECTION

- Hand extension:
  - May require ample movements due to the direct mapping with 3D world
  - Sometimes elements are difficult to reach
  - May be more intuitive if virtual world represents some real world

3D SELECTION

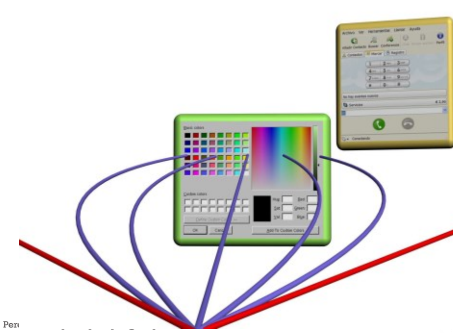


3D SELECTION

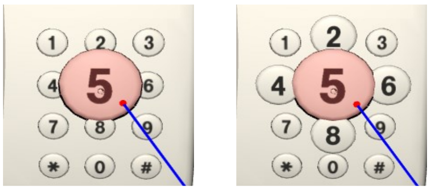
- Ray-based techniques:
  - Hand position + wrist orientation
  - Head position and hand direction
- Problems:
  - Visible objects may be occluded to the ray
    - Difficult to reach
    - Selection of objects needs to visit all of them
    - Sticky targets, enlarging objects, flatten regions…
  - Region selection not easy

3D SELECTION

3D SELECTION



3D SELECTION



INTERACTION DESIGN AND MEASURES

Pere-Pau Vázquez
Dept. Computer Science – UPC