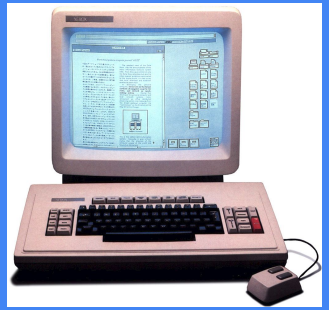
Input Models

Jorge Garza & Janet Johnson
COGS 230 / CSE 216

User Technology: From Pointing to Pondering



Stu Card

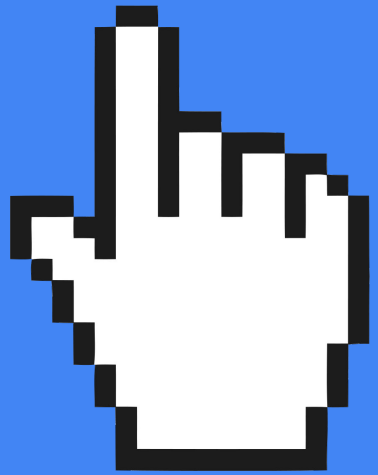
Thomas Moran

User technology and Pointing devices

“Understand The Personal part of personal workstations” --(Stuart & Thomas)

- The Physical Interface
- The Cognitive Interface
- The Conceptual Interface
- The Task Interface

The Physical Interface: Pointing



Click Me

Click Me

Fitt’s Law

$$ID = \log_2(D/W)$$

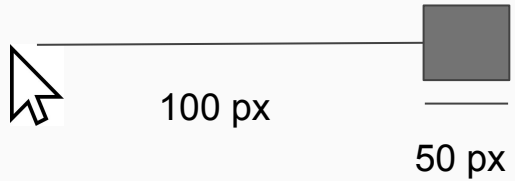
ID: Index of Difficulty

D: Distance

W: Width

Example

- Distance = 100 pixels, Target Width = 50 pixels

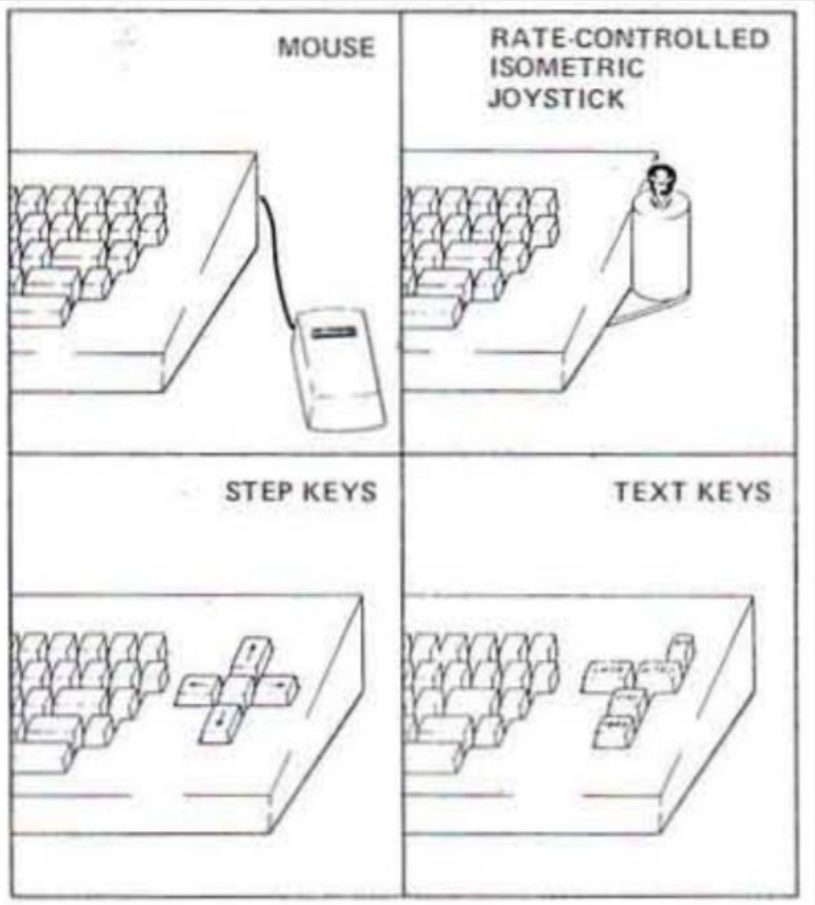


ID = log ((2*100)/50) = **0.60**

- Distance = 200 pixels, Target Width = 100 pixels

ID = log ((2*200)/100) = **0.60**

Which pointing device is better?



Time for Demo!

Which one is Better?

Demo-> https://guardado.github.io/FittsLaw_CSE216/

Results on increasing Target:

Distance

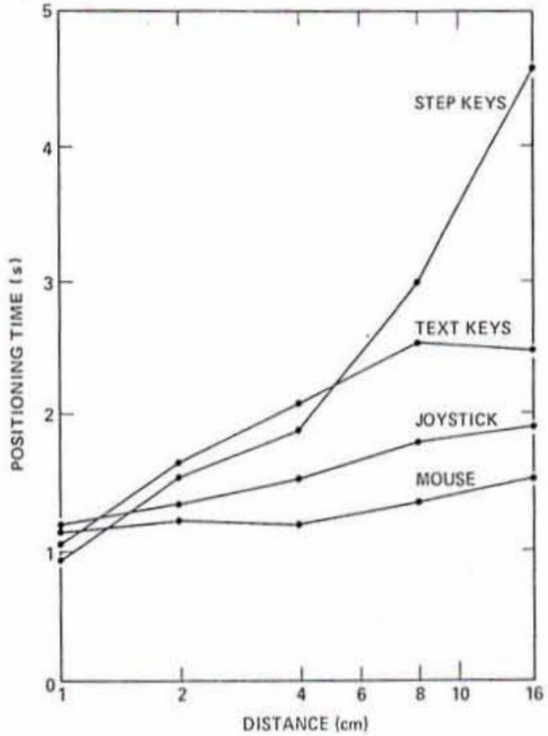


Figure 3. Effect of target distance on positioning time.

Size

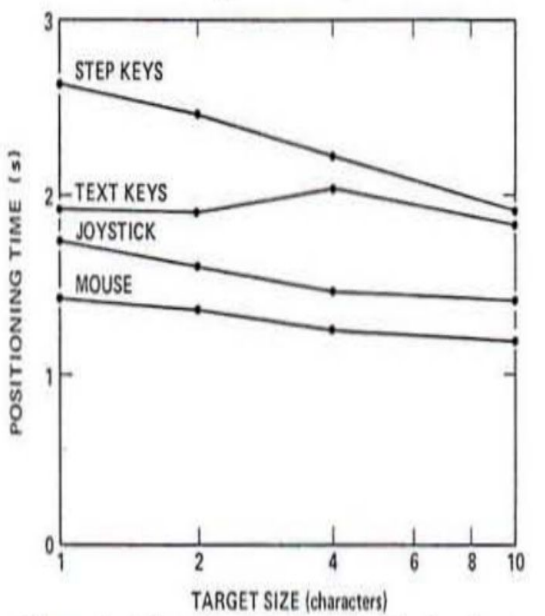


Figure 4. Effect of target size on positioning time.

Error

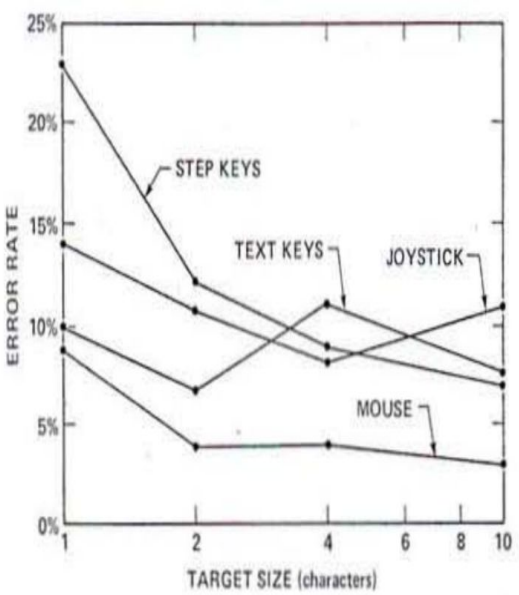


Figure 5. Effect of target size on error rate.

What’s the advantage of the pointing stick?

Fitt’s law

$$T = a + b\ (\ ID + 1)$$

$$T = a + b \log\ (D\ /\ S + c)$$

T : Time to complete the movement.
a and b: constants that depend on the choice of input device and are usually determined empirically by regression analysis.

Bits of different devices.

Device	Study	IP (bits/s)
Hand	Fitts (1954)	10.6
Mouse	Card, English, & Burr (1978)	10.4
Joystick	Card, English, & Burr (1978)	5.0
Trackball	Epps (1986)	2.9
Touchpad	Epps (1986)	1.6
Eyetracker	Ware & Mikaelian (1987)	13.7

Reference:
MacKenzie, I. Fitts' Law as a research and design tool in human computer interaction. Human Computer Interaction, 1992, Vol. 7, pp. 91-139

Where else you can find Fitts’s Law?

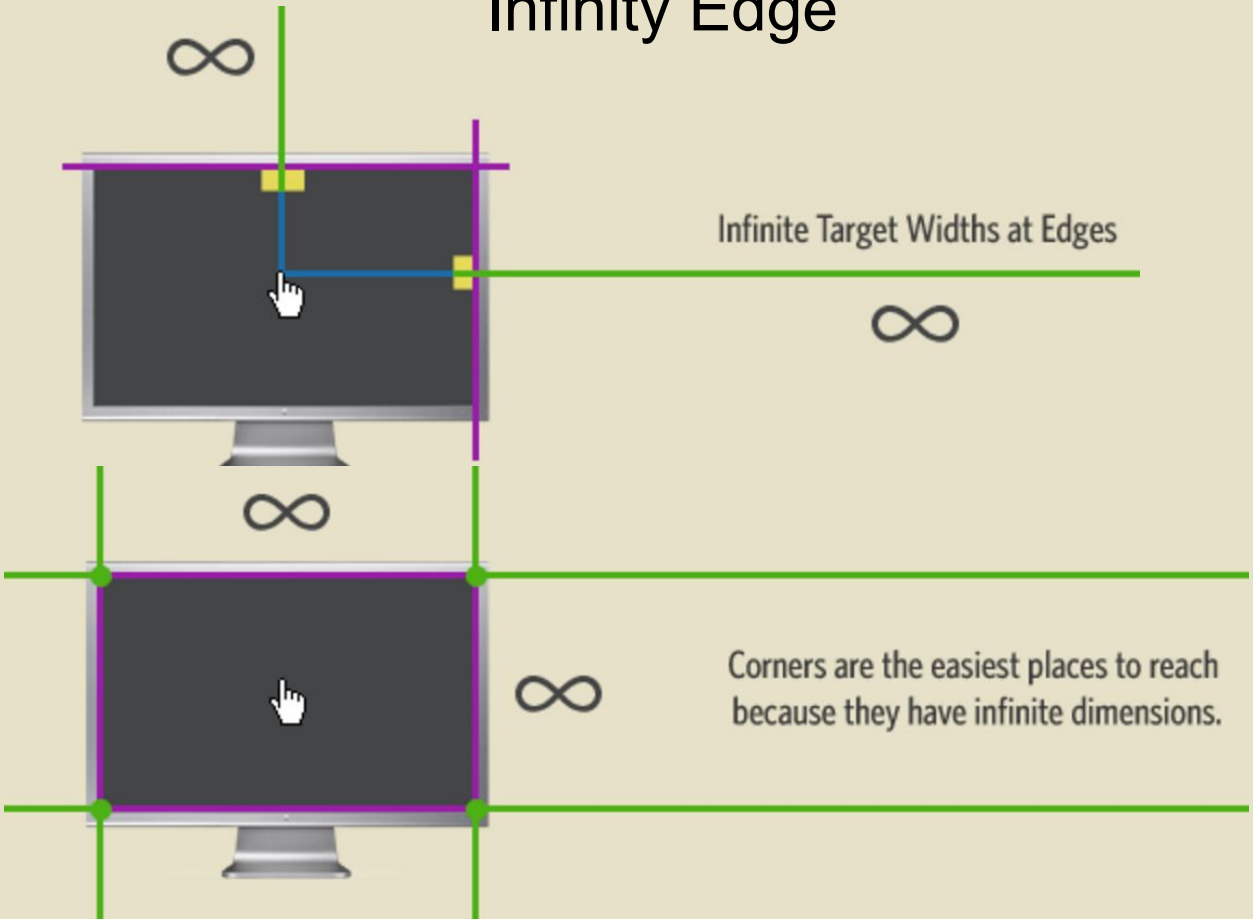

esc
F1
F2
F3
F4
F5
F6
F7
F8
F9
F10
F11
F12
delete
tab
Q
W
E
R
T
Y
U
I
O
P
caps lock
A
S
D
F
G
H
J
K
L
enter
return
shift
Z
X
C
V
B
N
M
shift
fn
control
option
command
command
option


START
STOP
STOP

Can you think of scenarios that don’t fully obey Fitts’s Law?

Fitt's Law: Worth Noting

Infinity Edge

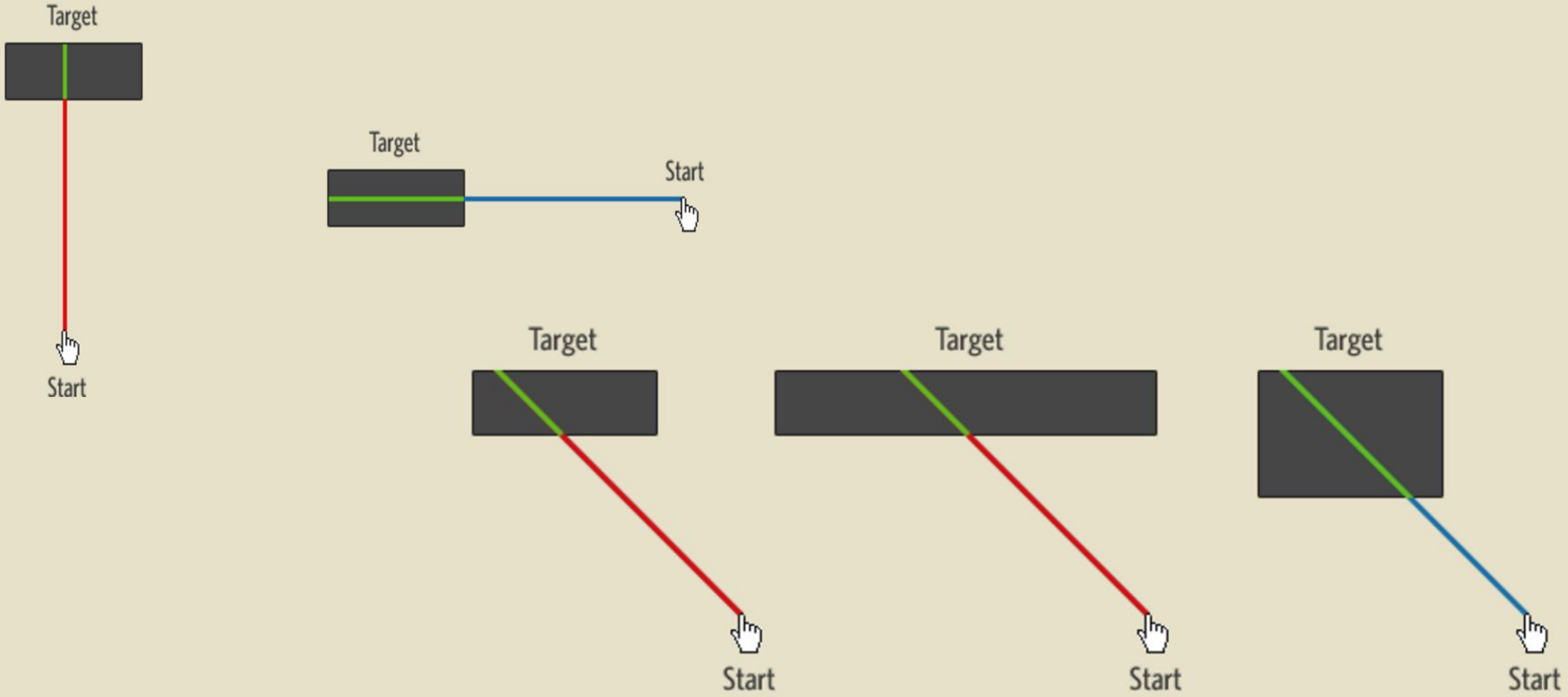


Fitt's Law: Worth Noting

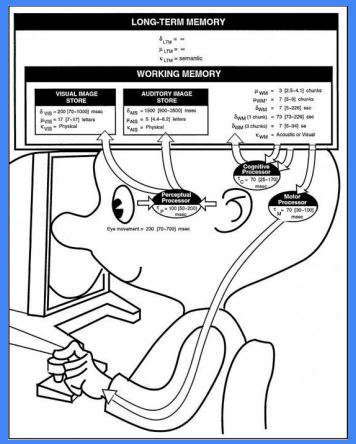
Physical vs Virtual Pointing

Fitt's Law: Limitations?

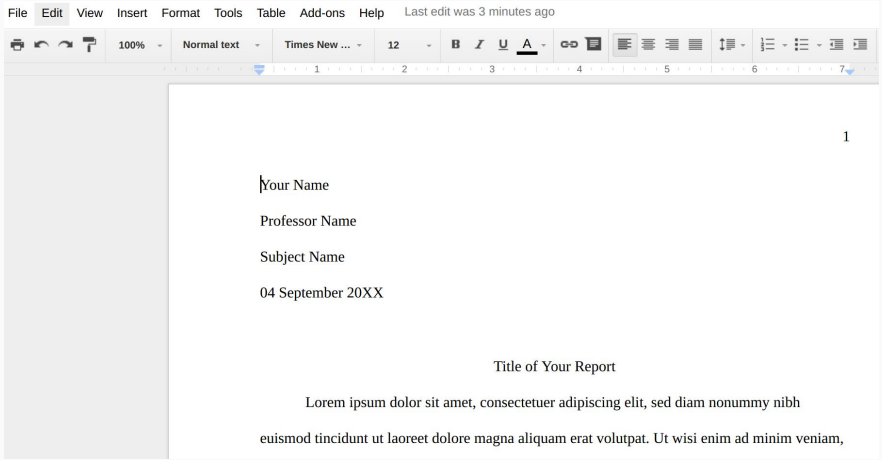
It's Made of Lines



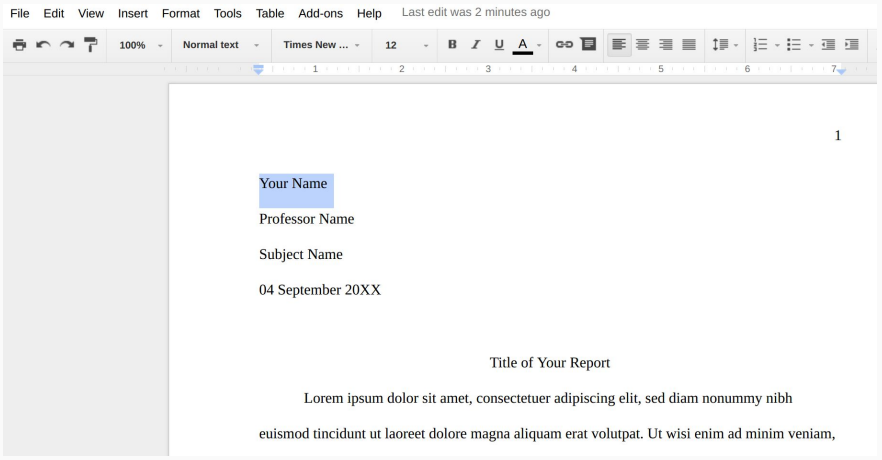
The Cognitive Interface: Cognitive Skills



Which text editor is better?



No selection available.



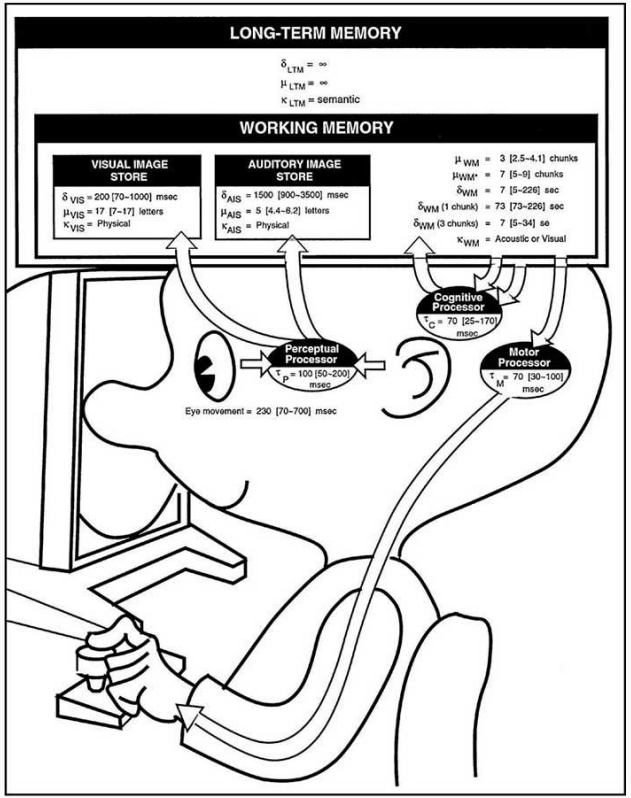
Selection available.

How long will take the user to edit a text?

What are the methods the user will choose to edit the text?

The Human Lag....

The Model Human Processor



Sense: Perceptual Processor

↓

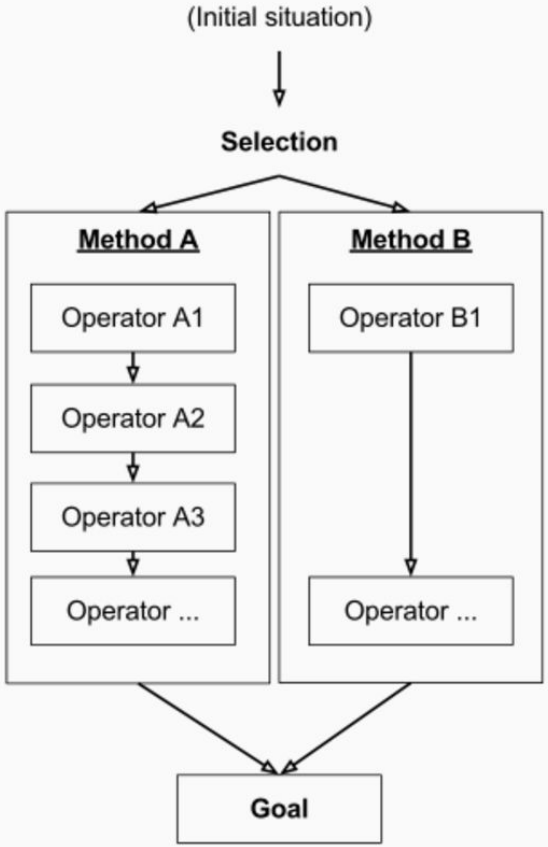
Think: Cognitive Processor

↓

Act: Motor Processor

Bounded rationality *- "Users acts are limited by their knowledge and processing ability."*

GOMS



What will the user do?

“Cognitive Skills”
(Previous Experience)

- Familiar **Goals**
- Skills you can do **(Operators)**
- Repertory of **Methods** you have use before.
- **Selection** of Methods

Keystroke-Level Model

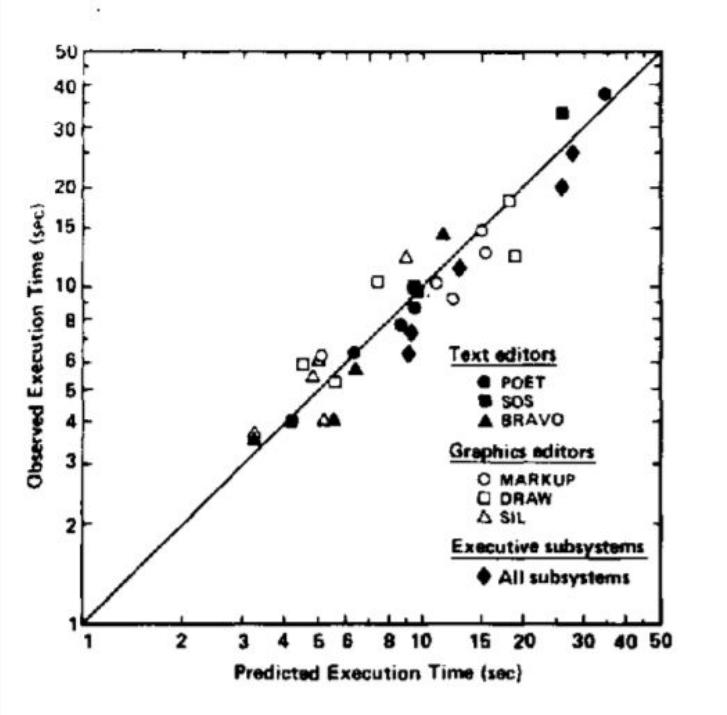
Only **O**perations and **M**ethods part of (G**OM**S).

KLM - *Operators*

- Operators
  - **K** Press a **K**ey or button, (0.08 to 1.20) secs
  - **P** **P**oint with mouse, (1.1) secs ← **Fitts' Law**
  - **H** **H**ome hands to keyboard or peripheral device (0.4)
  - **D** **D**raw line segments (0.9*n + 0.16*L) secs
  - **M** **M**ental preparation (1.35) secs
  - **R** System **R**esponse (system/activity dependent)

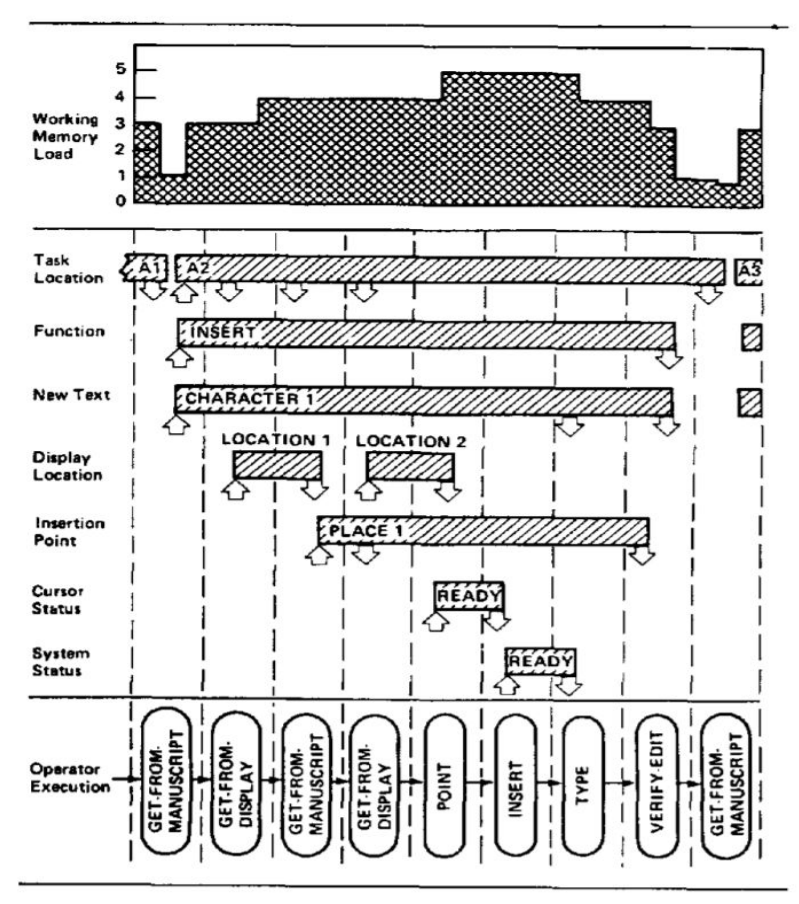
- You can Evaluate Time of different Methods.
- Keystrokes and Mouse Time are constants
- Intended for Designers.

Keystroke-level model predictions vs Observed Times



“Several schemes for selecting text in the Star text editor were proposed” -- (Stuart & Thomas)

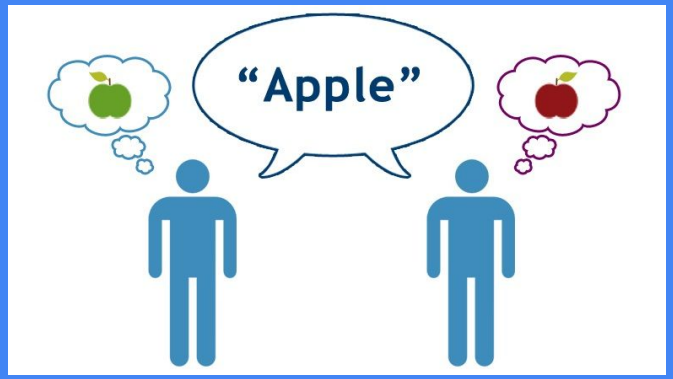
Working Memory Load and Unit Task



- User divides tasks into smaller tasks called “**Unit-tasks**”

- User performance is degraded as the **Working Memory Load** increases.

The Conceptual Interface: Mental Models.

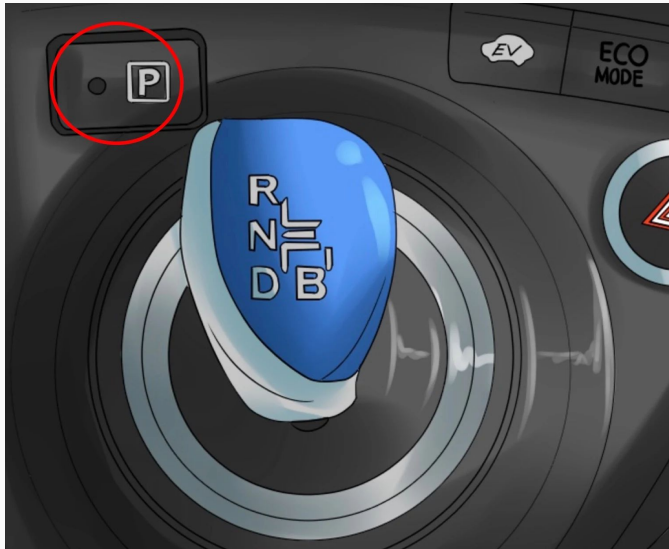


Mental Models

User model vs **intended user model**? Examples?

“I think AI is an area where user model and intended user model differ very much. Nowadays AI and ML are used like buzzwords, they're present in almost every single system one could imagine - so much that companies are now taking advantage of this fact. I as a company could say that my washing machine is AI powered and it washes the clothes using super smart algorithms. An example "intended" user model here could probably be to create a feeling of smart power utilization. Instead, mental model a layman could formulate would be that the machine is self sufficient and an all-knowing beast in the cloth washing department.” --- Tushar Koul

“One example I can give to elaborate the topic is the Toyota Prius gear shifts. the gear shift, the parking button is on the dashboard of the car instead of the stick. The designer’s model (intended user model) would be that the user clicks that button to park the car. However, someone who is new to prius but has been driving since a long time will expect the parking mode to be on the gear shift.” --- Rahul Ramath



Solving Math Problems with a calculator

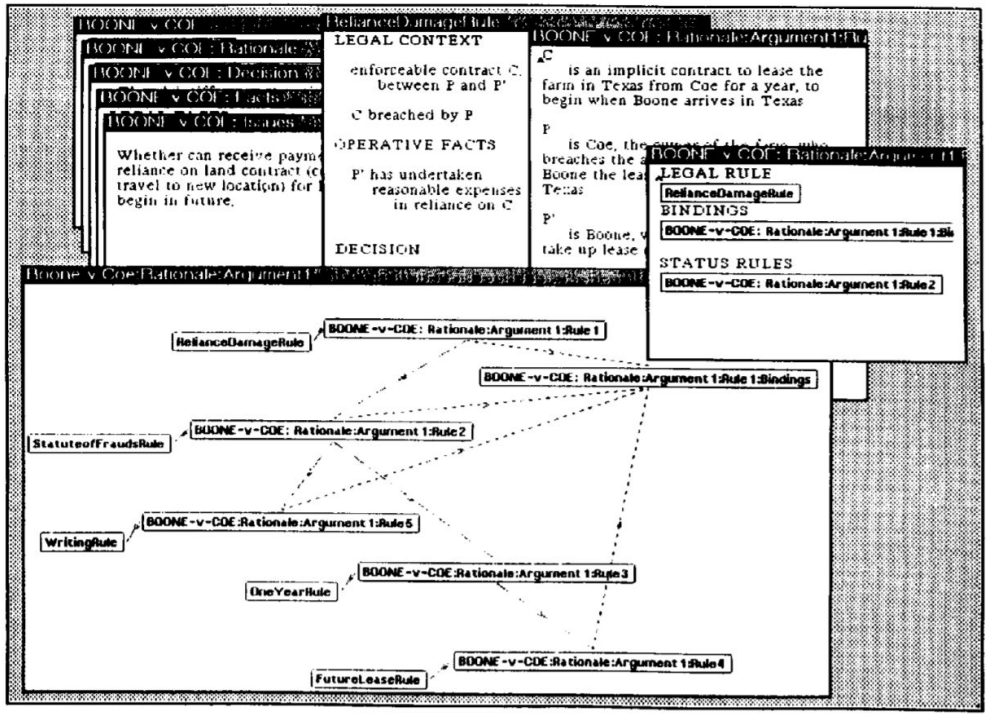
Problem Type	No-Model Users	Model Users
Routine	98	95
Complex	87	88
Invention	25	67

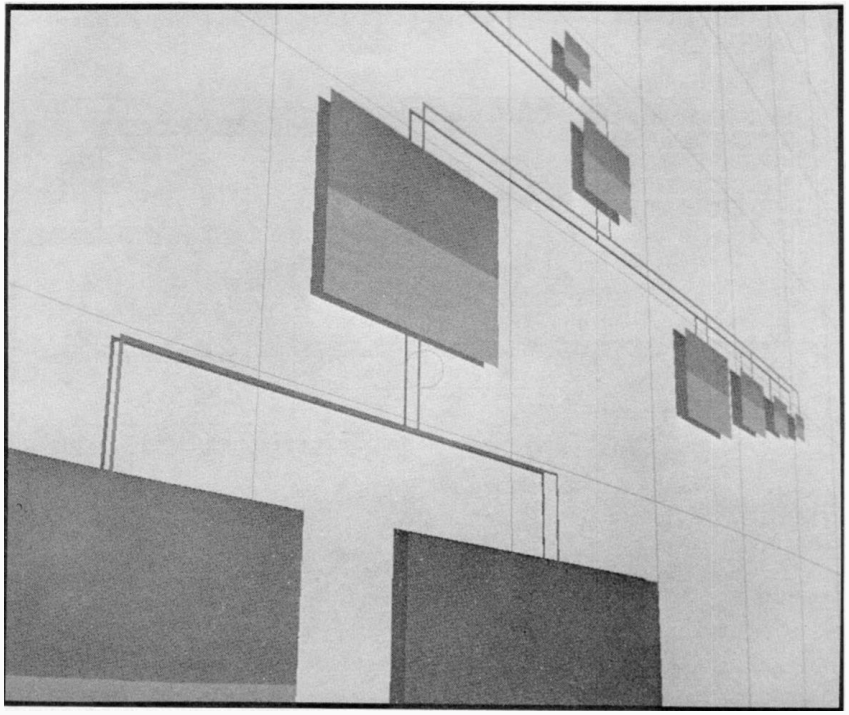
“Keep designs simple, consistent, and clear enough for users to grasp” -- (Stuart & Thomas)

The Task Interface: Pondering Ideas.

Pondering Ideas : Idea Browsing

“Aid users in structuring and Manipulating ideas”



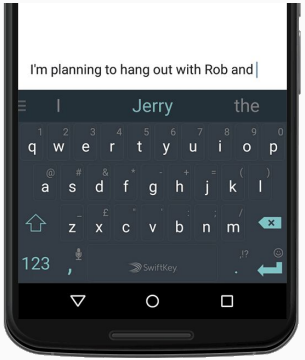


Limitations of "NoteCards" system vs the Experimental Dandelris browser?.

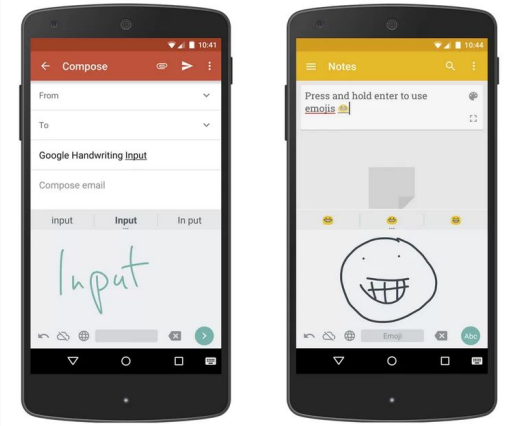
The Word-Gesture Keyboard

How did we get here?
Keyboard Types

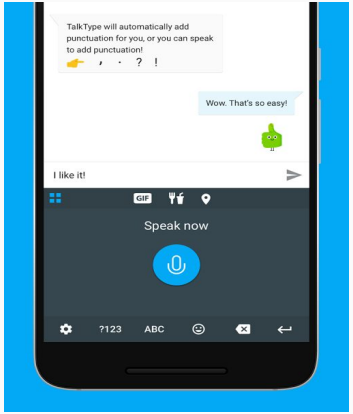
Physical Keyboards



Virtual Keyboards



Handwriting Recognition



Voice Recognition

How did we get here?
Montgomery's keyboard

- Make use of the ability to use a continuous stroke instead of just a striking key gesture.

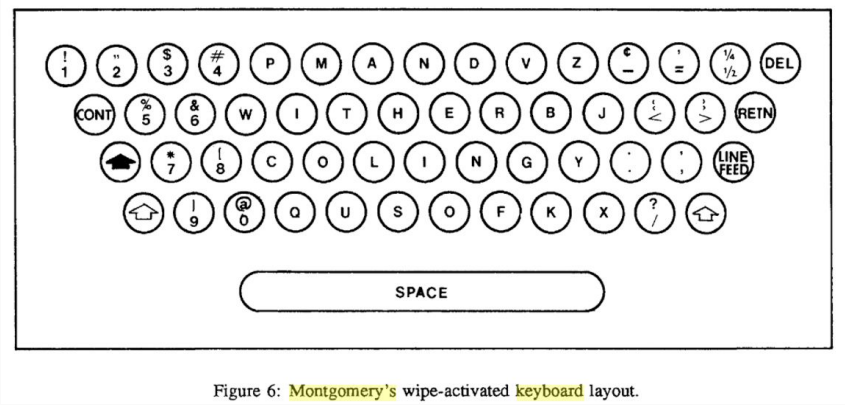


Figure 6: Montgomery's wipe-activated keyboard layout.

How did we get here?
The Metropolis Keyboard (2000)

- Use Fitts-Digraph-Energy

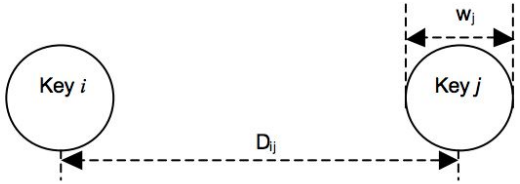


Figure 2. For given key size and distance, Fitts' law predicts the time of tapping from one key to another

Figure 7. (Part of) the dynamic simulation model: frequent digraphs are connected with stronger springs

- Optimization: Metropolis algorithm

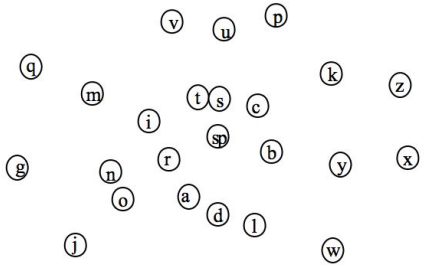


Figure 9. Early stage of Metropolis random walk. The system is at a high energy state, moving towards lower energy state.

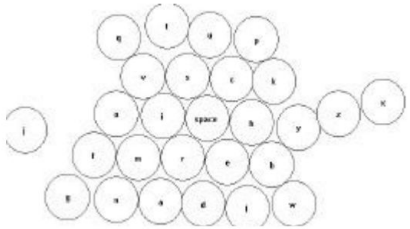


Figure 10. Middle stage of Metropolis random walk. Keys have been descended to a lower energy state. They are getting packed.

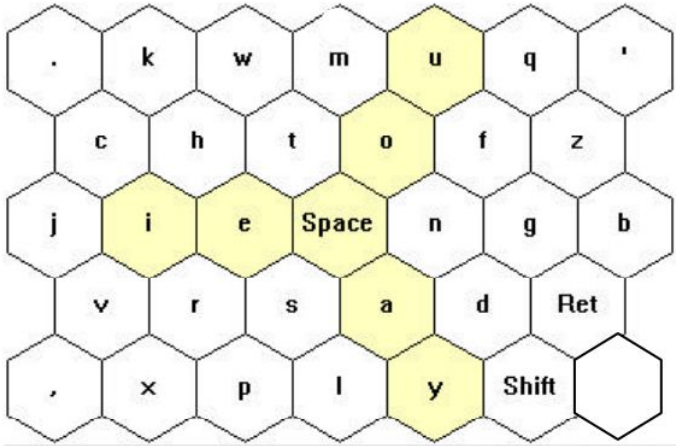


Figure 12. The Metropolis keyboard (43.1 wpm)

Considerations/Factors for text-entry methods

- Speed
- Cognitive load
- Ease of developing proficiency

Word-Gesture Keyboard: What is it?

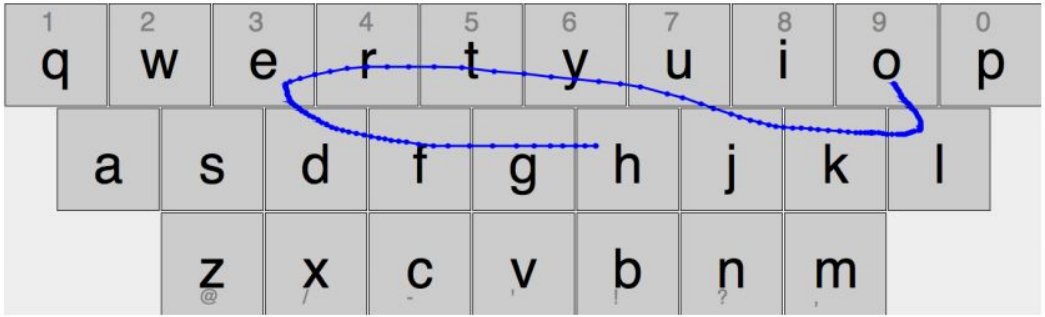
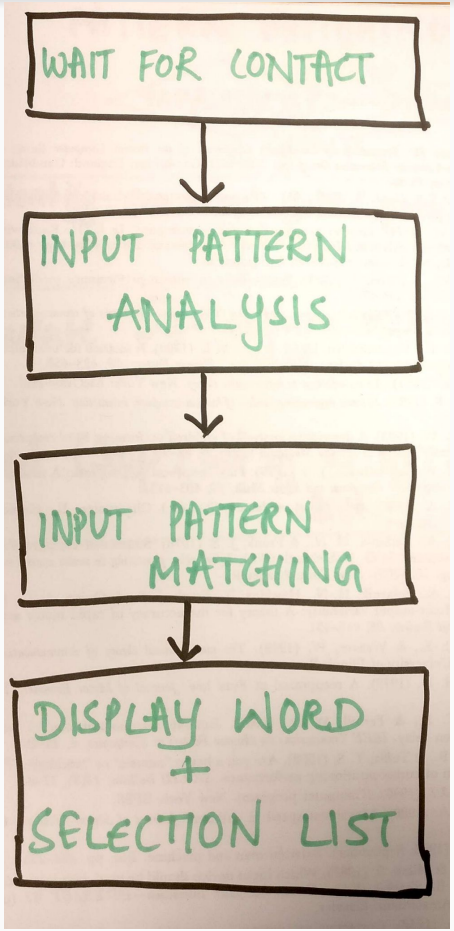


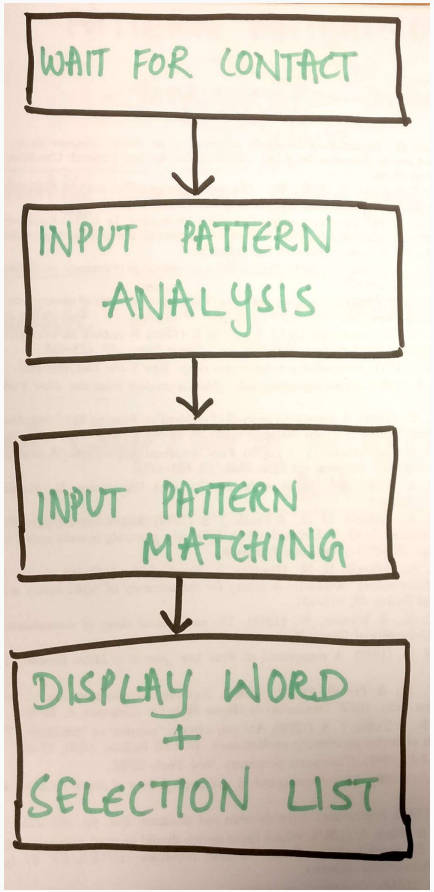
Figure 6: Swipe path of the word 'hello'

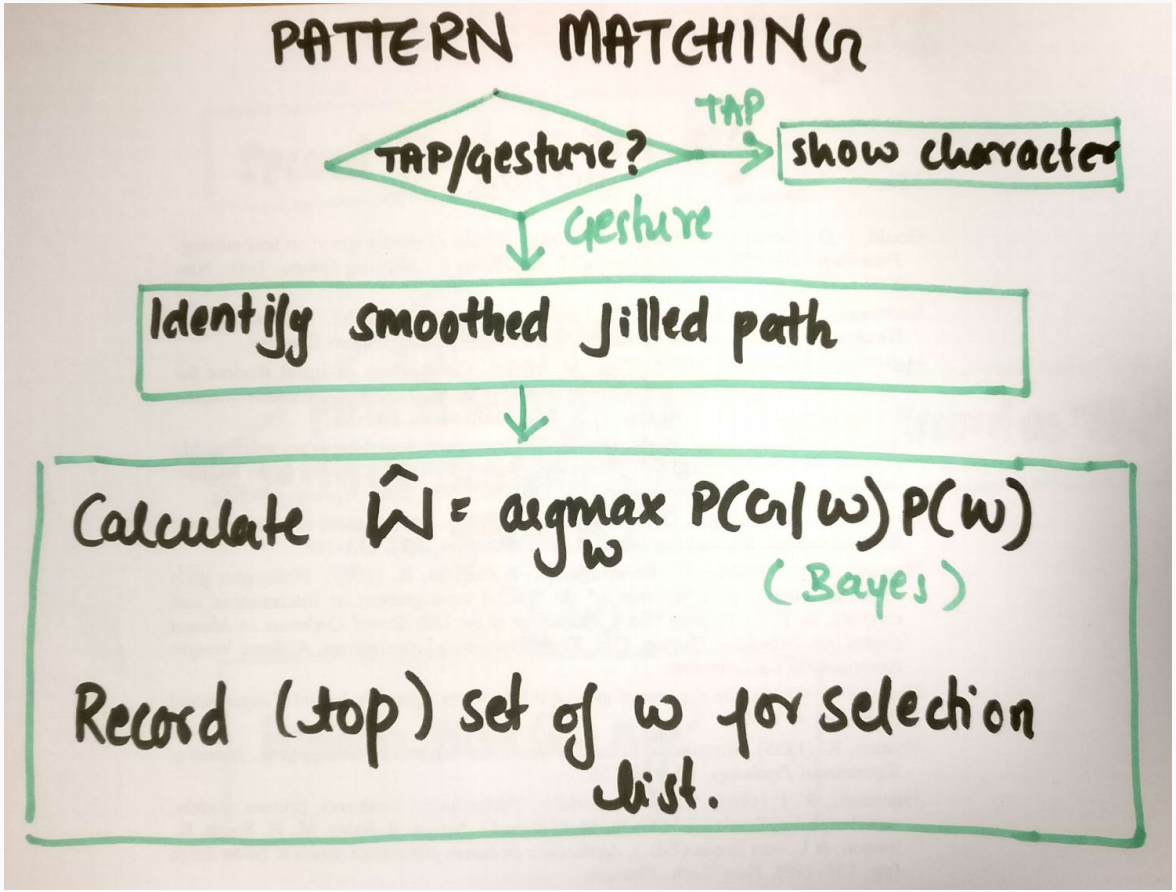
Features a qwerty keyboard but instead of having to tap every character to enter a word the user 'swipes' a finger or stylus across the characters of the word.

Word-Gesture Keyboard: How does it work?



Word-Gesture Keyboard: How does it work?





Word-Gesture Keyboard: How does it work?

Total time to write word W

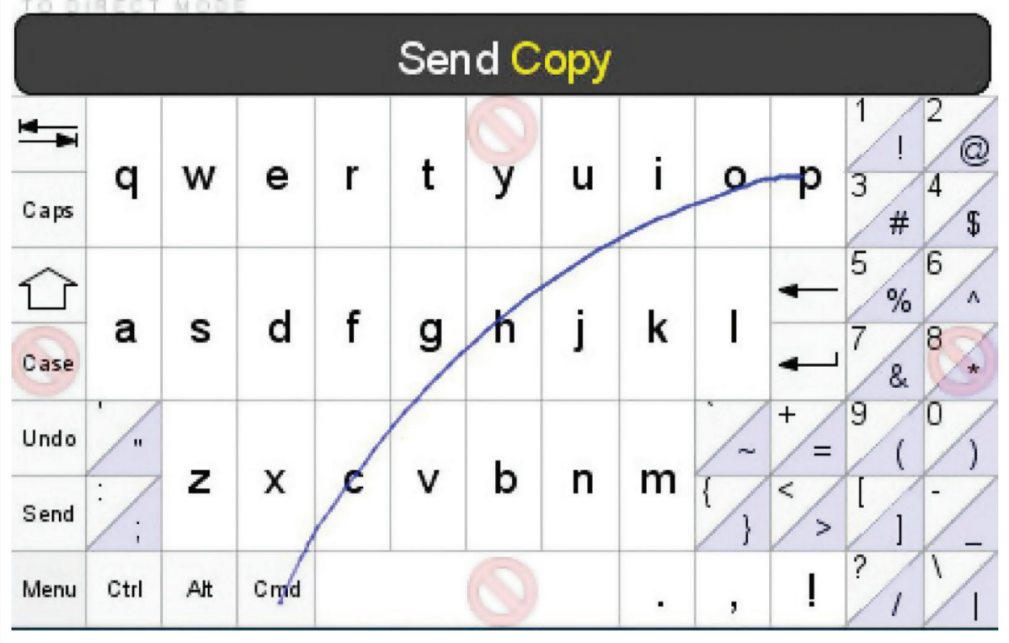
$$t_n(W) = Na + b \sum_{k=1}^{N-1} log_2 \left(\frac{D_{k,k-1}}{S} + 1 \right)$$

Considerations/Factors for text-entry methods

- Speed
- Cognitive load/Ease of use
- Developing proficiency/Ease to efficiency

Little tweaks / Novel Functions

- Command Strokes
- Case Key
- Auto ending/spacing



Given the success of gesture-based keyboards and that of eye-tracking as an input method - could combining the two provide a superior text input system? Why? Why not?

Thank You