VIM with EyeTracker

Future User Interfaces 2017

Kevin Schibli
Guillaume Pythoud
Soumaya El Hariri

Plan

- VIM (Vi IMproved)
- Modalities
- Tobii Eye Tracker 4C
- Pedal
- Hypothesis and evaluation
- Result
- Conclusion
- Demo

VIM (Vi IMproved)

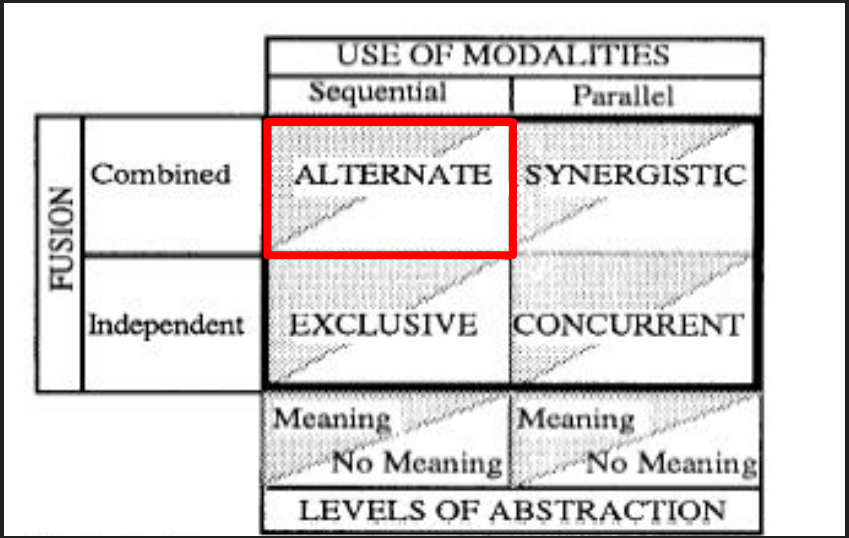
- Command line text editor
- Keyboard only
- Multiple modes:
  - Normal mode
  - Input mode
  - Visual mode
  - Etc.
- Movement + Command

Modalities

V.S.

Design space

CASE



CARE

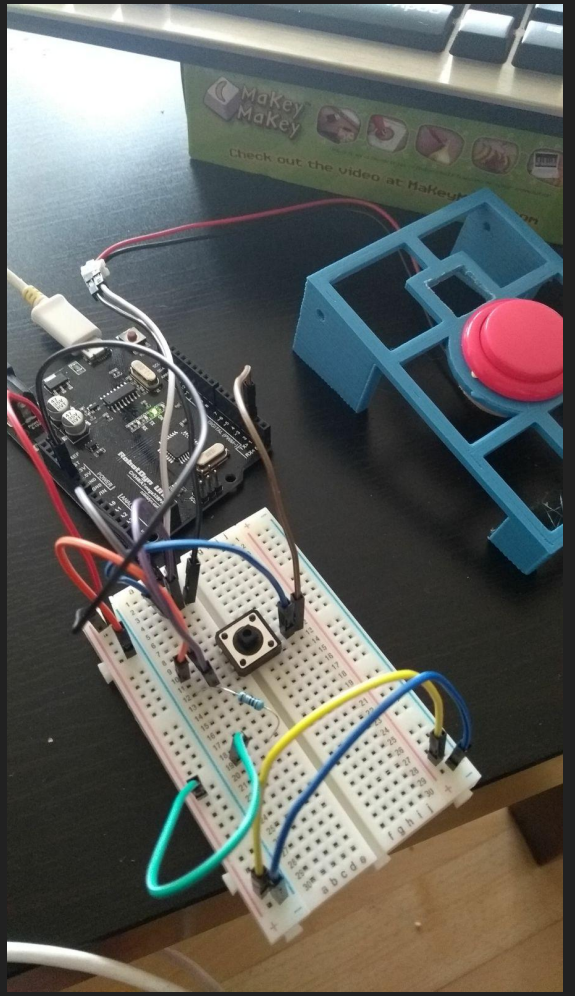
- **C**omplementarity
- **A**ssignment
- **R**edundancy
- **E**quivalence

Tobii Eye Tracker 4C

- Designed for gaming
- Real time tracking
- Gaze detection (no flickering)
- SDK available

Pedal

- *Arduino* that sends “on” when pedal is pushed and “off” when it is released through serial



Hypothesis

H1: It’s faster to edit source code with the eye tracker, pedal and keyboard than just with mouse and keyboard

H0: It takes the same time to edit source code with the eye tracker, pedal and keyboard or with mouse and keyboard

Variables

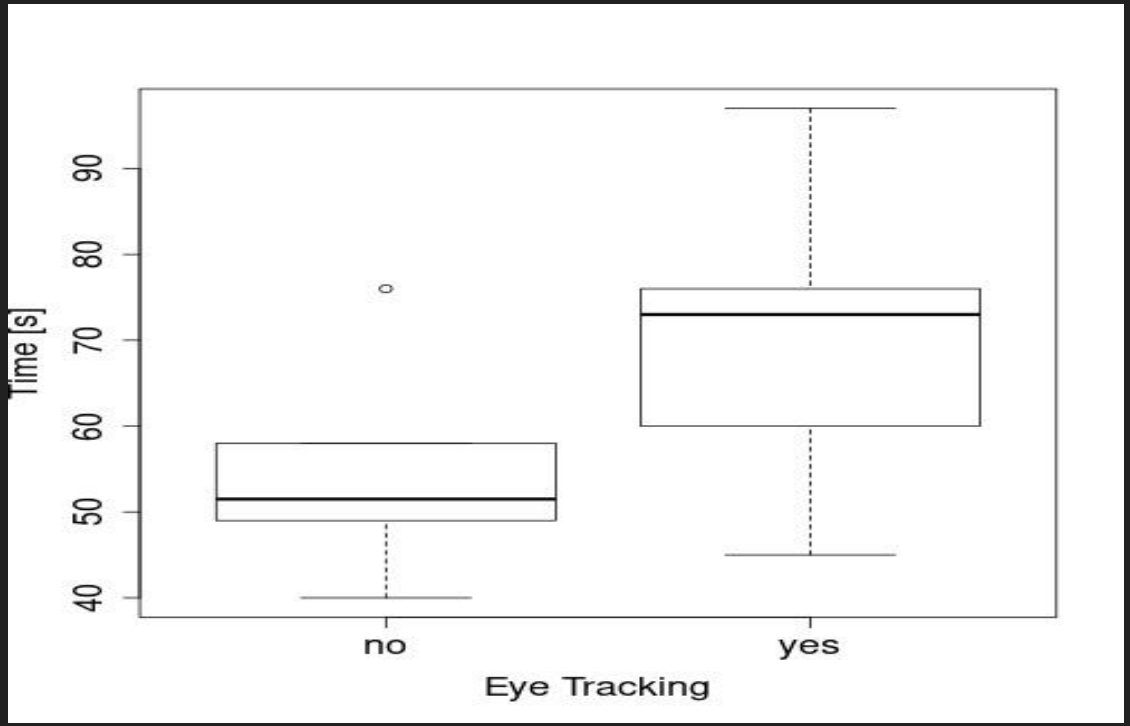
- Dependent variable: time to complete the task on the script

- Independent variables: the lines of code, the number of changes to make, the hardware

Evaluation

- Within group
- Subjects already know to use Vim
- 2 different files of Java code
- Keyboard, pedal and eye tracker VS keyboard and mouse
- 6 changes to make
- 6 subjects

Result



T test:
t = -1.8857,
df = 8.9375,
p-value = 0.09219

Means
No = 54s
Yes = 71s

Conclusion

- H0 is accepted
- Not significant, probably due to the low number of subject
- Keyboard and mouse is quicker but
  - Subjects never use eye tracker
  - Eye track precision was not perfect
  - Other factors: vim skills, apprehension of the task, typing speed
- The subject still appreciate using the eye tracker as a modality

Demo

Questions