Devices

Keyboards and Keypads

- We now give a brief introduction to the most important families of input and output devices.
- We first review text entry including keyboards and keypads as well as their layout, physical design, and accessibility adaptations. We discuss text entry techniques for mobile devices.
- Pointing is another common interaction for interfaces
- Finally we present both traditional as well as novel display technologies, focusing on particularities of large and small displays as well as wearable computing

Keyboards and keypads

- **Keyboard layouts**
- *QWERTY layout* puts frequently used letter pairs far apart, thereby increasing finger travel distances.
- The *Dvorak layout* increases the typing rate of expert typists from about 150 words per minute to more than 200 words per minute and even reduce errors.
- A third keyboard layout is the *ABCDE style*, which has the 26 letters of the English alphabet laid out in alphabetical order. The rationale here is that non-typists will find it easier to locat the keys.

- **For people with motor impairments**
- adaptive keyboards, where keys are lowered instead of raised to aid acquisition
- Onscreen keyboards accessed using alternative input devices like head pointers or oversized trackballs.
- dictionary-based autocompletion as well as automatic error correction .
- For visually impaired users BrailleTouch provides nonvisual input methods for one-handed or two-handed Braille typing on multi-touch smartphone
- Pointing devices such as mice, touchpads, or eye-trackers may be used for data entry and replace keyboards altogether.
- wearable devices, such as a wristband or ring form factor, to enter text .
- for mobile settings, there is increasing use of speech input instead of keyboards
- Large-print keyboards are available for vision-impaired users

Keys

- The keys tend to have slightly concave surfaces for good contact with fingertips and a matte finish to reduce both reflective glare and the chance of finger slips.
- Keypresses require a 40- to 125-gram force and a displacement of 1 to 4 millimeters, which enables rapid typing with low error rates while providing suitable feedback to users.
- An important element in key design is the profile of force displacement.
- When the key has been depressed far enough to send a signal, the key gives way and emits a very light click. This tactile and audible feedback is extremely mportant in touch typing.
- membrane keyboards that use a nonmoving surface are difficult to use for extensive touch typing.
- Such keyboards are durable and therefore acceptable for challenging environments such as fast-food restaurants, factory floors, or amusement parks

- Keys, such as the space bar, Enter key, Shift key, or Ctrl key, should be larger than others to allow easy, reliable access.
- Other keys, such as Caps Lock and Num Lock, should have a clear indication of their state, such as by physical locking in a lowered position or by an embedded light.
- . The placement of the cursor-movement keys (up, down, left, and right) is important in facilitating rapid and error-free use.
- The popular and compact inverted-T arrangement of arrow keys allows users to place their middle three fingers in a way that reduces hand and finger movement
- Some large keyboards reuse the peripheral eight keys on the numerical keypad (all keys except the central 5 key) to simplify diagonal movements.
- For such keyboards, the Num Lock key is used to toggle between keypad and arrow mode.
- In some applications, such as games, where users spend hours using the movement keys, designers reassign letter keys as cursor-movement keys to minimize finger motion between the movement keys and other action keys.
- The WASD keys are often used for this purpose.
- Auto-repeat feature, where repetition occurs automatically with continued depression, may improve performance, but control of the repetition rate must be provided to accommodate user preferences (this is particularly important for very young users, older adult users, and users with motor impairments).

Mobile text entry

- As computers morph into new form factors—such as tables, tablets, and phones—as well as become universally usable for a broader population of users from different backgrounds, nationalities, and capabilities, text entry is also changing beyond the traditional keyboard.
- Most older or low-cost mobile devices provide only a numeric keypad.
- Entering text using keypads requires multiple taps, where users hit a number key multiple times to cycle through several letters assigned to that key.
- In some mobile devices physical keyboards are favored in place of soft keyboards, some still use a
- traditional QWERTY keyboards
- . Physical keyboards are still preferred by many mobile users who need to enter large amounts of text using their phones, such as to manage e-mail on the go.

- In touchscreen technology physical mobile keyboards are increasingly being replaced by virtual, or so-called “soft,” keyboards, where the keyboard is merely a visual representation on the touchscreen
- Projection keyboards, where the physical world is appropriated to display an image of the keyboard, are based on the same principle.
- The benefit of soft keyboards is that they can be dynamically relabeled, such as for a new character set or layout, as well as rescaled and rotated to fit the physical display and device orientation.
- Soft keyboards lack the tangible and tactile feedback of a physical keyboard, they are difficult to use
- for eyes-free operation and typically yield only modest performance, around 20 to 30 words per minute.
- Several methods exist to improve text entry on touchscreens.
- Just like keypad-based text-entry methods can make use of dictionarybased or predictive text-entry algorithms, current touchscreen
- Textentry methods commonly suggest possible word completions given an input string.
- More advanced techniques use language models to predict the word the user is trying to write based on the current sentence;
- Another text entry method is simply to write by hand on a touchsensitive surface, typically with a stylus, but character recognition remains error prone

- Another promising method is to allow shorthand gesturing on a keyboard instead of tapping on a touchscreen keyboard, using shapes that match the tapping patterns. Long-term studies confirm that it is possible to achieve good text-entry performance with this technique .

Pointing Devices

- The new generation of touch displays invites users to tap, drag, and pinch the images on the screen directly.
- With complex information displays such as those found in computer-assisted design tools, drawing tools, or air-traffic-control systems, it is often convenient for the user to point at and select items. This direct manipulation approach is attractive because the users can avoid having to learn commands, reduce the chance of typographic errors on a keyboard, and keep their attention on the display.
- The results are often faster performance, fewer errors, easier learning, and higher satisfaction. Pointing devices are also important for small devices and large wall displays that make keyboard interaction less practical

- There are many ways to categorize pointing devices, such as physical device attributes (rotation or linear movement), number degrees of freedom (horizontal, vertical, yaw, pitch, etc.), and positioning (relative or absolute).

Pointing tasks and control modes

- Pointing devices are useful for seven types of interaction tasks:
- 1. **Select.** Choosing from a set of items. This technique is used for traditional menu selection, the identification of objects of interest, or marking an object in a slide deck.
- 2. **Position.** Choosing a point in a one-, two-, three-, or highe rdimensional space. Positioning may be used to place shapes in a drawing, to place a new window, or to relocate a block of text in a figure.
- 3. **Orient.** Choose a direction in a two-, three-, or higherdimensional space. The direction may rotate a symbol on the screen, indicate a direction of motion, or control the operation of a device, such as a robot arm.
- 4. **Path.** Define a series of positioning and orientation operations. The path may be realized as a curving line in a drawing program, a character to be recognized, or the instructions for a cloth-cutting or other type of machine.

- 5. **Quantify.** Specify a numeric value. The quantify task is usually a one- dimensional selection of integer or real values to set parameters, such as the page number in a document, the velocity of a vehicle, or the music playback volume.
- 6. **Gesture.** Perform an action by executing a pre-defined motion. Examples of gestures include dwelling on an object to bring up a context menu, swiping to the left (or right) to turn a page forward (or backward), and pinching (or separating) your fingers to zoom out (or in).
- 7. **Text.** Enter, move, and edit text in 2-D space. The pointing
- device indicates the location of an insertion, deletion, or change
- More elaborate text tasks include centering, setting margins and font sizes, highlighting (boldface or underscore), and page layout.

- Pointing devices can be grouped into those that offer *direct control* on the screen surface, such as the touchscreen or stylus, and those that offer *indirect control* away from the screen surface, such as the mouse, trackball, joystick, graphics tablet, or touchpad.

- **Direct control devices** (easy to learn and use, but hand may obscure display)
- Touchscreen (single- and multi-touch)
- Stylus (passive and active)

- **Indirect control devices** (take time to learn)
- Mouse
- Trackball
- Joystick
- Pointing stick (trackpoint)
- Touchpad
- Graphics tablet

- **Novel devices and strategies** (for special purposes)
- Bimanual input
- Eye-trackers
- Sensors (accelerometer, gyroscopes, depth cameras)
- 3-D trackers
- Data gloves
- Haptic feedback
- Foot controls
- Tangible user interfaces
- Digital paper
- **Criteria for success**
- Speed and accuracy
- Efficacy for task
- Learning time
- Cost and reliability
- Size and weight

- Another way to think about pointing devices is whether they use
- absolute or relative input.
- Touchscreens, graphics tablets, and eyetrackers also use an input model where the input (motor) space is directly mapped to the output (visual) space.
- This is called *absolute input*, since one point in motor space corresponds to one point in visual space.
- *Relative input*, on the other hand, deals with translations (and rotations) from a current position and includes devices such as the mouse, joystick, and trackball.

Direct-control pointing devices

- **Touchscreens** are the canonical direct control pointing devices and allow users to interact directly with the visual content of the screen by touching it with their fingers.
- Because of their natural *affordance*, i.e. their form inviting appropriate action, touch-enabled screens are often integrated into applications directed at novice users in which the keyboard can be eliminated and touch is the main interface mechanism.
- Early touchscreen implementations had problems with imprecise pointing, as the software accepted the touch immediately (the land-on strategy), denying users the opportunity to verify the correctness of the selected spot.
- These early designs were based on physical pressure, impact, or interruption of a grid of infrared beams.

- High precision designs dramatically improved touchscreens.
- The resistive, capacitive, or surface-acoustic-wave hardware often provides up to 1600 × 1600 pixel resolution, and the so-called lift-off strategy enabled users to point at a single pixel.
- This lift-off strategy has three steps: Users touch the surface and then see a cursor that they can drag to adjust its position; when they are satisfied, they lift their finger off the display to activate.
- High-precision touchscreens have transformed mobile devices such as tablets and phones, to the point that it has become natural for users to be able to directly point to objects on the mobile display.
- Users perpetually ask for mobile devices to become smaller, lighter, and more powerful, touch computing has led to current mobile devices consisting almost entirely of a touchscreen.
- Pointing using the user's own fingers is prone to the "fat finger" problem , where the user's hand and fingers occlude on-screen content.
- New techniques such as Shift and occlusion-aware interfaces try to remedy this by displacing the screen content based on the user's touch interaction.
- Another way to avoid the fat finger problem is to use a *stylus,* which has a familiar and comfortable feel for most users while simultaneously minimizing hand-screen occlusion.
- Most stylus interfaces (also called "pen-based interfaces") are based on touchscreen

- users can write with a stylus for more natural handwriting and increased motion control but can
- also use a finger for quick selection .
- In fact, common capacitive touchscreens, which form the majority of today’s tablets and smartphones, can be interacted with using a lowcost blunt-tipped stylus with a capacitive tip.
- Using a stylus on a standard touch display may result in unintentional touches if users rest their hands on the display; such a situation calls for *palm rejection* techniques that discard interaction resulting from the hand based on shape or on the timing of finger and stylus input. There is
- also risk of losing the stylus.

- Beyond mobile devices, the availability of high-precision touchscreens has opened the door to many professional applications in banking, medical, or military systems.
- Furthermore, because touchscreens can be made to be very robust, they are particularly appropriate for
- public-access kiosks and mobile applications.
- Designers of public access systems value touchscreens because there are no moving parts and durability in high-use environments is good (the touchscreen is the only input device that has survived at Walt Disney World theme parks).
- Strategies have been described to provide access to touchscreen systems, such as for information kiosks or voting systems for people who are vision-impaired or blind, are hard of hearing or deaf, have trouble reading or are unable to read at all, or have physical disabilities For kiosk
- designs, arm fatigue can be a problem, which can be addressed by tilting the screen and providing a surface on which to rest the arm.
- On the other hand, kiosks are generally not used for extensive interactive sessions.
- In general, arm fatigue for mid-air or unsupported interaction can be measured using the Consumed Endurance metric, which is based on a biomechanical model of the arm.

Indirect-control pointing devices

- Indirect pointing devices separate the input (motor) space from the output (display) space, thus minimizing hand fatigue, by providing a surface for the hand to rest as well as eliminating hand-screen occlusion, by keeping the spaces apart.
- However, they require the hand to locate the device and also demand more cognitive processing
- and hand/eye coordination to bring the on-screen cursor to the desired target.
- The *mouse* is the most common indirect pointing device and is appealing because of its low cost and wide availability.
- While using a mouse, the hand rests in a comfortable position, buttons on the mouse are easy to press, long motions can be done rapidly by moving the forearm, and positioning can be done precisely with small finger movements. However, users must grab the mouse to begin

- work, desk space is consumed to operate it, and users must separate their attention between the motor and display space.
- Other problems are that pick-up and replace (also called *clutching*) actions are necessary for long motions and some practice is required to develop skills (usually from 5 to 50 minutes, but sometimes much
- more for older adults or users with disabilities).
- The variety in terms of mouse technologies (physical, optical, or acoustic), number of buttons, placement of the sensor, weight, and size indicates that the designers and users have yet to settle on one preferred design

- The *trackball* is controlled by spinning a ball along two axes and has sometimes been described as an upside-down mechanical mouse.
- It is usually implemented as a rotating ball, 1 to 15 centimeters in diameter, that moves a cursor on the screen as it is moved .
- The trackball is wear-resistant and can be firmly mounted in a desk to allow users to hit the ball vigorously and to make it spin.
- Trackballs have also been embedded in control panels for air-traffic-control or museum
- information systems, and they are commonly used in video game controllers

- The *joystick*, was initially used in aircraft-control devices and early computer games, has dozens of versions with varying stick lengths and thicknesses, displacement forces and distances, anchoring strategies for bases, and placement relative to the keyboard and screen.
- Joysticks are appealing for tracking purposes (to follow or guide an object on a screen), partly because of the relatively small displacements needed to move a cursor, the ease of direction changes, and the opportunity to combine the joystick with additional buttons, wheels, and triggers .
- The *directional pad* (or D-pad) originated in game consoles and consists of four directional arrows arranged in a cross with a trigger button in the center.
- The *pointing stick* (or *trackpoint*) is a small isometric joystick embedded in keyboards between the letters G, B, and H . It is sensitive to pressure and does not move.
- It has a rubber tip to facilitate finger contact, and with modest practice, users can quickly and accurately use it to control the cursor while keeping their fingers on the keyboard home position. The pointing stick is particularly effective for applications such as word processors that require constant switches between the keyboard and the pointing device.
- Because of their small size, pointing sticks can easily be combined with other devices such as keyboards

- **Touchpads** offer the convenience and precision of a touchscreen while keeping the user’s hand off the display surface.
- Users can make quick movements for long-distance traversals and can gently rock their fingers for precise positioning before lifting off.
- .

- The *graphics tablet* is a touch-sensitive surface separate from the screen, usually laid flat on the desk/table or in the user’s lap. This separation again allows for comfortable hand positioning and keeps the users’ hands off the screen. graphics tablet is appealing when users’ hands can remain with the device for long periods without switching to a keyboard. For this reason, graphics tablets are often popular with digital artists who engage in drawing and sketching operations. Furthermore, the graphics tablet permits adding application options, such as palettes, tools, and brushes, beyond the screen itself to its surface, thereby preserving valuable screen space and providing both guidance to novice users as well as easy access to experts. Graphics tablets are typically operated using a finger, pencil, puck, or stylus through acoustic, electronic, or contact position sensing

Fitts's Law

- It is the law of human hand movement.
- Often referred to simply as Fitts's Law (or even Fitts' Law), this micro-scale HCI theory allows designers to decide on the optimal locations and sizes of buttons and other elements when laying out screens as well as indicates which pointing devices are best suited to performing common tasks.
- Fitts noticed that the time required to complete hand movements was dependent on the distance users had to move, *D*,and the target size, *W*. Doubling the distance (say, from 10 cm to 20 cm) resulted in longer completion times, but not twice as long.
- Increasing the target's size (say, from 1 cm to 2 cm) enabled users to point at it more rapidly
- Since the time to start and stop moving is constant, an effective equation for the movement time (*MT*) for a given device, such as a mouse, turns out to be
- $MT = a + b \log (D/W + 1)$
- Where *a* approximates the start/stop time in seconds for a given device and *b* measures the inherent speed of the device. Both *a* and *b* need to be determined experimentally for each device. 200 msec/bit, *D* were 14 cm, and *W* were 2 cm, then the movement time *MT* would be 300 + 200 log2(14/2 + 1), which equals 900 milliseconds.

- 10.4 Reading assignment