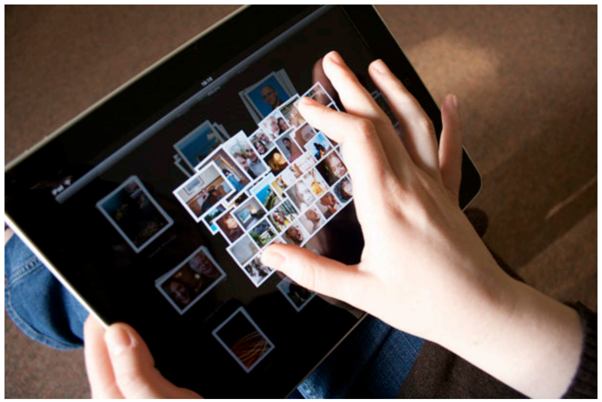
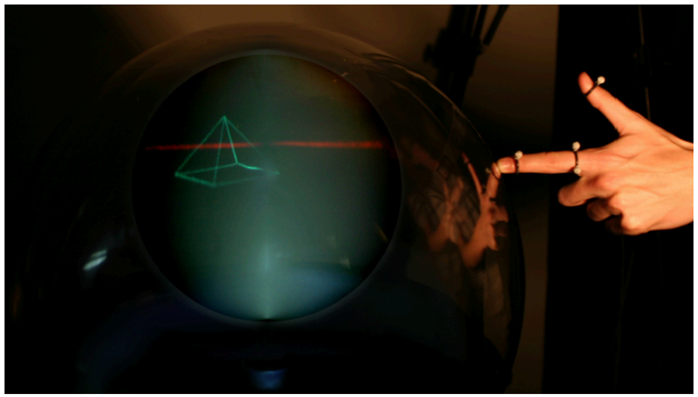
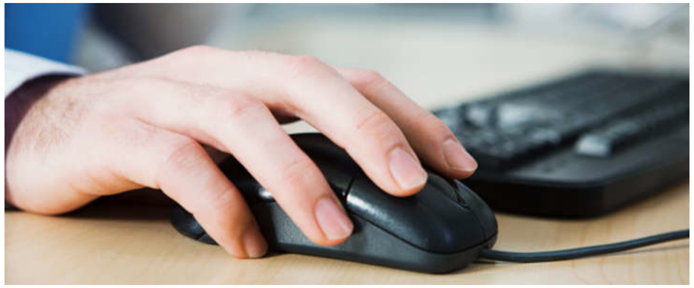
Input: pointing devices, input-output mappings, CD gain, mid-air interaction, problems of direct input and solutions

Input devices vs. Finger-based input

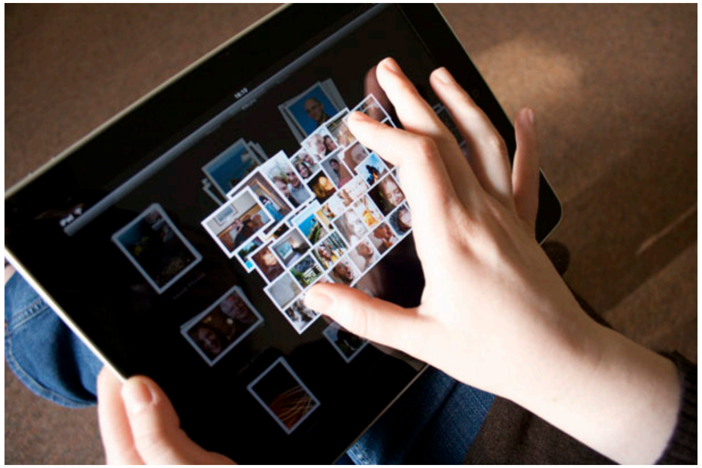




Indirect vs. Direct pointing



Indirect: The position of the cursor is controlled by the device



Direct: Fingers manipulate visual objects directly on the screen

Absolute vs. Relative pointing

Absolute: 1-to-1 mapping between input and output space

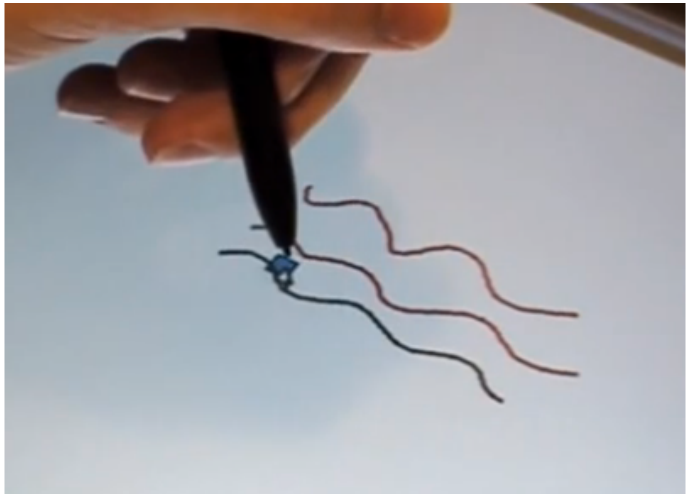
indirect

direct

Relative: Input controls the relative position of the cursor (always indirect)

Hovering mode

Tracking the position of the pointing device (e.g., the pen) or the finger from distance



Hover widgets http://www.youtube.com/watch?v=KRXfaZ8nqZM

Absolute pointing

Direct input

- Hovering feedback is not indispensable as there is a clear mapping between pen/fingers and the screen
- Main drawback: occlusion problems

Wacom Cintiq

Indirect input

- « Hovering » is indispensable: users must know the position of the cursor before starting drawing

regular graphics tablet

Relative pointing

Common devices: mouse and touchpad

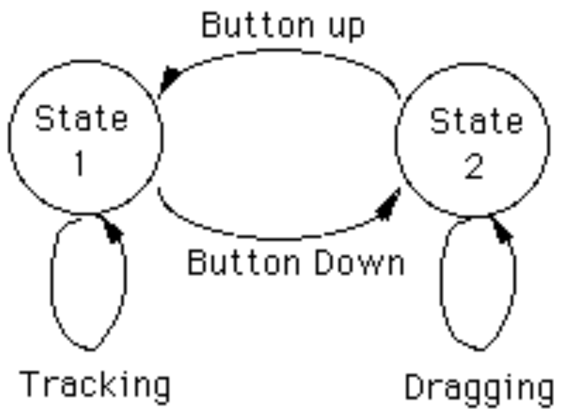
« Clutching » instead of « hovering » mode

- Lift the mouse or finger to « re-calibrate » movement
- Use of smaller input space to traverse a larger output space

How would you map the input space of the tablet to the output space of the wall?

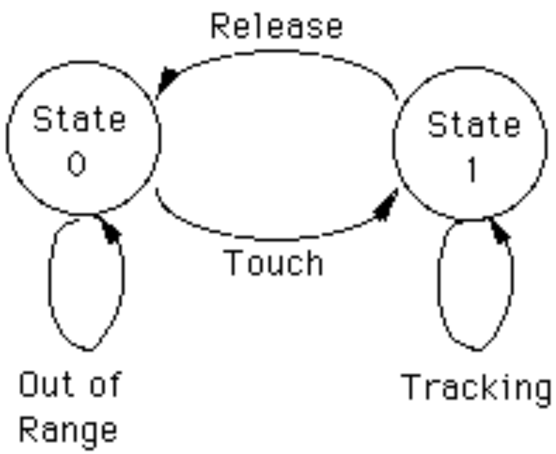
Smarties: https://www.lri.fr/~chapuis/publications/CHI14-smartiestk.mp4

Buxton's 3-state model (1990)



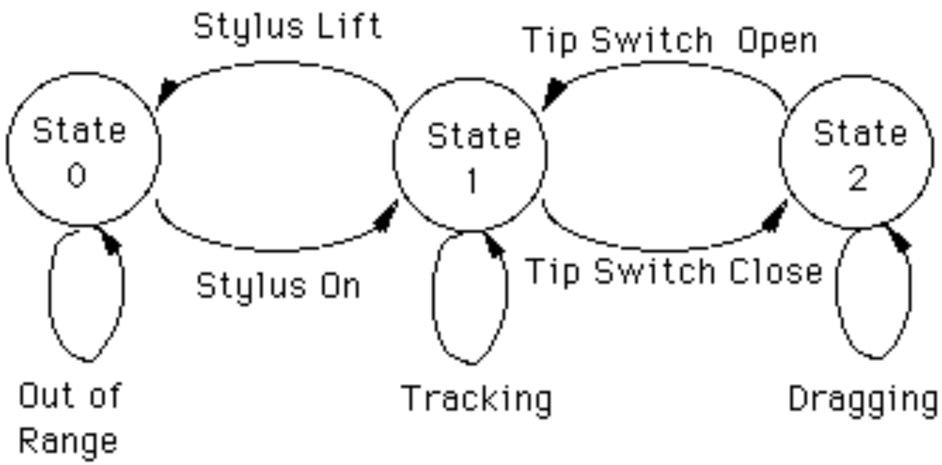
A. Two-state model for mouse

Buxton's 3-state model (1990)



B. Two-state model for a touch tablet

Buxton's 3-state model (1990)



C. Three-state model for a gaphics tablet with stylus

Relative pointing: Mappings

Position control: maps human input to the position of the cursor (or object of interest)

Examples: mouse, touchpad

Rate (or velocity) control: maps human input to the velocity of the cursor (or object of interest)

Examples: joystick, trackpoint

Trackpoint

Isotonic vs. Isometric devices

Iso*tonic* (iso-tonic = equal tension/force):
Absence of resistance, free movement

- Mouse, pen, human arms, etc.

Iso*metric* (iso-metric = equal measure):
Absence of movement, resistance as we press

Isotonic vs. Isometric devices

Iso*tonic* (iso-tonic = equal tension/force):
Absence of resistance, free movement

- Mouse, pen, human arms, etc.

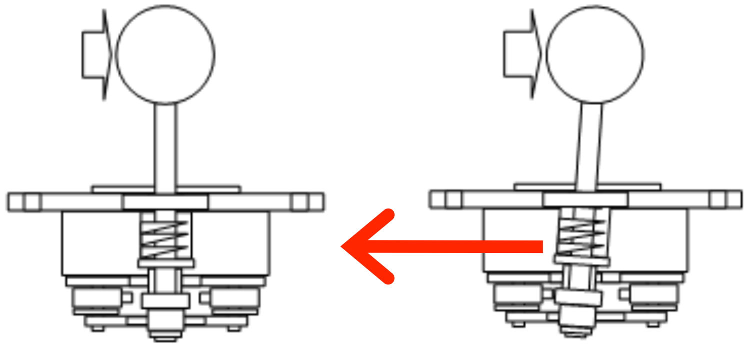
Iso*metric* (iso-metric = equal measure):
Absence of movement, resistance as we press

Elastic: Resistance increases with movement

- Joystick, trackpoint

Elastic/Isometric devices

There is a neutral position



As we apply force, an opposing force develops

Self-calibration: I we free the device, the opposing force bring the device to its neutral position

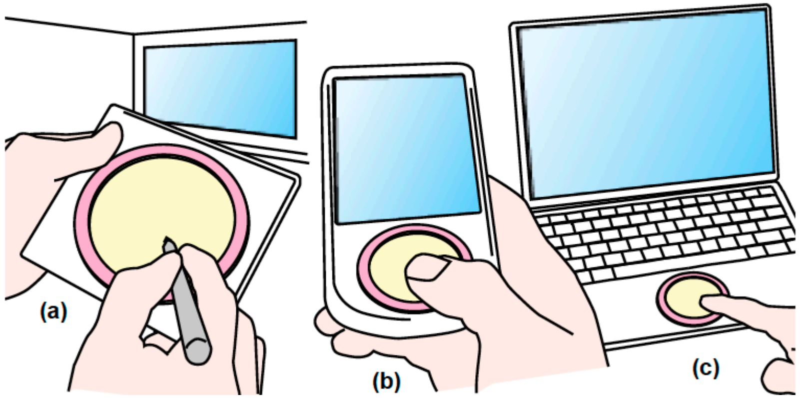
General principles

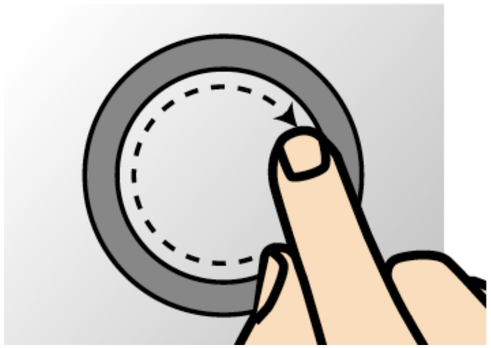
Isotonic devices (e.g., mouse) most appropriate for position control

Elastic/isometric devices (e.g., joystick) most appropriate for rate (velocity) control

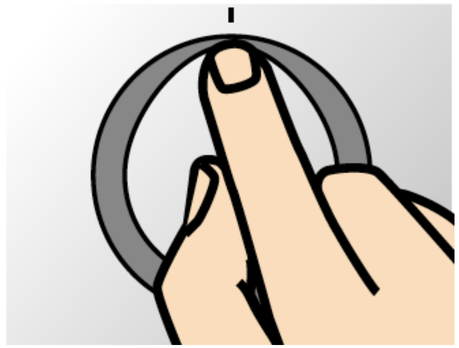
Mixed control (Casiez et al., 2007)

How can we increase the input space of a trackpad to reduce clutching: trackpad + trackpoint





Position control



Velocity control

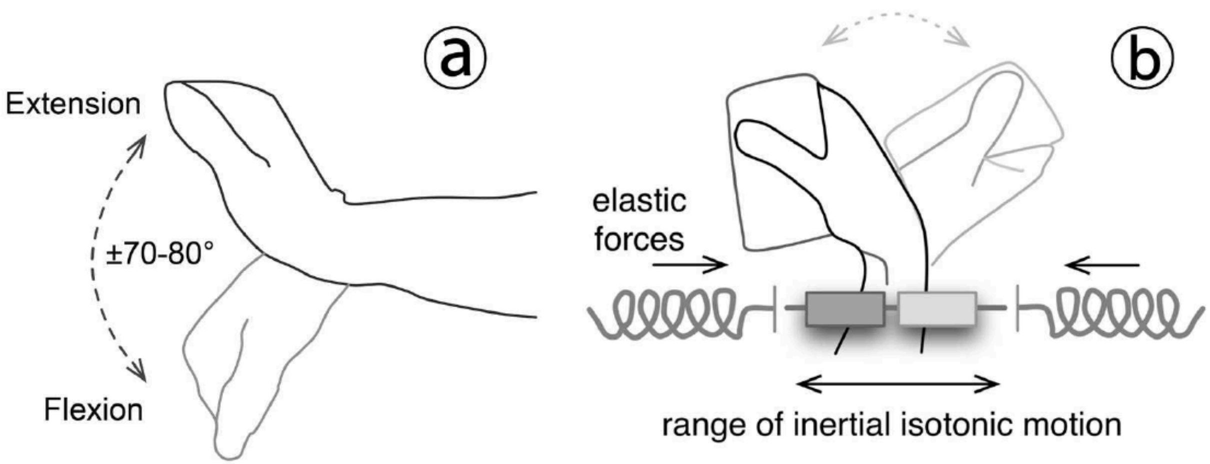
RubberEdge http://www.youtube.com/watch?v=kucTPG_zTik

Mixed control

The wrist as a mixed-control device (Tsandilas et al. 2013)

position control around the neutral wrist position

rate control near extemes angles

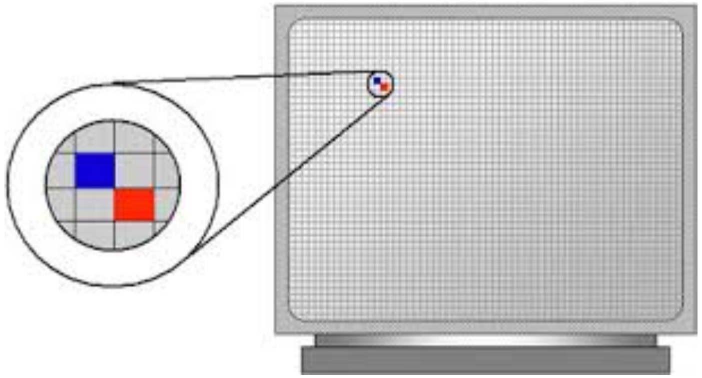


No need for clutching

Output resolution

Dots per Inch (DPI)

For screens where dots are pixels, we use the term Pixels per Inch (PPI)



Input resolution (isotonic devices)

Input resolution often measured in **counts per inch (CPI)**

- Also refered to as Dots per Inch (DPI)

A modern mouse: 400 to 10000 CPI

- Detection of displacements between 64μm and 2.54μm (about the size of a bacterium)

Input resolution (isotonic devices)

Input resolution often measured in **counts per inch (CPI)**

- Also refered to as Dots per Inch (DPI)

A modern mouse: 400 to 10000 CPI

- Detection of displacements between 64μm and 2.54μm (about the size of a bacterium)

« Useful » resolution: 200-400 CPI (Aceituno et al. 2013)

- Maximum resolution that users can benefit from

Control-Display (CD) gain

$$CD_{gain} = V_{pointer} / V_{device}$$

$V_{pointer}$: velocity of cursor
V_{device} : velocity of input device

Control-Display (CD) gain

$CD_{gain} = V_{pointer} / V_{device}$

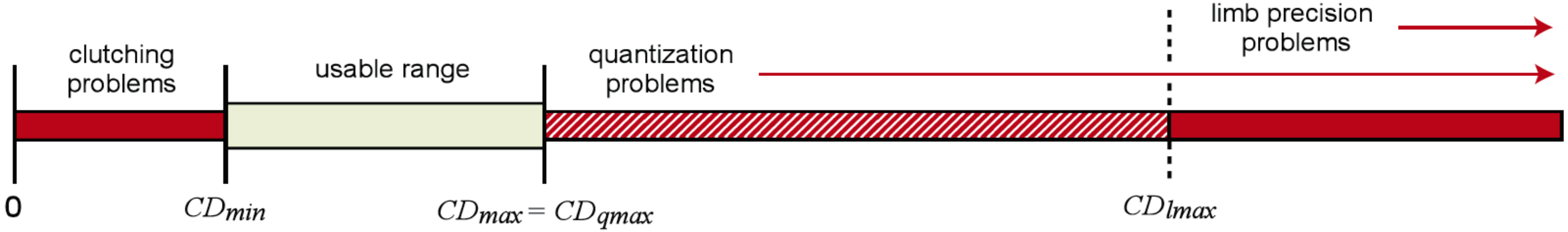
$V_{pointer}$: velocity of cursor
V_{device} : velocity of input device

CD_{gain}=1
When the mouse moves 1cm, the cursor also moves 1cm

CD_{gain}< 1
The cursor moves slower than the mouse: Better precision

CD_{gain} > 1
The cursor moves faster than the mouse: Faster, less clutching

Range of usable CD gains



from Casiez et al. (2008)

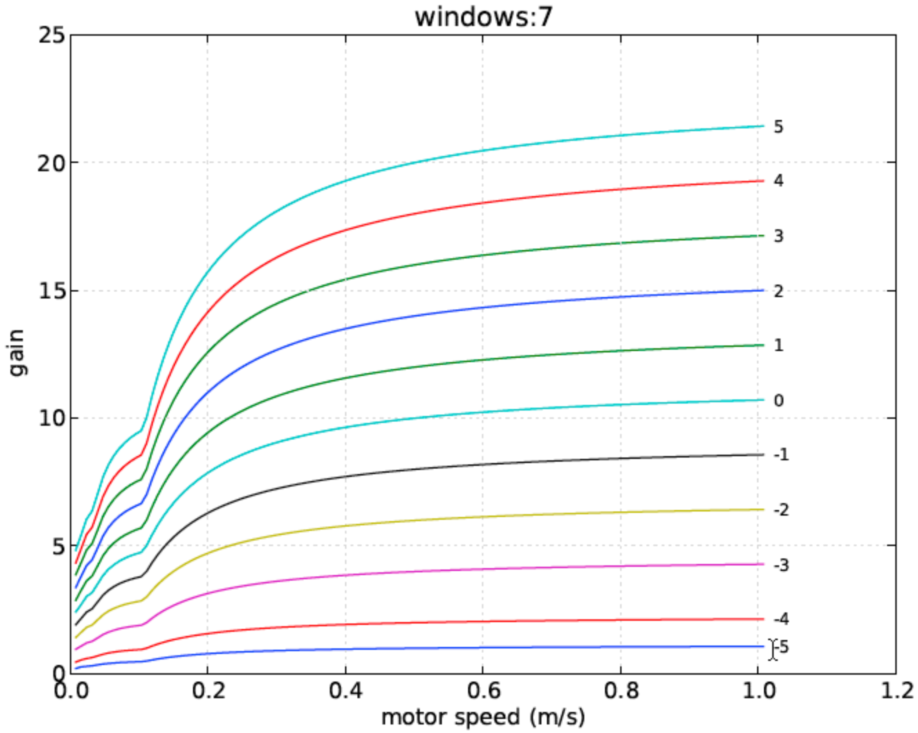
Pointer acceleration

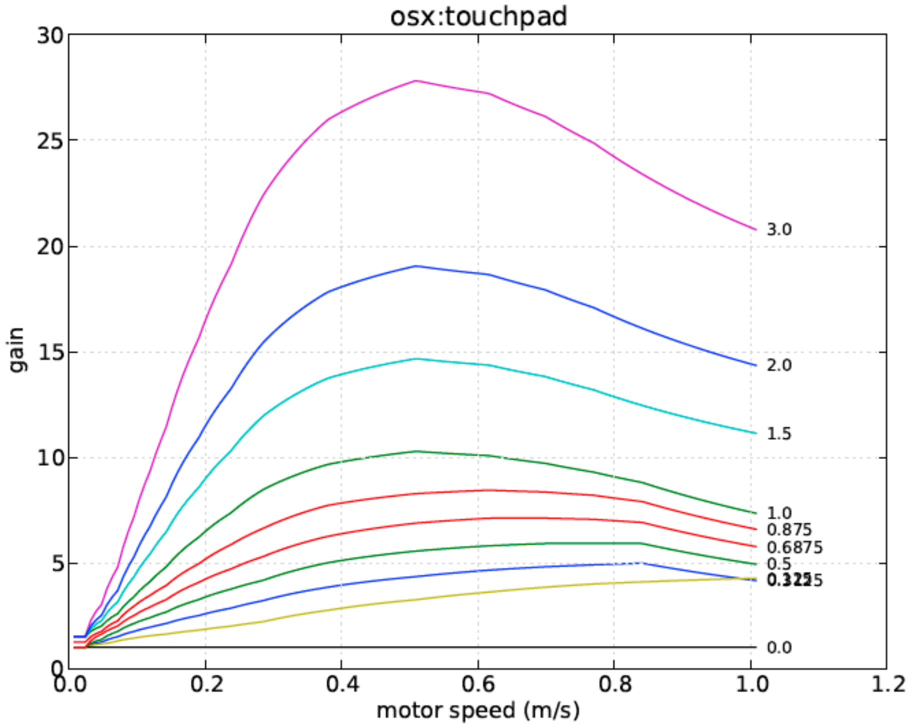
The CD gain is not constant but changes as a function of the speed of the device

- The faster I move the device, the faster the cursor (acceleration)
- Slow movements cause the CD gain to decrease: better precision

Acceleration functions

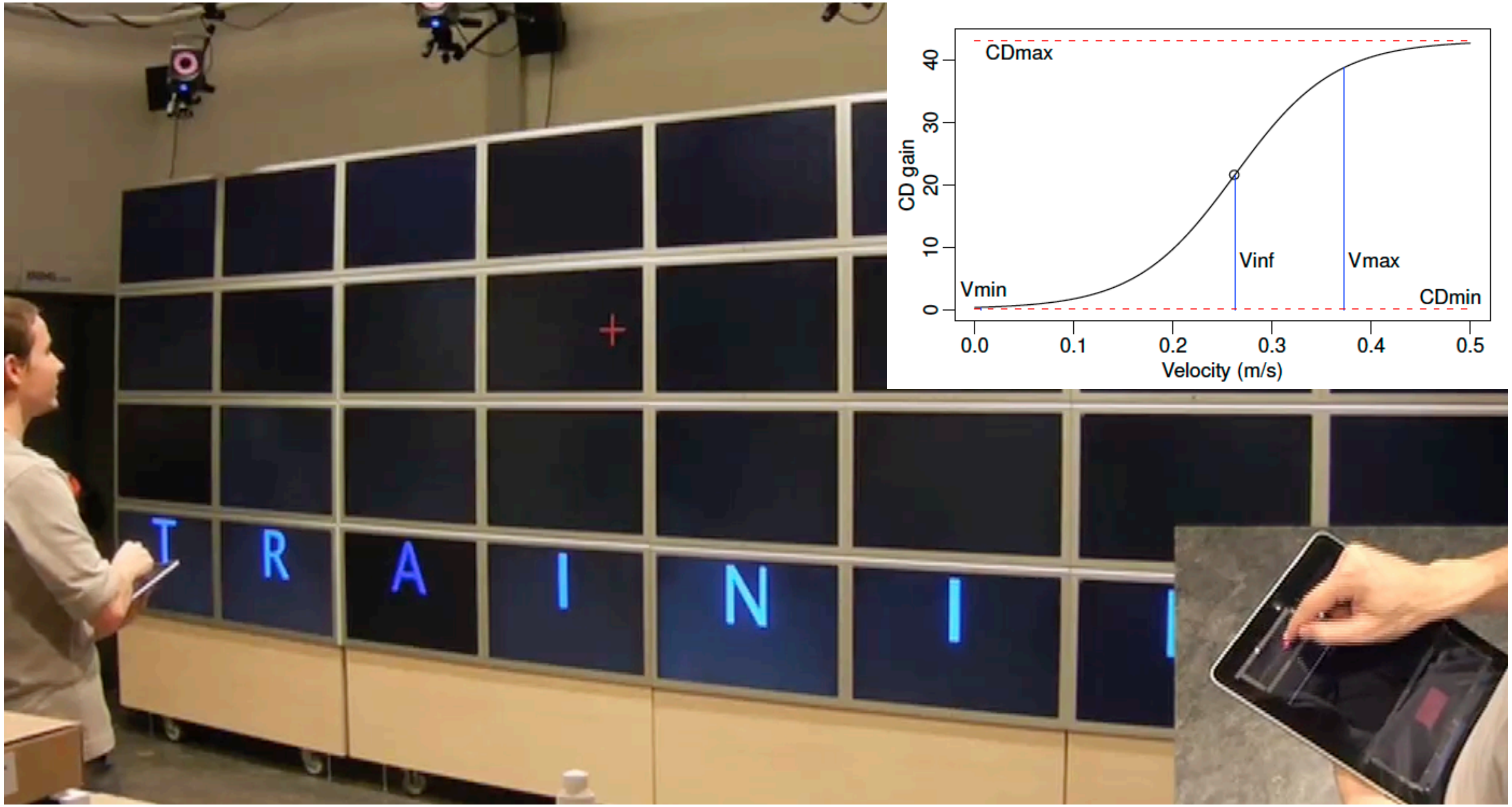
Also known as transfer functions





from Casiez and Roussel (2011)

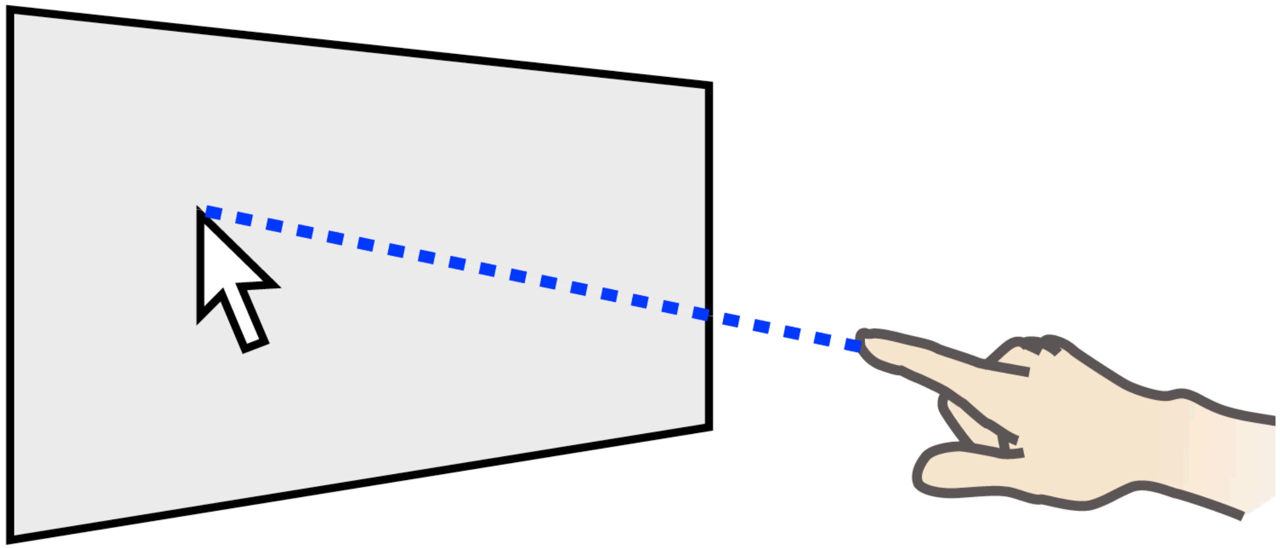
Nancel et al. (2013) found that with a good acceleration function, users could be very accurate and fast acquiring targets on a large high-resolution display even when the available input space was very small



Laser pointing – RayCasting

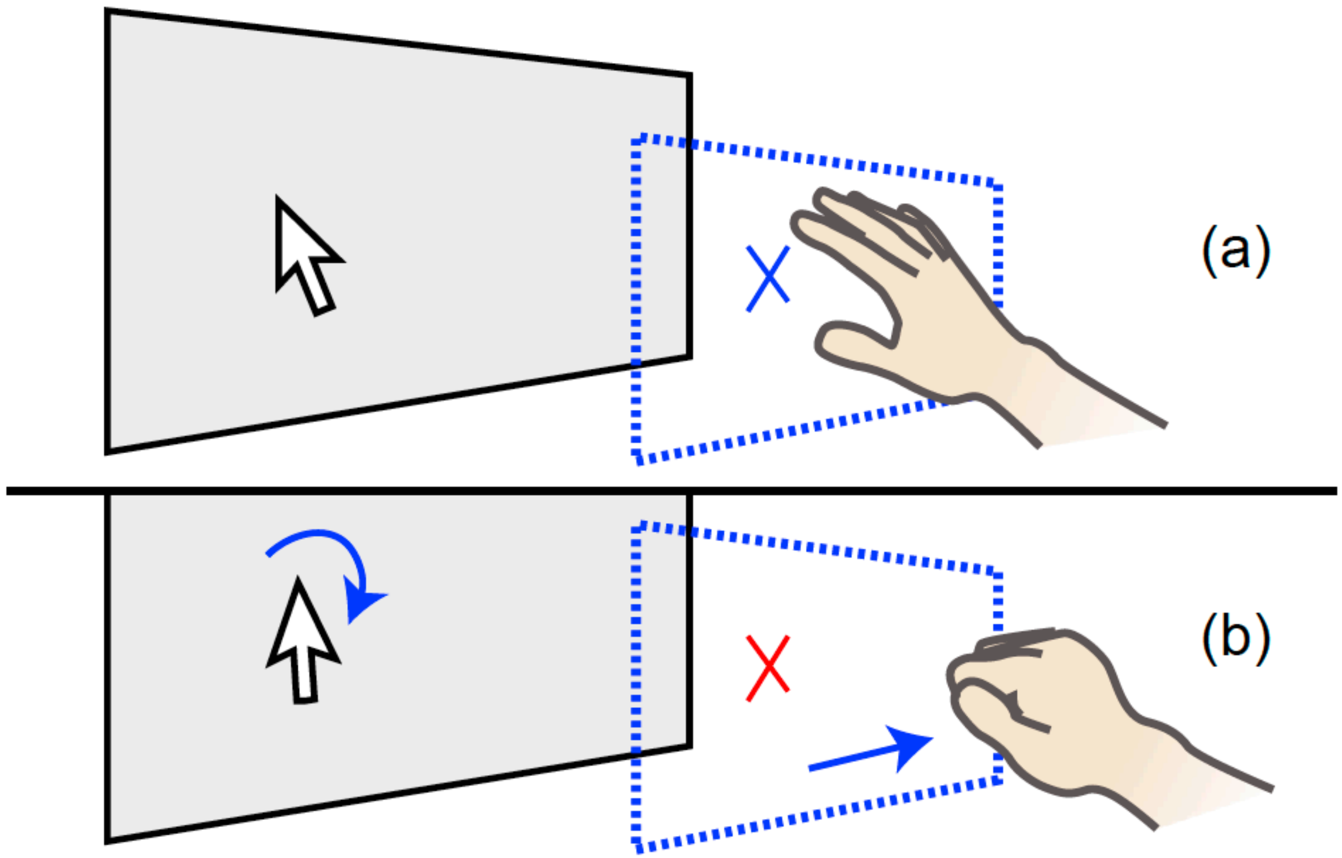
Main strength: Natural, as the device or hand points directly to the target

Drawback: Sensitive to hand tremor and tracking precision. Depending on the distance of the user, small hand movement can cause large displacements, inappropriate for accurate pointing from distance



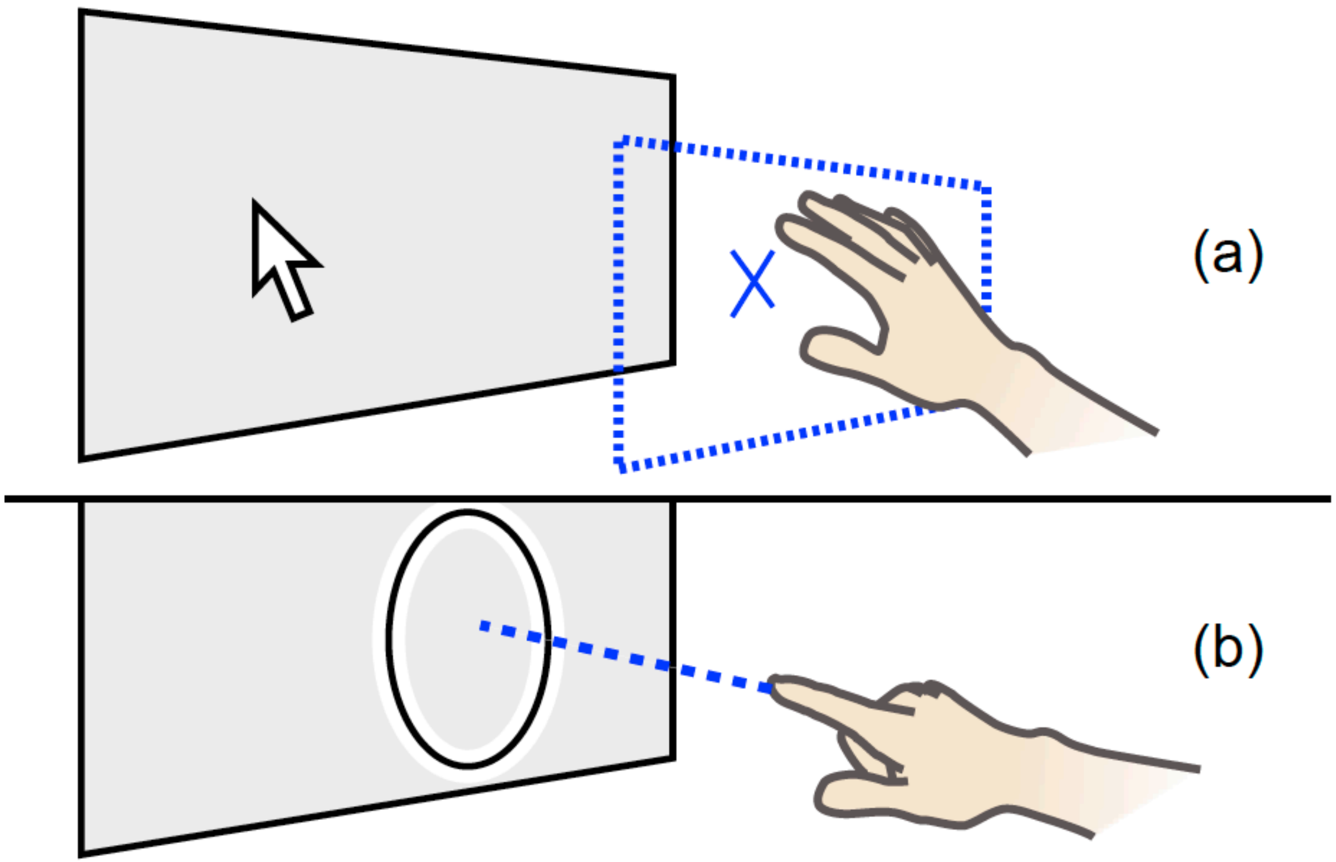
Solutions

Relative Pointing + Clutching (Vogel & Balakrishan, 2005)



Solutions

Hybrid Control (Vogel & Balakrishan, 2005)



http://www.youtube.com/watch?v=j26JQxMhBog

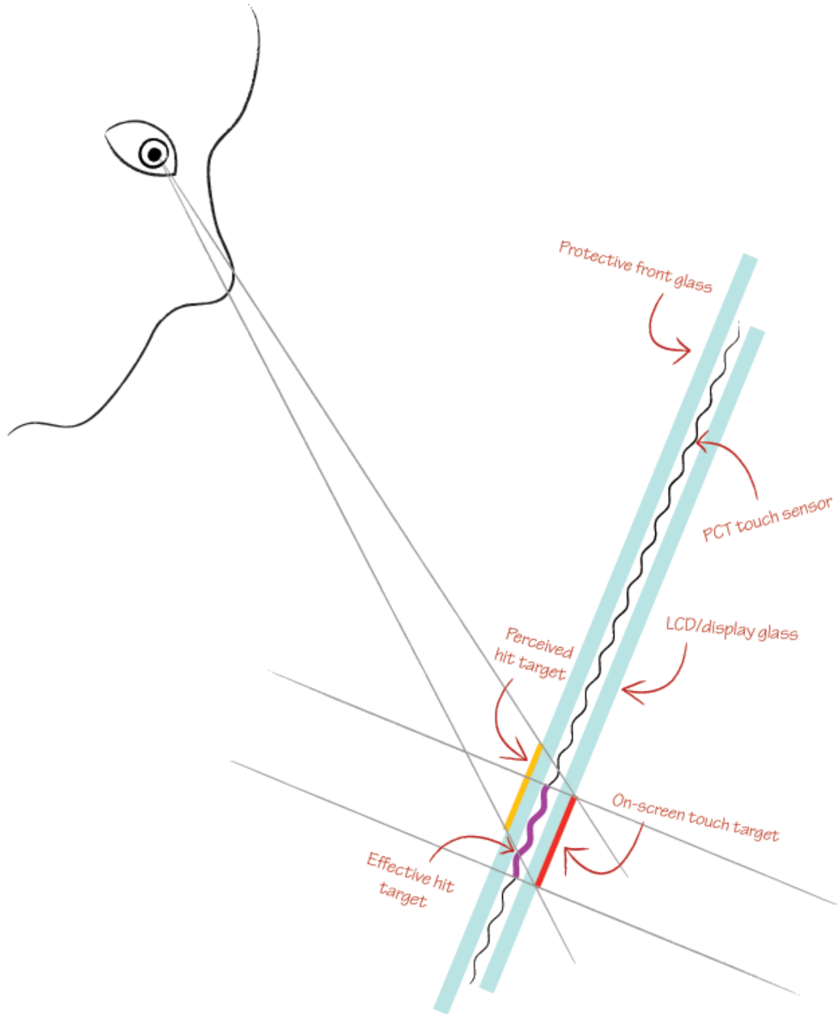
Direct input

Strengths: The user interacts directly with the objects as in the real world

Drawbacks: Lower accuracy due to occlusion, parallax, limited input resolution of the human limbs

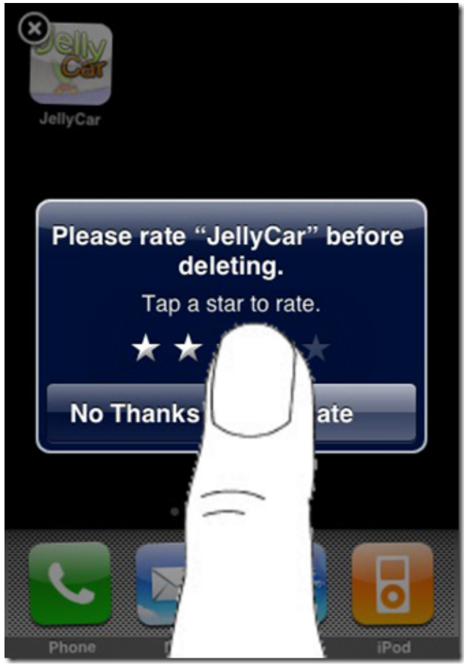
The parallax problem

Incorrect perception of where the target is

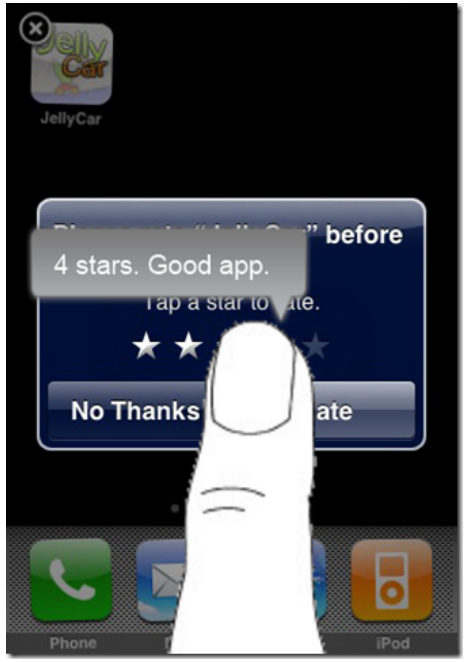


Occlusion problems

The finger covers the object of interest. Here, the letter under the finger grows and moves upwards to reduce the problem.



Problematic design



Better design

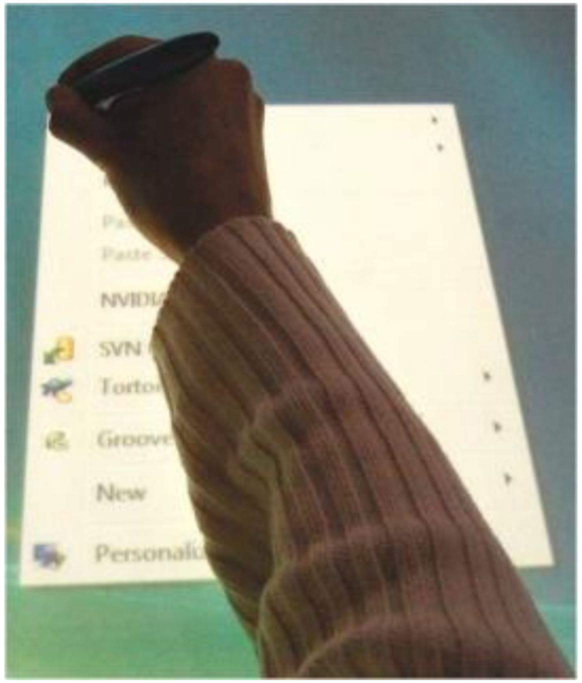
Examples from http://podlipensky.com/2011/01/mobile-usability-sliders/

Occlusion problems

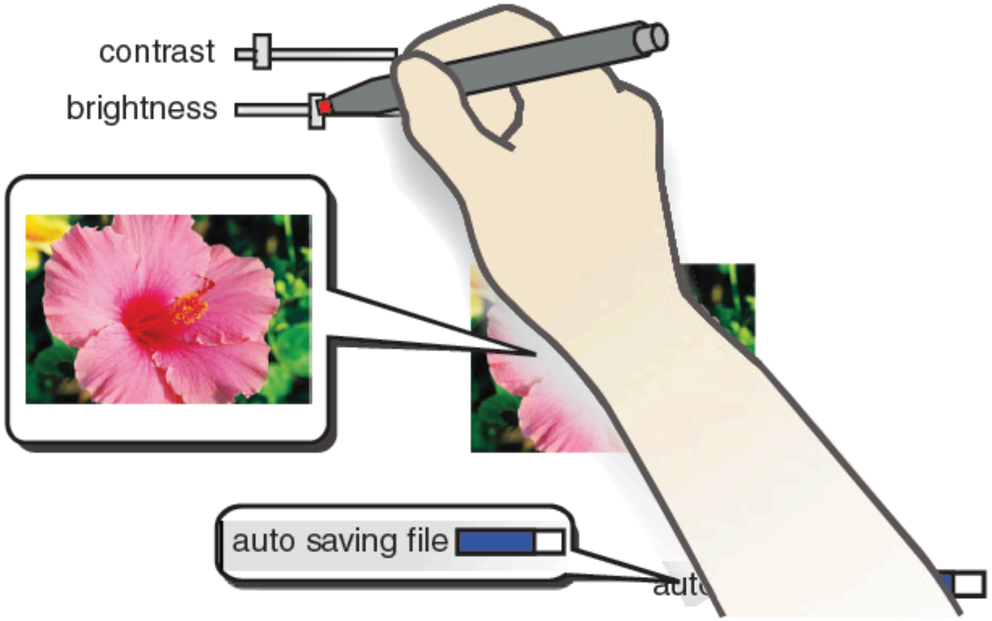
Sliding Widgets (Moshovich, 2009)
Replacing push buttons by sliding ones to reduce ambiguity due to occlusion or parallax problems (crossing-based selection)

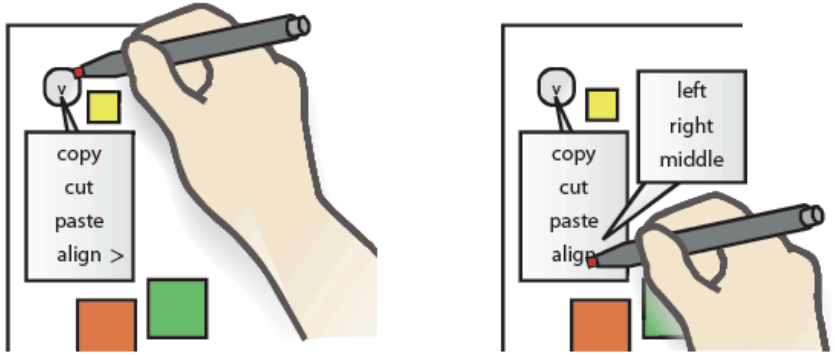
http://www.youtube.com/watch?v=Pw5nmLSYrvE

Hand occlusion

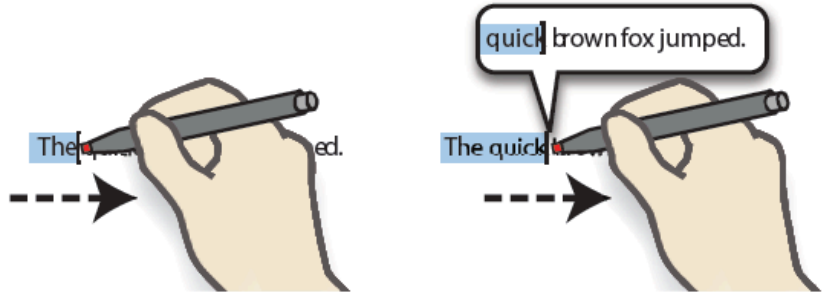


Occlusion-Aware Interfaces
(Vogel & Balakrishan, 2010)





(a) Occlusion-Aware Pop-Ups

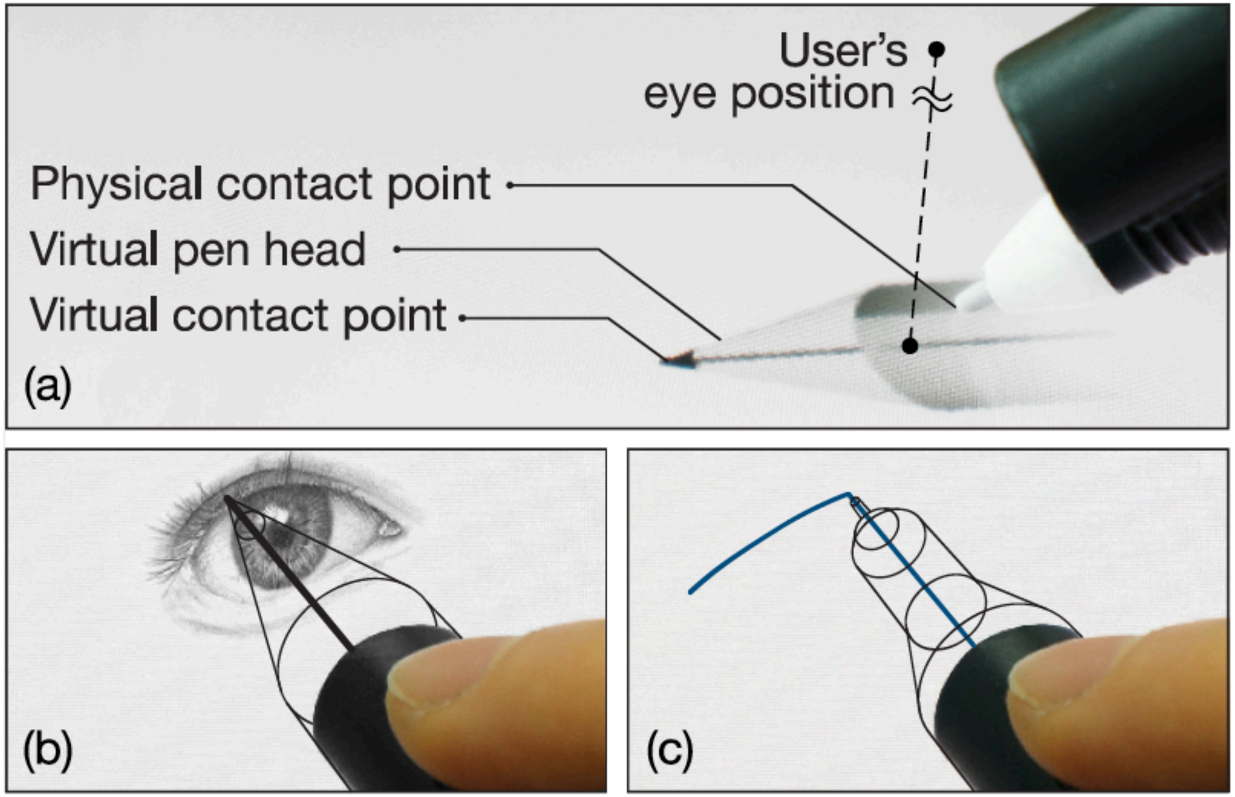


(b) Occlusion-Aware Dragging

http://www.youtube.com/watch?v=j-b9q4ZjLHo

Other clever solutions

PhantomPen (Lee et al, 2012)



http://www.youtube.com/watch?v=r62wxK3Rma4