How can we select targets faster?

(Practicing what we’ve learned)
Large Displays: Changing the Cursor Geometry T. Grossman & R. Balakrishnan, CHI 2005

- Area Cursor
  - Area cursor (e.g., square/circle) can make it easier to select a small object
    - But, what if multiple objects enter the cursor area?

- Bubble Cursor
  - Bubble cursor automatically changes size so that it encloses only the single closest object
    - Changes shape if needed to enclose a single target
  - Closest target computed from Voronoi diagram of objects

One of many techniques that automatically warp the position/shape/size of the cursor, dragged object, or potential targets

Large Displays: Automated Warping of Drag Targets P. Baudisch et al., Interact 2003

- Drag-and-pop
  - User starts to drag object
  - System creates proxies for potential targets in desired directions
    - Connected by “rubber band”
    - Animated closer to dragged object for Fitts’s Law “advantage”
  - User selects desired proxy
  - Faster than drag-and-drop when > 1 bezels crossed on multi-monitor wall

Issues
  - Warped targets are bunched up
  - Instant warp can be confusing

One of many techniques that automatically warp the position/shape/size of the cursor, dragged object, or potential targets

Large Displays: Manual Warping of Cursor and Drag Objects  
H. Benko & S. Feiner, CHI 2005

- Multi-Monitor Mouse
  - User can warp cursor and object being dragged to a different "frame" using a trigger
  - Pointer placement in new frame
    - Fixed location (e.g., center)
    - Frame relative
    - Frame dependent (e.g., last location)
- Trigger
  - Mouse or kbd button
  - Head orientation
  - Mouse location
- Users preferred frame relative, mouse button,
- Faster crossing > 1 bezels on desktop

http://www1.cs.columbia.edu/~benko/publications/2005/Benko_MultiMonMouse_CHI05_small.avi

Large Displays: Dealing with Seams  
J. Mackinlay & J. Heer, CHI 2004

- Seams between displays (e.g., bezels) can cause confusion → Make UI seam-aware
  - Take monitor geometry into account when drawing
  - Lay out objects to keep them from being obscured by seams
    - Line up arcs over seams
    - Don’t split nodes across seams

Seam-unaware Seam-aware
Stylus UIs: Crossing-Based Interfaces
J. Accot and S. Zhai, CHI 2002

- Replace *pointing at a target* with *crossing a goal*
  - Especially good for selecting thin objects
  - Recall the steering law for a fixed width straight tunnel: $MT = a + b \left(\frac{A}{W}\right)$, where $W$ is width of tunnel

Small Displays: Indicating Off-Screen Objects
P. Baudisch & R. Rosenholtz, CHI 2003; S. Gustafson et al., CHI 2008

- Small displays make it difficult to see off-screen objects
  - *Halo*: Surround object with circular “halo” arc just big enough to be visible
    - Halo location & curvature make it easy for user to infer object position
    - Used in Second Life maps
  - *Wedge*: Later work replaces the arc with a wedge whose off-screen tip is at the object
    - Wedges can be automatically rotated to avoid overlap, unlike arcs
      - Helps disambiguate close objects

Small Devices: Finger Input

Soft Keyboards: Key Presses

- Apple iPhone
  - QWERTY
  - Multi-touch
  - Entry on key release
  - Confirmation pop-ups to address occlusion
  - Dictionary used to
    - Correct misspellings
    - Correct mistypings by weighting keys based on proximity to touch areas
    - Change sizes of letter target zones based on initial substring

http://help.apple.com/iphone/10/#/iph3c50f96e

Small Devices: Finger Input

Soft Keyboards: Strokes

- SHARK (Shorthand-Aided Rapid Keyboarding, later commercialized as ShapeWriter), Swype
  - Stroke between keys, approximating words in dictionary
  - Recognition software resolves ambiguity, including missed keys


Swype (http://www.swype.com), also acquired by Nuance (http://www.nuance.com)

Not eyes-free

P. Kristensson and S. Zhai, UIST 2004
Small Devices: Finger Input
Soft Keyboards: Bimanual Strokes

- Bimanual gesture keyboard
  - Each hand strokes through letters on its half of the keyboard
- How to terminate a word?
  - Finger-release. Lifting both fingers off the screen ends the word
  - Space-required. Space key ends the word
- Users preferred finger-release
- Both approaches theoretically more efficient than unimanual stroke keyboards, but
  - Unimanual faster!
  - Bimanual required more mental effort
  - Bimanual more comfortable, less physically demanding

Not eyes-free

Small Devices: Pen Input
New Layouts, Strokes

- QWERTY
  - Conventional
  - ~ 30 wpm
- Metropolis
  - Optimized using Fitts’s Law, based on digram pairs, using random walk, simulated annealing
  - ~ 43 wpm
Small Devices: Pen Input Strokes, Bezel Constraints

- Edgewrite
  - Unistroke: One stroke per character
  - Enter characters by traversing edges and diagonals of a square hole in plastic template
  - Only sequence of corners traversed matters
  - Square hole enforces (easy-to-make) cardinal-direction gestures

Eyes-free and bump-resistant
- Use of strokes along template border makes it easy for users who are disabled or in motion