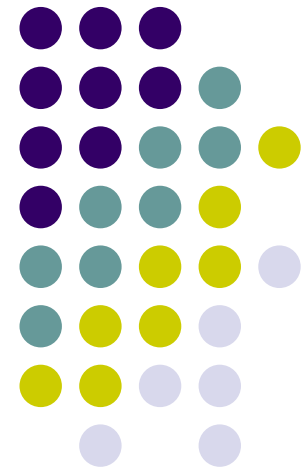


Remote Controller Revisited

Jeonghwa Yang, Eric Yang,
Sungkeun Park, John Gibby





Introduction

- Everyday device
- Interface between a user and an appliance
- However... many users still find it difficult to use them





Problem Statement

- Why is it difficult to use them?
 - Interface is too complex
 - Have too many functions not used frequently
- What about advanced functions?
 - Don't know how to use them
 - Satisfied with basic functions
- How can we improve it?
 - Make it simple: Most important functions only
 - Advanced functions should be still accessible

Requirements

- Simplicity
- All-in-one
- Comfortable Physicality
- Understandability



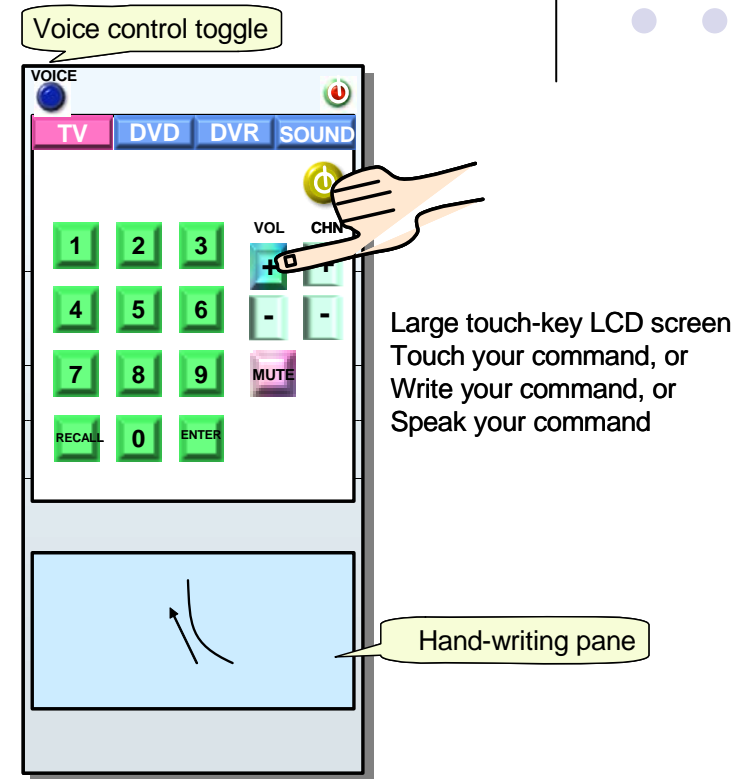
Design Alternatives



Holographic

Drawback:

- No physical presence
- Difficult to address multiple users
- High cost



Touch-write-speak

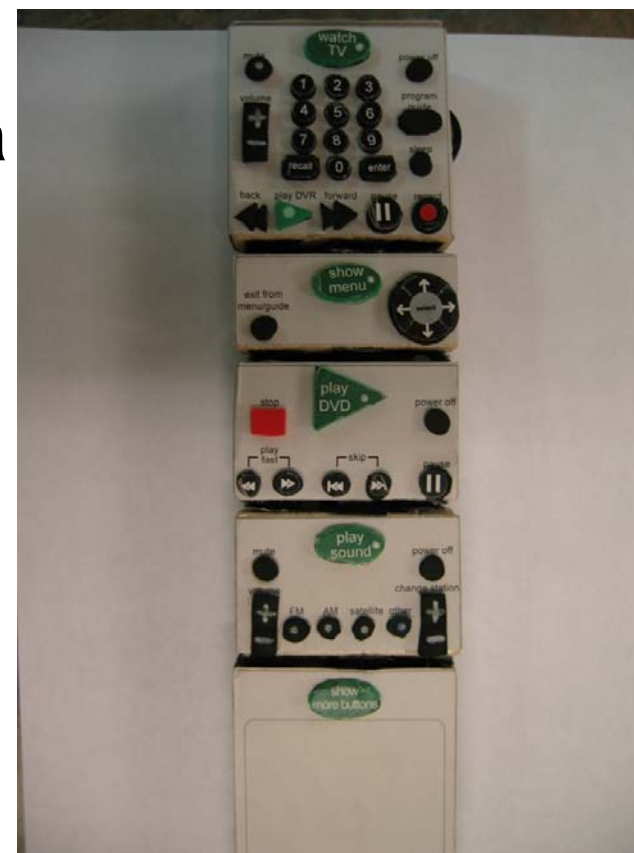
Drawback:

- Understandability issue
- Unfamiliar
- May not be intuitive to users
- Knowledge in the head



Modular: Our Final Choice

- All-in-one, modular design
 - One-to-one mapping between a module and an appliance (modeless)
- Extensive LED feedback
- Activity-centered design
- Minimal number of immediately visible controls
- Good physicality



Modular: Prototype



Module: TV with DVR



Module: common menu

Modular: Prototype



Module: DVD



Module: sound system

Modular: Prototype



Module: “show more buttons” (touch screen)

Evaluation: Cognitive Walkthrough



- Emphasizes learnability/understandability, a key requirement for us, and exploratory learning
 - Our users don't read manuals
- Allowed for design improvements before testing with users
- Helped us get past our familiarity and see from user's perspective (but not enough!)
 - Helps recognize possible error paths
- Tasks chosen:
 - 1. complex task dealing with DVD, TV, sound system
 - 2. simpler task dealing mostly with DVR

Evaluation: Cognitive Walkthrough



- Problems & suggestions
 - mute button functionality
 - “more functions” to “show more buttons” label
 - feedback from task buttons
- Disadvantages
 - Difficult to “think like a user”
 - After interview results, we saw our walkthroughs could’ve been more fruitful

Evaluation: Think Aloud, Structured Interview



- Think Aloud allowed us to hear users' thoughts and see their actions as they learned the interface – directly related to understandability
- Structured Interview complemented Think Aloud results

Users Selected



- We interviewed users convenient to us, but tried to have a good variety
 - 14 users
 - Ages: from 11 to 81
 - Mixture of male/female
 - Mixture of tech savvy v. non-gadget oriented
 - Various cultures



Evaluation Results

- Users liked...
 - Simple: Minimal number of buttons
 - Modular: Need only one remote control
 - Easy: Big Green button approach
 - LED feedback
- Users didn't like...
 - Physicality: size
 - Initial surprise: lack of power buttons
 - channel wheel
 - menu control module
 - Some labels: recall, enter, play fast, program guide
 - Sound system module needs CD functionality
 - Need to show radio station setting

With more resources



- Improve prototype fidelity
 - Lack of working LED's, operational channel wheel
 - “size” lesson: you can't make them believe the prototype is better than it is
 - Provide fidelity necessary to get the most valuable feedback
- Usability specs based on a representative range of available universal remote controllers
- Provide users a greater range of comparison devices and/or prototypes
- Wizard of Oz setup to make it appear that prototype is really controlling devices
- Evaluation experiment
 - More objective evaluation of various button layouts

Questions?





Other things we learned

- Problem definition: do enough research
- Make sure you really understand what a “requirement” is, and spend more time understanding user requirements
 - What, not how
 - User language
 - Duke was right! When you find the right requirements, there aren’t too many, and they really seem “right”
- Ditto for usability specifications
 - Helpful exercise to formulate them early, before finalizing a design



How we could have done better

- Make sure whole team reads off same evaluation methodology page
- Heuristic evaluation and left-handed users
- Use a heuristic evaluation customized for your particular requirements

How we could have done better

- Cognitive Walkthroughs:
 - the questions are powerful and insight-provoking...
 - But somewhat hard to understand; particularly the first one
- Try harder to answer the questions honestly, from the perspective of the user
- Use cognitive walkthrough to “wring out” a lot of problems with the design, even before the prototype stage (when it’s easier to change)





Evaluation Plan: Methods we didn't use

- Heuristic evaluation – we did this type of evaluation a lot when deciding between our three alternative designs
 - But, *expert* heuristic evaluation would've been helpful information
 - We used the 10-part heuristic eval from class, then repeated emphasizing our requirements
 - General heuristics lost sight of the key requirements
- Experiment
 - Would have required several team members to administer effectively
 - Task time was not a direct requirement
 - Would have been good, for objective comparison of prototype to existing products – number of errors made, speed of comprehension
 - Non-working prototype would complicate experiment and comparison with working products

Evaluation Plan – Usability Specifications



- **Simplicity:**
 - 50% less buttons than the control device
 - 80% of the users answer “yes” to question 1
- **All-in-one:**
 - replaces multiple remotes for the basic functions and 90% of the other functions.
- **Comfortable physicality:**
 - average of questions #4 and #5 > 3.7
 - < 15% of users report negative comment related to physicality
 - < 5% of users rate the design “1” on questions #4 and #5.
- **Understandability:**
 - average of question answers > 3.7
 - < 30% of users report negative understandability
 - < 10% “no” or “maybe not” cognitive walk-through answers
 - every button labeled, few/no multiple uses