

# Assignment 6

## Task 1

a) The independent variables in this experiment are:

1. **Method of Off-screen Visualization (methodToUse):** This variable determines whether the off-screen pointing method used is "halo" or "wedge".
2. **Intrusion (intrusion):** This variable represents how much the off-screen visualization pokes into the screen.
3. **Distance of Targets (distances):** The distances at which targets are placed one of the given number of pixels away from the device edge.

b) The dependent variable in this experiment is dmc, which is the distance between (cx,cy) and (mx,my). This is the measured value that depends on the independent variables.

c) In a within-subjects design, attention must be paid to the arrangement of the variants (counter-balancing, ordering of the conditions) to avoid order effects. Order effects occur when the order in which the treatments are applied influences the outcome of the experiment. For example, a participant's performance may improve over time due to learning or worsen due to fatigue. Counterbalancing is used to control for order effects by varying the order of treatments between participants, so that each treatment is equally likely to be given in any order.

e) Participants' preference

Participants	Preferred technique
1	Halo
2	Halo
3	Wedge

f) Based on the graph, it appears that wedge technique might enable more precise localization than halo technique, particularly for objects farther away from the center. This is because the wedge technique extends outward from the edge of the viewport, indicating the object's location with a pointy end, whereas the halo technique uses a full circle around the edge of the viewport, and the growing size of the circle might obscure the exact location of distant objects.

The width of the visible area is likely to influence the result as well. A narrower viewport would necessitate using a larger halo or wedge to represent the same off-screen area, potentially reducing the precision for locating objects, especially those further from the center.

## Task 2

a) In real-world use, the "Shift" technique is more suitable due to its simplicity and ease of implementation compared to "Escape." Similar techniques applied in commercial mobile user interfaces include Fitts's Law optimization, gesture-based interactions, dynamic target

resizing, and adaptive UI layouts. These methods enhance target selection by optimizing target sizes, leveraging gestures, dynamically resizing targets, and adapting interface layouts to different contexts. Overall, "Shift" is preferred for its effectiveness in improving target selection while being straightforward to implement.

## **b) Bezel Input for Text Annotation in Mobile PDF Reader**

**Concept:** This design utilizes bezel gestures on the side of the mobile device (phone or tablet) for text annotations in PDF files. It leverages the underutilized bezel area and provides quick access to common annotation options while minimizing screen space usage.

### **Design:**

1. **Visual Cues:** Thin colored lines (yellow, green, red) are displayed along the bezel corresponding to the annotation options: Highlight (Yellow), Alt-Highlight (Green), Underline (Red). A fourth, white or transparent line can represent "Clear".
2. **Target Area:** A narrow strip (around 5-7mm) on the bezel next to the screen acts as the touch-sensitive area for initiating the bezel gesture.

### **How to Annotate:**

1. **Place Finger:** While viewing the PDF document, place your finger on the desired bezel line corresponding to the annotation you want to apply (e.g., yellow line for highlight).
2. **Swipe In:** With your finger still on the line, swipe it inwards onto the touchscreen briefly (around 2-3cm). This swipe gesture initiates the annotation.
3. **Drag (Optional):** For highlighting or underlining, you can then drag your finger across the text you want to annotate. Lifting your finger after dragging applies the annotation to the selected text.
4. **Clear Annotation:** Similar to highlighting/underlining, place your finger on the white/transparent line and swipe inwards to clear any existing annotations on the selected text.

### **Benefits:**

- **Quick Access:** Bezel gestures provide one-handed access to annotations without needing to switch tools or menus within the app.
- **Non-Intrusive:** The design keeps annotations options off the main screen, minimizing clutter and maximizing document viewing space.
- **Intuitive:** The color-coded lines and swipe gestures offer a clear and learnable interaction method.

### **Additional Considerations:**

- Haptic feedback can be added to confirm successful annotation application.
- The bezel input area can be customizable to match user preferences (e.g., left or right bezel).

- Advanced options like text strikethrough or adding comments can be included with a long-press on the bezel lines.

**c)**

- **Lack of Pressure Sensitivity:** Unlike traditional touchscreens, back-of-device interactions might not directly involve pressing the surface. This eliminates the ability to differentiate between a light tap and a firmer press.
- **Limited Area Recognition:** The two-state model often focuses on a single point of contact. Back-of-device interactions could involve touches in larger areas or even gestures like swipes or squeezes. These wouldn't be captured by a simple down-up model.
- **Context-Dependence:** Back-of-device touches might require additional context for interpretation. For example, a double tap on the back might mean a screenshot depending on the current activity. The two-state model wouldn't inherently capture this context.

**d) There's a constant trade-off between the size of tappable targets on touchscreens:**

- **Larger Targets:**
  - Pros: Easier to hit, reduces errors, especially for users with motor skill limitations or smaller screens.
  - Cons: Takes up more screen space, can lead to accidental touches on closely placed targets.
- **Smaller Targets:**
  - Pros: More screen space for content, cleaner interface.
  - Cons: Harder to hit, can lead to frustration and errors.

Finding the right balance depends on the target audience, the type of content, and the intended use of the application. For instance, a game might require larger targets for fast-paced actions, while a productivity app might prioritize a cleaner interface with smaller targets.