CS6750 Homework 2

Kangni Dang
kdang49@gatech.edu

1 QUESTION1

One task that has become almost invisible to me through learning is cooking with a cookbook and kitchen tools (e.g., measuring cups, timers, and thermometers). Initially, I spent considerable time thinking about every component of the interface. This included reading and re-reading the recipe instructions, constantly checking quantities with measuring tools, adjusting the heat on the stove, and keeping track of cooking times manually with a timer. Each step required focus, and the interfaces—like the cookbook's text and the physical tools—were always at the forefront of my attention.

My initial thought process: Measuring: I had to focus on finding the correct measuring cups and reading the recipe for precise quantities. Temperature Control: Adjusting the stove required conscious monitoring, watching how food responded, and using a thermometer to ensure the right doneness.

My current thought process: With repetition, many of these interactions have become second nature. I no longer need to focus on how to measure ingredients or check the stove heat constantly—I can "eyeball" quantities or adjust the heat based on experience. The timer is still necessary, but I intuitively know when something is nearly done. The cookbook itself has also faded into the background, as I remember many recipes or know how to modify them on the fly.

Redesigning the Interface: To make this interface more invisible more quickly, a smart cooking assistant app could provide proactive guidance. This app could sync with smart kitchen tools like scales, thermometers, and stove controls. Instead of manually measuring or adjusting, the app could suggest quantities, auto-adjust heat, and monitor temperatures through sensors. For timing, integrated timers based on the current cooking process could dynamically adjust, removing the need for constant manual setup. These features would speed up the learning process and reduce the need for constant attention.

2 QUESTION2

Here I choose using a smartphone to make video calls, including placing and receiving a call as the task domain. Here's how visual, auditory, and haptic perception are used to provide feedback to the user in this task.

How each of these 3 are used to give the user feedback: Visual Perception: Visual feedback is crucial when placing, receiving, and managing video calls. he interface shows buttons and icons (e.g., phonebook, call button) to initiate the call. During the call, users receive visual cues such as the timer indicating call duration, mute/unmute status, and connection strength through network signal icons. Visual feedback comes through the caller's name or number appearing on the screen, along with the option to accept or decline the call. Users can see the video feed of themselves and the person they're calling. When network issues arise, the app may display visual alerts like "Poor Connection" or pixelation of the video feed. Auditory Perception: Users can see the video feed of themselves and the person they're calling. When network issues arise, the app may display visual alerts like "Poor Connection" or pixelation of the video feed. Users can see the video feed of themselves and the person they're calling. When network issues arise, the app may display visual alerts like "Poor Connection" or pixelation of the video feed. Users can see the video feed of themselves and the person they're calling. When network issues arise, the app may display visual alerts like "Poor Connection" or pixelation of the video feed. Haptic Perception: Users can see the video feed of themselves and the person they're calling. When network issues arise, the app may display visual alerts like "Poor Connection" or pixelation of the video feed. Users can see the video feed of themselves and the person they're calling. When network issues arise, the app may display visual alerts like "Poor Connection" or pixelation of the video feed. Users can see the video feed of themselves and the person they're calling. When network issues arise, the app may display visual alerts like "Poor Connection" or pixelation of the video feed.

Design new ways of using visual, auditory, and haptic feedback for making video calls on a smartphone: Visual Feedback: A real-time facial expression gauge could provide feedback during a call. The system would use facial recognition to analyze the mood of the person you're talking to (e.g., happy, neutral, upset) and display small icons or subtle color changes around their video feed (green for positive, red for negative, blue for neutral). This would help users better understand non-verbal cues, especially in poor video quality or low light. Auditory Feedback: An audio quality assistant could provide real-time feedback about connection quality. Instead of only hearing audio distortions when the network is poor, the system could provide a subtle background tone, such as a soft chime, indicating a drop in audio quality. As the network improves, the chime fades, letting users know when to speak or pause, avoiding interrupted conversations. Haptic Feedback: Directional haptic cues could be used to notify users when their face is out of the camera frame. If the user moves too far left, a soft vibration could occur on the left side of the phone, guiding them back into view. Similarly, vibrations on the top or bottom of the device would encourage the user to adjust their position, ensuring optimal framing for the call.

A different kind of human perception is thermal perception (the sense of temperature). In the context of making video calls on a smartphone, thermal feedback could be used to indicate the phone’s battery or connection quality. For example, if the device starts to overheat due to prolonged use or poor connection, the phone could subtly warm up to alert the user that they might need to take action (e.g., end the call, switch to audio-only mode, or charge the device). This thermal change could serve as a physical reminder without interrupting the call.

3 QUESTION3

Tip 1: Emphasizing Essential Content While Minimizing Clutter

Violation: A restaurant menu app often violates this tip by overwhelming users with too many options, categories, and images on a single screen. Navigating through cluttered sections, such as appetizers, main courses, desserts, and special offers, can make it difficult for users to focus on choosing their meal.

Redesign: The app could prioritize essential content, such as popular or user-favorite dishes, and allow other categories to be collapsed until the user wants to explore them. Filters (e.g., dietary preferences) could further streamline the experience, helping users find what they want quickly, reducing cognitive overload.

Tip 2: Offloading Tasks from the User onto the Interface

Violation: A public transit ticketing machine requires users to manually select routes, tickets, and payment methods, often with unclear instructions, forcing them to navigate multiple steps under time pressure.

Redesign: The machine could automatically suggest ticket options based on common routes, or use location-based services to pre-fill destinations. Users could simply confirm or modify the suggestion rather than entering all the details manually. Integrating contactless payment options would further simplify the process, offloading many tasks to the system and making the experience more user-friendly and efficient.

4 QUESTION4

Original Interface: Car Stereo System

The car stereo interface, especially older models, is often intolerant of user errors. For example, changing the radio station or volume requires precise button presses on small, closely spaced buttons while driving. Accidentally pressing the wrong button can lead to changing the station, adjusting the wrong settings, or even causing distractions while driving. This is especially easy to do while focusing on the road, and the penalty is significant because it creates a dangerous driving environment, taking the driver's attention away from the road.

Redesign with Constraints:

Introducing constraints can limit the possibility of error. One way to do this is by disabling or restricting access to certain features while the car is in motion. For instance, detailed adjustments like browsing through menus for radio stations or settings could be locked until the car is stationary, preventing the driver from making complex interactions that require visual and mental attention while driving. Only simple, essential controls like volume adjustment would remain active.

Redesign with Improved Mappings:

Improved mappings would ensure that controls have an intuitive relationship with their functions. The stereo could be redesigned so that physical dials are mapped to specific functions, such as a large, easily accessible dial for volume and separate tactile buttons for switching between preset stations. This would make it easier for drivers to understand the system without looking at it, reducing the chance of accidental presses.

Redesign with Improved Affordances:

Improved affordances could involve touch-sensitive or voice-activated controls that give clear, tactile or auditory feedback. For example, larger, raised buttons with distinct textures for volume and stations would let users differentiate functions by feel. Additionally, voice commands could allow drivers to say "volume up" or "next station" to minimize physical interaction with the system, avoiding errors altogether.

These redesigns would improve the interface's error tolerance by making it more intuitive and safer for drivers.