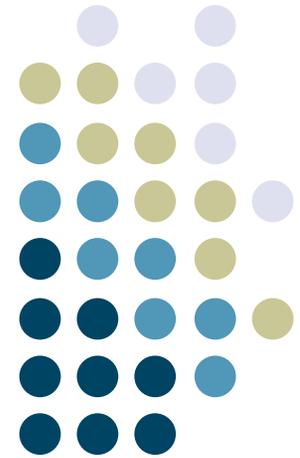
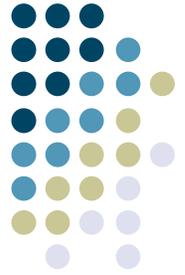


Input (part I: devices)



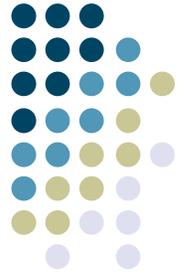
**Georgia
Tech**





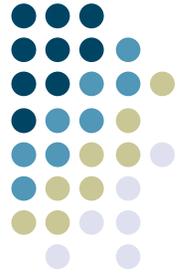
Where we are...

- Two largest aspects of building interactive systems: output and input
 - Have looked at basics of output
 - Now look at input



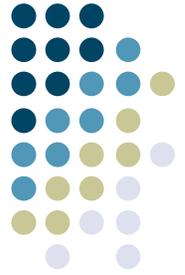
Input

- Generally, input is somewhat harder than output
 - Less uniformity, more of a moving target
 - More affected by human properties
 - Not as mature
- Will start with low level (devices) and work up to higher level



Input devices

- Keyboard
 - Ubiquitous, but somewhat boring...
 - Quite mature design
- QWERTY key layout
 - Where did it come from?



QWERTY key layout

- Originally designed to spread out likely adjacent key presses to overcome jamming problem of very early mechanical typewriters
 - Often quoted as “intentionally slowing down” typing, but that’s not true
 - Arrangement of letters to keep typebars from getting stuck
 - (Common letter pairs on alternating hands)

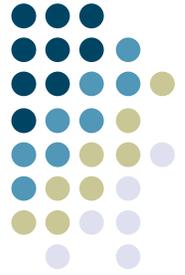




QWERTY keyboard layout

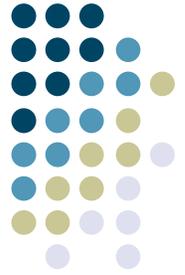
- Other layouts have been proposed
 - Dvorak is best known
 - Widely seen as better
 - Experimental and theoretical evidence casts doubt on this
 - Alternating hands of QWERTY are a win since fingers move in parallel





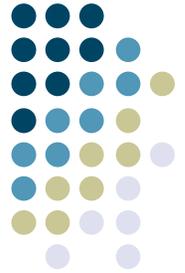
QWERTY keyboard layout

- Whether or not Dvorak layout is better, it did not displace QWERTY
 - Lesson: once there is sufficient critical mass for a standard it is nearly impossible to dislodge (even if there is an apparently good reason to do so)



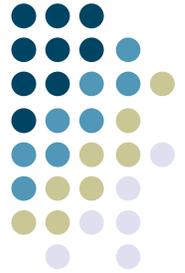
Keyboards

- Repetitive Stress Injury
 - First comes up here, mouse tends to be a little worse for most people
- Take this seriously for yourself!
 - Can be a VERY bit deal
 - Biggest thing: adjust your work environment (e.g. chair height)



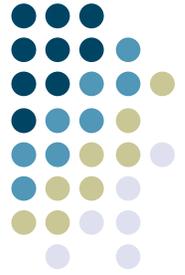
Buttons

- Similar to keyboard, but not for typing letters but for symbols
 - separate collection of keys with typically same form but different purpose
 - now see as “function keys” that come standard with keyboards
 - also show up on e.g., mouse



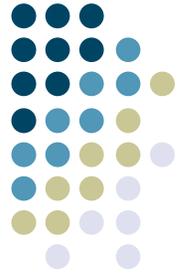
Buttons

- Buttons often bound to particular commands
 - e.g., function keys
 - Improved quite a bit with labels
 - Software changeable labels would be ideal, but we don't typically get this



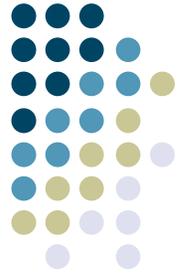
Valuators

- Returns a single value in range
- Major impl. alternatives:
 - Potentiometer (variable resistor)
 - similar to typical volume control
 - Shaft encoders
 - sense incremental movements
- Differences?



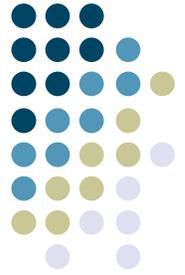
Valuator alternatives

- Potentiometer
 - normally bounded range of physical movement (hence bounded range of input values)
 - Keeps residual position in device
- Shaft encoder
 - Unbounded range of movement
 - No residual position in device



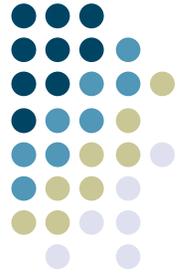
Locators (AKA pointing devices)

- Returns a location (point)
 - two values in ranges
 - usually screen position
- Examples
 - Mice (current defacto standard)
 - Track balls, joysticks, tablets, touch panels, etc.



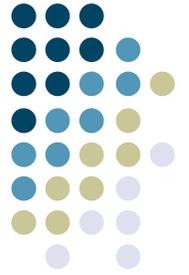
Locators

- Two major categories:
 - Absolute vs. Relative locators



Absolute locators

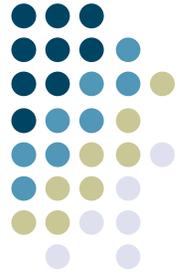
- One-to-one mapping from device movement to input
 - e.g., tablet
 - Faster
 - Easier to develop motor skills
 - Doesn't scale past fixed distances
 - bounded input range
 - less accurate (for same range of physical movement)



Relative locators

- Maps movement into rate of change of input
 - e.g., joystick (or TrackPoint)



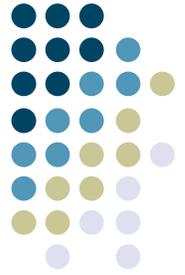


Relative locators

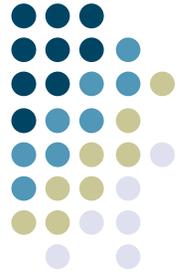
- More accurate (for same range of movement)
- Harder to develop motor skills
- Not bounded (can handle infinite moves)

Q: is a mouse a relative or absolute locator?

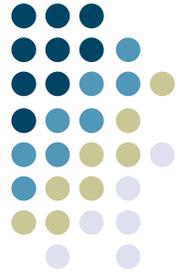
Georgia
Tech



Q: is a mouse a relative or absolute locator?

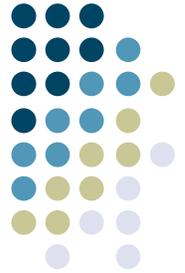


- Answer: No
- Third major type:
“Clutched absolute”
 - Within a range its absolute
 - Can disengage movement (pick it up) to extend beyond range
 - picking up == clutch mechanism



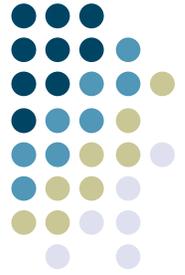
Clutched absolute locators

- Very good compromise
 - Get one-to-one mapping when “in range” (easy to learn, fast, etc.)
 - Clutch gives some of benefits of a relative device (e.g., unbounded)
- Trackballs also fall into this category



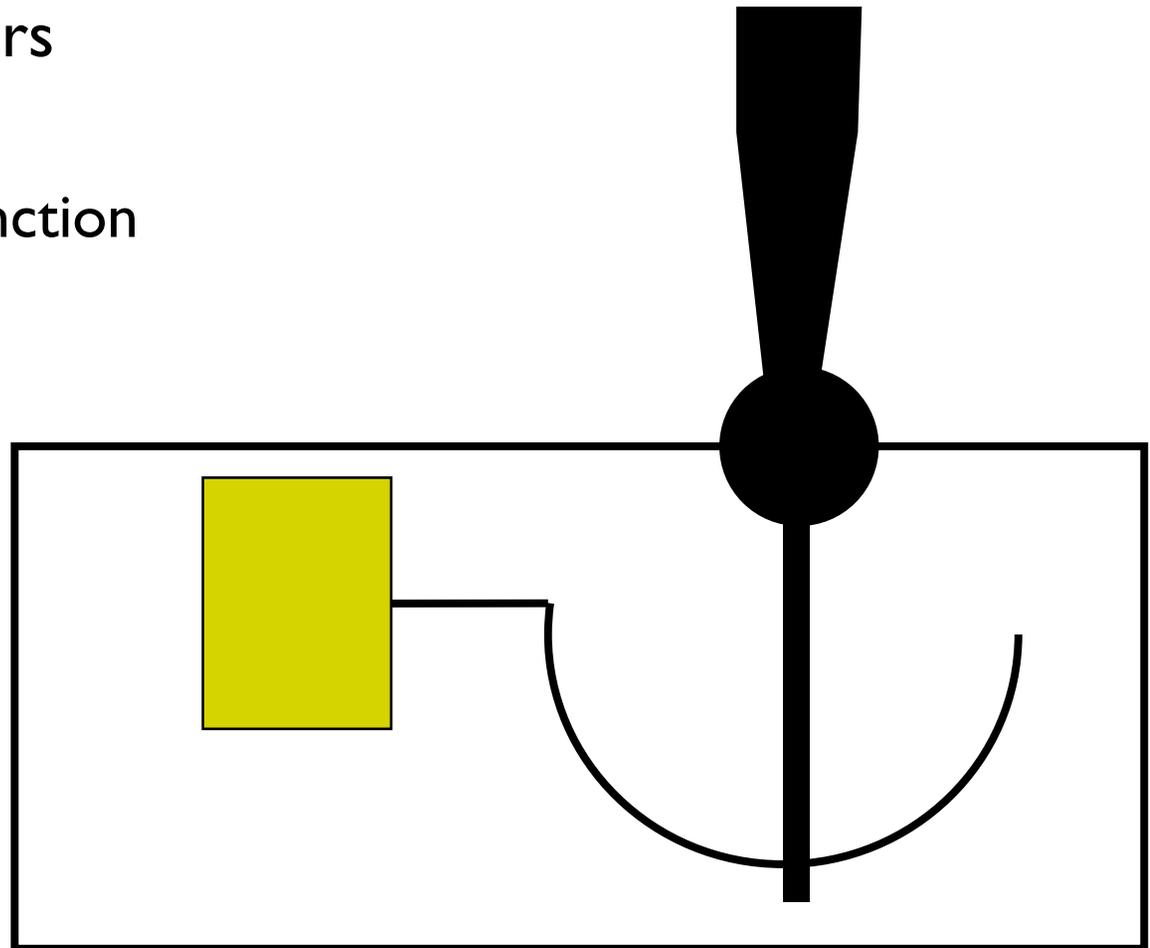
Device specifics: joysticks

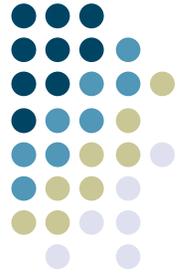
- self centering
- relative device
- possible to have absolute joysticks, but scaling is bad



Joystick construction

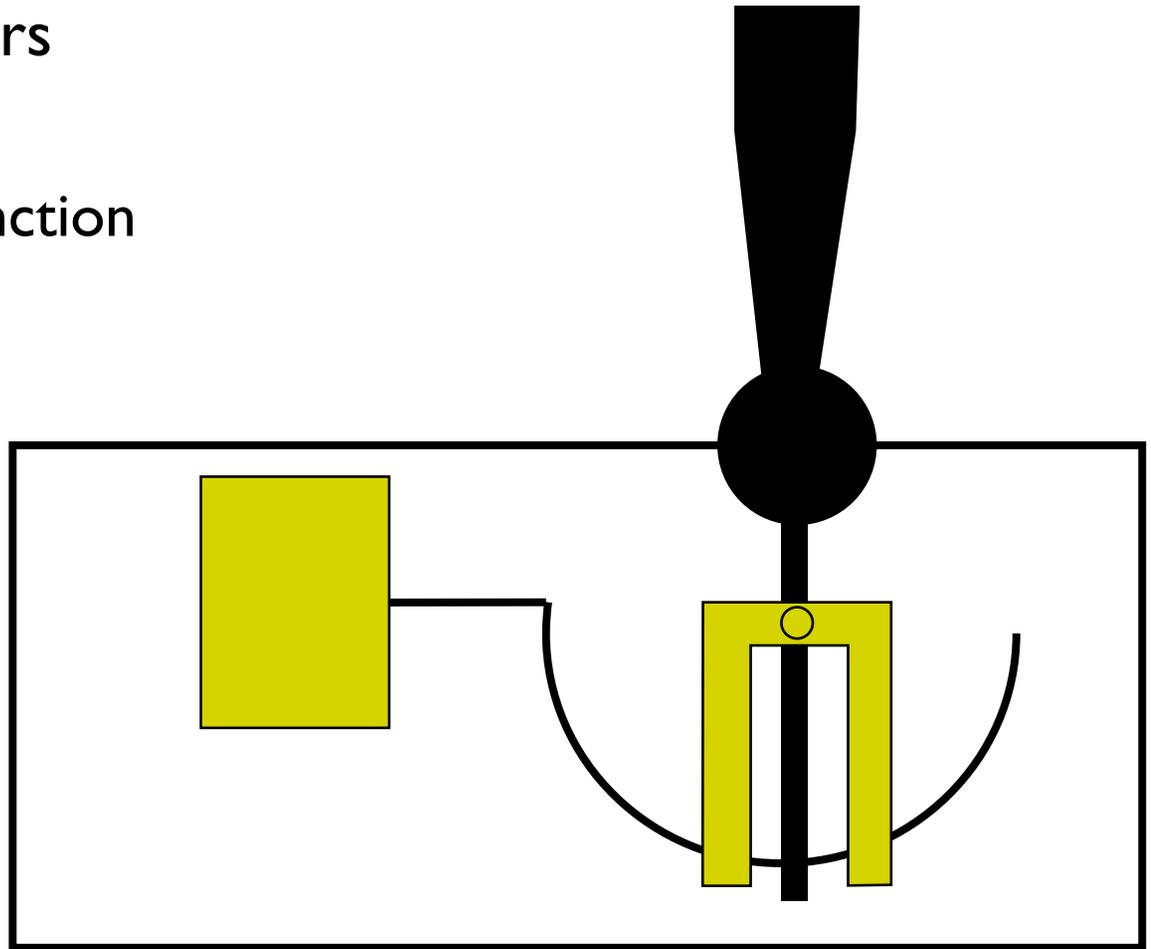
- Two potentiometers
 - x and y
 - resistance is a function of position

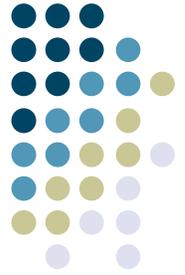




Joystick construction

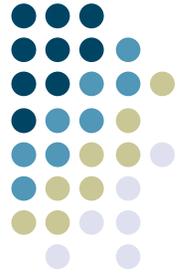
- Two potentiometers
 - x and y
 - resistance is a function of position





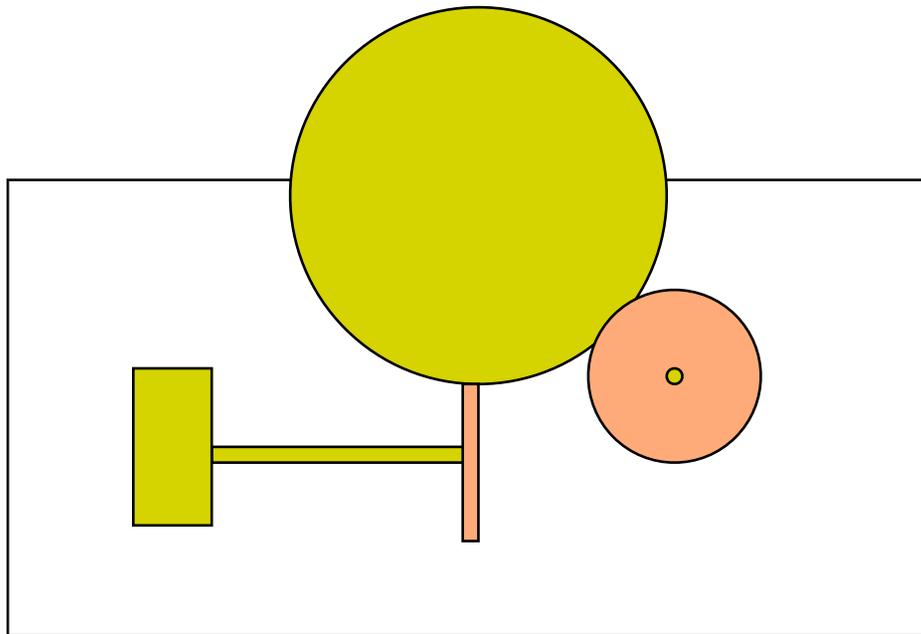
Joystick construction

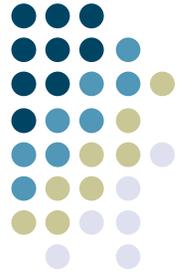
- TrackPoint (IBM technology)
 - uses strain gauge sensors
- Also can be implemented with switches
 - one in each direction
 - Fixed speed of movement



Trackballs

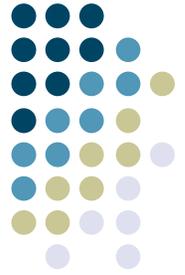
- (Typically large) ball which rolls over 2 wheels





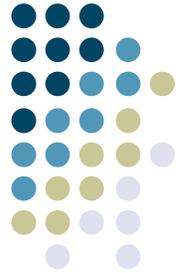
Trackballs

- Clutched absolute
 - but with small movement range
- Infinite input range, etc.
- Properties vary quite a bit
 - scaling of movements
 - mass of ball
 - high mass ball can act as a relative device by spinning it



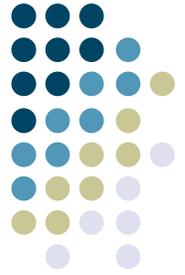
Mouse

- Clutched absolute
 - infinite range, etc.
- How is it constructed?



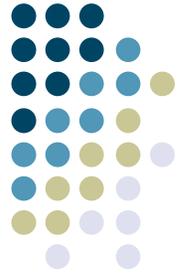
Mouse

- Clutched absolute
 - infinite range, etc.
- How is it constructed?
 - Turn a trackball upside down



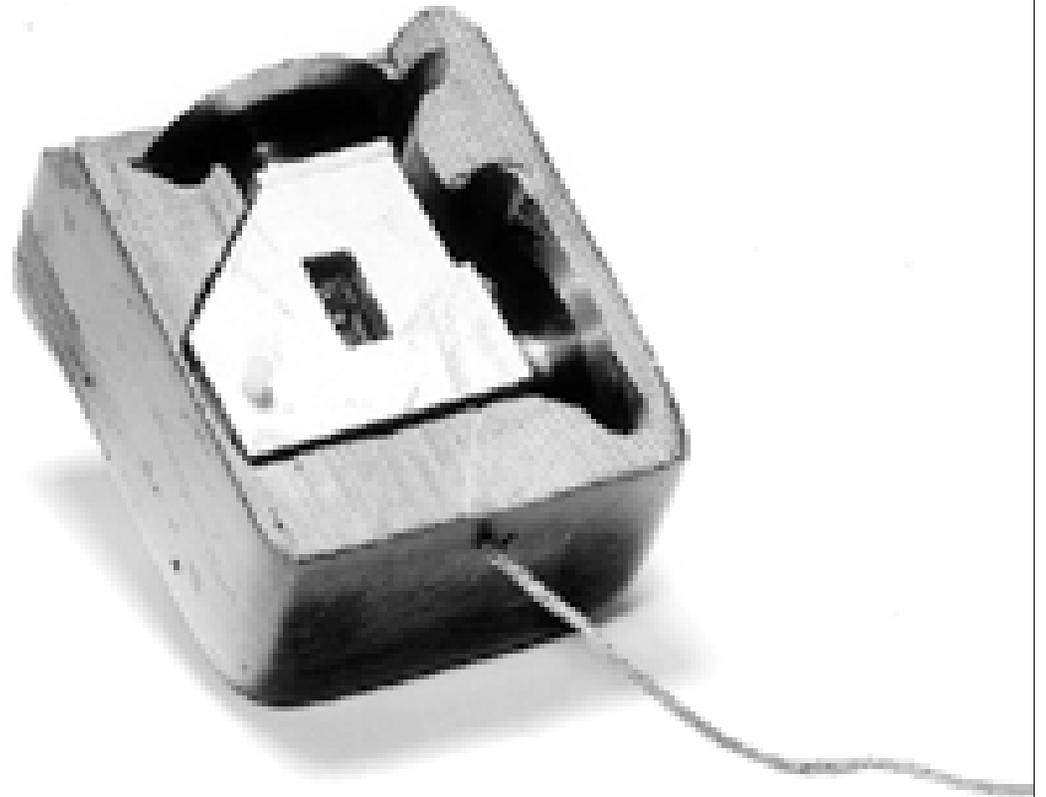
Mouse

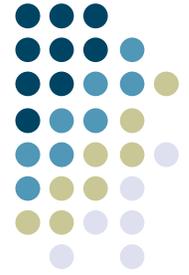
- Current dominant device
 - so much so that some people call any pointing device a “mouse”
 - overall a very good device



Mouse

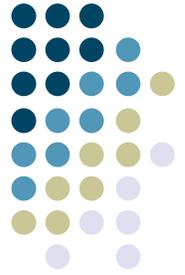
- Invented by Douglas Engelbart et al. ~1967





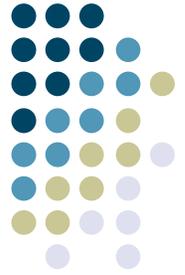
Touch panel

- What kind of a device?



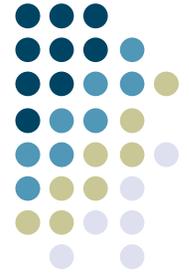
Touch panel

- Absolute device
- Possible to do input and output together in one place
 - actually point at things on the screen
- Resolution limited by size of finger (“digital input”)
 - Or requires a pen



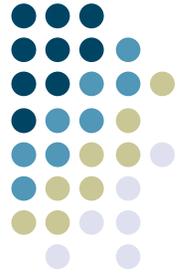
Touch panel construction

- Membrane
 - resistive, fine wire mesh
- Capacitive
- Optical
 - finger breaks light beam
- Surface acoustic waves



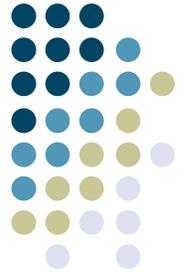
Drawing tablet

- Absolute or relative?



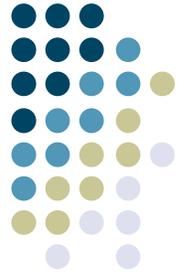
Drawing tablet

- Absolute device
- Normally used with pen / stylus
 - Allows “real drawing” (try drawing with a mouse vs. a pen)
 - Can often trace over paper images



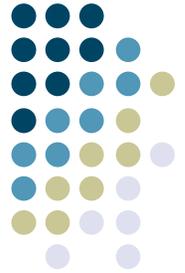
Construction of drawing tablet

- Traditional (“Rand”) tablet
 - middle 60’s
 - grid of wires (~100 / inch)
 - each wire transmits binary of its coord
 - stylus picks up closest
- Can also make pen transmitter and tablet receiver



Drawing tablet details

- Typically have tip switch
- May also have switch(es) on side of stylus
- Can also support a “puck” with buttons
- Best current devices can support multiple “pens” at the same time and sense rotation of a puck

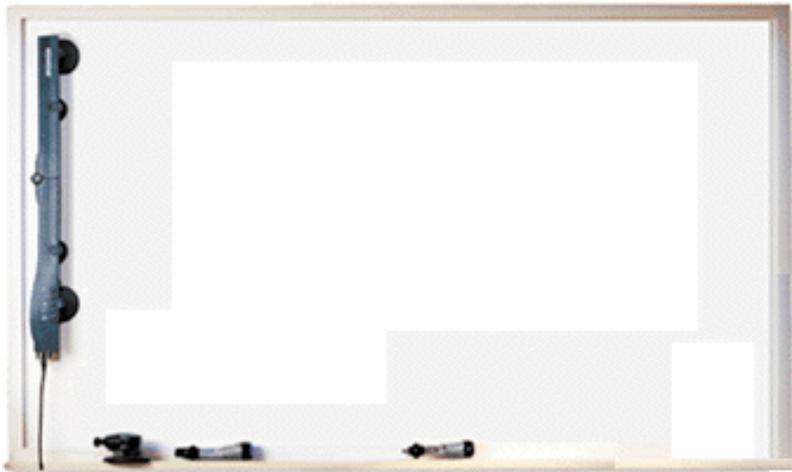


Alternate Approaches to Tablets

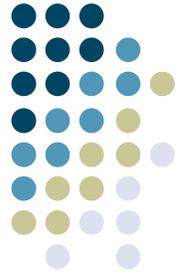
- Old acoustic (sort of a fun device)
 - stylus emits spark
 - strip microphones at edge of tablet
 - difference in arrival time of sound

Interesting device: Virtual Ink Mimio

Georgia
Tech

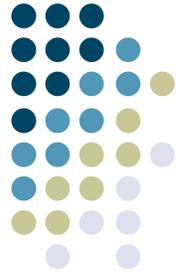


- Updated acoustic tablet
 - recording whiteboard
 - ultrasonic chirps
 - 100dpi resolution over ~8ft



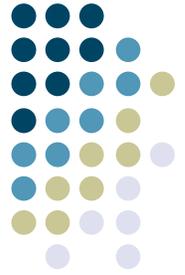
3D locators

- Can extend locators to 3 inputs
- Some fun older devices
 - 3D acoustic tablet
 - Wand on reels
 - Multi-axis joystick



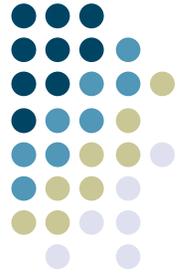
3D locators

- Typical for VR use: Polhemus
 - 6D device (x,y,z + pitch, roll, yaw)
 - Magnetic sensing technology
 - Doesn't work well near metal
 - Doesn't work well near deflection coils of CRT



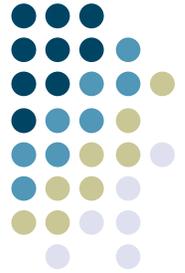
Light pen (a very old device)

- A “pick” device
 - returns ID of an “object” on the screen (not a position)
- For vector refresh displays
 - Vector refresh worked with small “display list processor”
 - Add register holding current obj ID
 - Photocell causes interrupt when beam passes (grab and return ID)



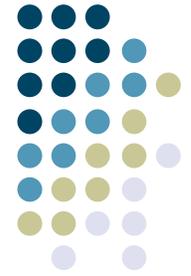
Light pen (a very old device)

- Can't really do this anymore
 - on raster display light pen is just a locator
- But its conceptually what we usually want for input: what object the user is pointing at
 - We will simulate in SW (“picking”)



Lots of other devices

- Still mostly KB + mouse, but increasing diversity
 - Cameras!
 - Lots of untapped potential in vision
 - Microphones
 - speech as data
 - speech recognition

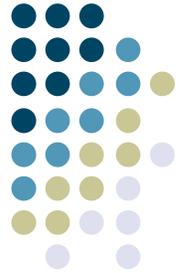


Lots of other devices

- Any favorites?

Some interesting ones I know about

Georgia
Tech



- Thumb Wheel
- DataGlove
- Motion detectors (and other sensors)
- Accelerometers
- Fingerprint readers
- RF tags (physical objects as tokens for data/action)
- Sub-gram resolution scales

**Georgia
Tech**

