INPUT DEVICES AND TECHNIQUES

Robert J.K. Jacob, Tufts University

INTRODUCTION

All aspects of human-computer interaction, from the high-level concerns of organizational context and system requirements to the conceptual, semantic, and syntactic levels of user interface design, are ultimately funneled through physical input and output actions and devices. This chapter considers the input half of this physical level of human-computer interaction, the final means by which the user communicates information to the computer. It is also called the Lexical level of the design of an interactive system, in contrast to the successively higher Syntactic, Semantic, and Conceptual levels [Foley, 1990a].

Computer input once consisted of such actions as setting switches and knobs and plugging and unplugging jumper wires in patch boards. For many years after that, the primary form of computer input was the punched card. Users or, more often, specialist keypunch operators punched the input information as holes in paper cards, which could then be read by computer peripherals. Next came the teletype, a device with a typewriter-like keyboard on which the user could type characters and cause corresponding electrical signals to be transmitted to the computer directly. Terminals, keyboards, and displays, the loose descendants of the teletype, continue to provide the principal form of computer input today.

Given the current state of the art, computer input and output are quite asymmetric. The amount of information or bandwidth that is communicated from computer to user is typically far greater than the bandwidth from user to computer. Graphics, animations, audio, and other media can output large amounts of information rapidly, but there are hardly any means of inputting comparably large amounts of information from the user. This is partly due to human abilities: we can receive visual images with very high bandwidth, but we are not very good at generating them. We can generate higher bandwidth with speech and gesture, but computers are not yet adept at interpreting these. User-computer dialogues are thus typically one-sided. New input devices and media can help redress this imbalance by obtaining data from the user conveniently and rapidly, but, relative to output, input has been a neglected field of research, particularly in comparison with the great strides made in computer graphics.

UNDERLYING PRINCIPLES

The fundamental task of computer input is to move information from the brain of the user into the computer. Progress in this area attempts to increase the useful bandwidth across that interface by seeking faster, more natural, and more convenient means for users to transmit information to computers. On the user's side of the communication channel, input is constrained by the nature of human communication organs and abilities; on the computer side, it is constrained only by the input devices and methods that we can invent. Research in input and output centers around the two ends of this channel: the devices and techniques computers can use for communicating with people, and the perceptual abilities, processes, and organs people can use for communicating with computers. It then attempts to find the common ground through which the two can be related by studying new modes of communication that could be used for human-computer communication and developing devices and techniques to use such modes. Basic research seeks theories and principles that can predict user performance in new situations to guide the search for input media and the design of interfaces. In principle, the development of new input/output devices ought to be motivated or guided by the studies of human perceptual facilities and effectors as well as the needs uncovered in studies of existing interfaces. More often, though, the hardware developments have come first, and then HCI researchers try to find uses for the resulting artifacts.

The challenge in this field is, thus, to design new devices and types of dialogues that better fit and exploit the communication-relevant characteristics of humans. In doing so, two significant goals are bandwidth and naturalness. Increasing bandwidth simply means communicating more information per unit of time, and, other things being equal, improves the efficiency of user-computer communication.

Naturalness

In seeking naturalness, we attempt to make the user's input actions as close as possible to the user's thoughts that motivated those actions, that is, to reduce the ''Gulf of Execution'' described by Hutchins, Hollan, and Norman [Hutchins, 1986a], the gap between the user's intentions and the actions necessary to input them into the computer. The motivation for doing this is that it builds on the equipment and skills humans have acquired through evolution and experience and exploits them for communicating with the computer. Direct manipulation interfaces [Shneiderman, 1983a] have enjoyed great success, particularly with new users, largely because they draw on analogies to existing human skills (pointing, grabbing, moving objects in space), rather than trained behaviors. Virtual reality interfaces, too, gain their strength by exploiting the user's pre-existing abilities and expectations. Navigating through a conventional computer system requires a set of learned, unnatural commands, such as keywords to be typed in, or function keys to be pressed. Navigating through a virtual reality system exploits the user's existing, natural ''navigational commands,'' such as positioning his or her head and eyes, turning his or her body, or walking toward something of interest. The result is to increase the user-to-computer bandwidth of the interface and to make it more natural, because interacting with it is more like interacting with the rest of the world.

Interaction Tasks, Techniques, and Devices

A designer looks at the **interaction tasks** necessary for a particular application [Foley, 1990a]. Interaction tasks are low-level primitive inputs required from the user, such as entering a text string or choosing a command. For each such task, the designer chooses an appropriate **interaction device** and **interaction technique.** An interaction technique is a way of using a physical device to perform an interaction task. There may be several different ways of using the same device to perform the same task. For example, one could use a mouse to select a command by using a pop-up menu, a fixed menu (palette or toolbox), multiple clicking, circling the desired command, or even writing the name of the command with the mouse. An interaction technique represents an abstraction of some common class of interactive task, such as choosing one of several objects shown on a display screen. Research in interaction techniques studies these primitive elements of human-computer dialogues, which apply across a wide variety of individual applications. Its goal is to add new, high-bandwidth methods to the available store of interaction techniques or dialogue components. While the interaction techniques are specific artifacts that can be applied directly in practical applications, the most useful of them are general enough to be used in a variety of application, such as the pop-up menu.

In selecting an interaction device and technique for each task in a human-computer interface, simply making an optimal choice for each task individually may lead to a poor overall design, with too many different or inconsistent types of devices or dialogues. Therefore, it is often desirable to compromise on the individual choices to reach a better overall design, not only to avoid surrounding the user with an array of rarely-used devices but also to reduce the time penalty incurred in switching between devices. In some situations, the designer has broad freedom to choose input devices appropriate to the task. For example, the cockpit of a new airplane or a military command and control console or a control station for a surgical teleoperator can be outfitted with whatever devices best facilitate operator performance. In many other situations, the designer of a human-computer interface does not have much control over the input hardware environment; he or she might be designing a piece of software to be used on a standard workstation with a standard or widely available suite of standard input devices devices, usually a keyboard and mouse. In this case, the designer decides which tasks should be assigned to the mouse and which to the keyboard (or possibly provides the user with synonyms) and which interaction techniques should be used for each task. Here too the time penalty for switching the hand from one device to another is a factor: while the keyboard might be the optimal input device for choosing a number between one and five, if this task occurs between two mouse tasks, it may be better to provide a graphical menu to save the switching time between devices. With elegant design, it is sometimes possible to provide both interaction techniques as options to the user, without compromising the integrity of either.

In many situations, there are additional constraints on the range of input devices and interaction techniques the designer can choose. For example, an interface for a fighter airplane pilot must take into account the fact that the hands are usually already occupied with the task of operating the plane. Users operating under large gravity forces, under the ocean, on a rolling ship, or wearing a bulky spacesuit may impose additional constraints on the input methods one can choose. Interface design for handicapped users constrains the range of input choices in an analogous fashion. In each case, the best choice may not be the same as what would be chosen in an unconstrained situation.

Fitts' Law

User performance with many types of manual input depends on the speed with which the user can move his or her hand to a target. **Fitts' Law** provides a way to predict this, and is a key foundation in input design [Card, 1983a]. It predicts the time required to move based on the distance to be moved and the size of the destination target. The time is proportional to the logarithm of of the distance divided by the target width. This leads to a tradeoff between distance and target width: it takes as much additional time to reach a target that is twice as far away as it does to reach one that is half as large. Different manual input devices give rise to different proportionality constants in the equation. Some thus give better overall performance, and others, better performance either for long moves or short moves, but the one-for-one tradeoff between distance and target size remains.

Control-Display Ratio

Another way of characterizing many input devices is by their **control-display ratio.** This is the ratio between the movement of the input device and the corresponding movement of the object it controls. For example, if a mouse (the control) must be moved one inch on the desk in order to move a cursor two inches on the screen (the display), the device has a 1:2 control-display ratio. A high control-display ratio affords greater precision, while a low one allows more rapid operation and takes less desk space. An accelerator can be provided, so that the ratio increases dynamically when the user moves faster. This allows more efficient use of desk space, but can disturb the otherwise straightforward physical relationship between mouse movement and cursor movement [Jellinek, 1990a]. Of course, with a direct input device, such as a touch screen, the C-D ratio is always unity.

BEST PRACTICES

This section surveys the principal types of interaction devices in use today and emerging. Where possible, it is structured around the ‘‘output’’ mechanisms of the user’s body rather than the device technology, since the former are more likely to remain constant over time. The principal means of human output or computer input today is through the user’s hands, for example keyboards, mice, gloves, and 3D trackers; these are discussed first. Other limb movements are then considered, followed by voice, and, finally, eye movements and other physiological measurements that may be used as input in the future.

Hands — Discrete Input

Keyboards, attached to workstations, terminals, or portable computers are one of the principal input devices in use today. Nearly all of them now use a typewriter-like ‘‘QWERTY’’ keyboard layout, typically augmented with additional keys for moving the cursor, entering numbers, and special functions. Alternative keyboard layouts, such as the Dvorak layout, claim higher typing speed, but they have not been widely accepted because of the pervasiveness of the QWERTY layout. Another alternative that has been introduced is to retain the same assignment of letters to keys but to change the geometrical arrangement of the keys, in order to reduce strain on the hand and wrist during typing. Such a keyboard is typically divided into two halves, one for each hand, and these are pivoted away from each other, toward the respective hands; they may also be sloped upward in the center, better to fit the natural position of the hands.

Today's standard keyboard is widespread and relatively inexpensive to construct. As a result, it has been difficult to displace as the primary means of computer input. In recent years, the chief force serving to displace it has been the shrinking size of computers, as laptops, notebooks, palmtops, and personal digital assistants are being developed. The typewriter keyboard is becoming the largest component of such pocket-sized devices, the one component standing in the way of reducing its overall size, and this is beginning to provide a new driving force for developing alternatives to the keyboard.

As a computer peripheral, the keyboard simply transmits a signal each time a key is depressed (and, possibly, another signal when it is released). Some keyboards transmit a code for the character itself, that is, ''a'' if the a key is pressed, and ''A'' if it is depressed while holding the Shift key. Other keyboards transmit unencoded signals, that is, an individual signal for the pressing or releasing of each button on the keyboard: the a key would be transmitted as a pressing of the second key in the third row, for example; and a capital A would be transmitted as a sequence of raw events—the pressing of the Shift key, pressing of the a key, releasing of the Shift key, and so on. Encoding this sequence into a capital A would then be done in the computer. This approach provides the flexibility to define new encodings, new types of shift keys, and new key chord combinations within the software.

Chord Keyboard

Another type of keyboard is the chord keyboard. This is typically designed for one hand and has five keys, one for each finger, plus sometimes additional ones for the thumb (see Figure 1). Instead of pressing single keys, the user can press any combination of keys as a single chord. With five keys, this allows 31 combinations. The chord keyboard was originally introduced along with the mouse, with the intention that the user would use a mouse in the right (or dominant) hand and the one-hand chord keyboard in the other hand [Engelbart, 1968a]. While the mouse has since won widespread acceptance, the chord keyboard has not. Again, as computers become smaller, the benefit of a keyboard that allows touch typing with only five keys may come to outweigh the additional difficulty of learning the chords.

Function Keys

Hardware similar to that of the alphanumeric keyboard may also be used to provide individual, dedicated keys to invoke specific computer commands. These may be permanently labelled, special-purpose pushbuttons, or they may have labels that can be changed under computer control. An extreme, but effective use of permanently labeled function keys is found in the cash registers of fast-food restaurants, where a large array of special-purpose function keys is provided, one for every possible item that can be purchased. Variable labels for function keys can be provided by placing the keys near the edge of the display and using the adjacent portion of the display to indicate the key labels, or by providing small alphanumeric LED or LCD displays above each key. Variable labels can even be provided by a CRT display projected downward onto half-silvered mirror, so that the labels drawn on the CRT appear to float over the otherwise blank function keys [Knowlton, 1977a].

Hands — Continuous Input

While keyboards and their variants are the principal means of discrete manual input, a much wider variety of devices is in use for continuous input from the hands. In fact, a number of taxonomies have been proposed for organizing and understanding continuous, manual input devices. The first approaches centered around the idea of logical devices, where devices are grouped by the type of input they provide: for example, locator, string, valuator, choice [Foley, 1984a]. Another approach organizes continuous input devices by property and the number of dimensions sensed [Buxton, 1983a]. More recent approaches attempt to incorporate more of the ergonomic differences between seemingly similar devices into the taxonomy, to help guide the selection of the appropriate device for a task [Mackinlay, 1990a, Bleser, 1990a].

Based on these approaches, devices used for manually-operated continuous pointing or locating can be categorized along each of the following dimensions:

- *Type of motion: Linear vs. Rotary.* For example, a mouse measures linear motion (in two dimensions); a knob, rotary.
- *Absolute or Relative measurement.* For example, a mouse measures **relative** motion; a Polhemus magnetic tracker, **absolute.**
- *Physical property sensed: Position or Force.* A mouse measures position; an isometric joystick, force. For a rotary device, the corresponding properties are angle and torque.
- *Number of dimensions: one, two, or three linear and/or one, two, or three angular.* A mouse measures two linear dimensions; a knob measure one angular dimension; and a Polhemus measures three linear dimensions and three angular.
- *Direct vs. Indirect control.* A mouse is **indirect** (you move it on the table to point to a spot on the screen); a touch screen is **direct** (you touch the desired spot on the

screen directly).

- *Position vs. Rate control.* Moving a mouse changes the position of the cursor; moving a rate-control joystick changes the speed with which the cursor moves.
- *Integral vs. Separable dimensions.* A mouse allows easy, coordinated movement across two dimensions simultaneously (integral); while a pair of knobs (as in an Etch-a-Sketch toy) does not (separable) [Jacob, 1994a].

This covers the range of continuous, manually-operated devices. Discrete input devices, such as the keyboard, can be fit into the above (and some of the taxonomies discussed also cover discrete devices), but they do not appear in the variety that continuous devices do. Non-manually-operated devices fit less well into the above categories; they currently include foot or other body controls, voice input, eye trackers, and a variety of other physiological measuring instruments discussed later.

Given this space of possible continuous manual input devices, we discuss next some of the more common forms of devices currently in use.

One-Dimensional Valuator

A rotary (knob or thumbwheel) or linear (slide) potentiometer may be used for inputting a value along a single axis. Its analogue output is converted to a digital computer input each time the computer queries the associated A-D converter. A knob with a digital encoder may also be used. It simply transmits a interrupt signal to the computer each time it is turned a small amount. Unlike an analogue potentiometer, such a device typically does not have physical endpoints, but turns continuously. The range and meaning of knob movement can be thus arbitrarily modified by the software. In some applications that involve multiple parameters, a single ‘‘soft pot’’ of this type is provided. at any moment, just one of the parameters in the application is chosen to be assigned to this knob and may be adjusted. A dial box is sometimes used in computer graphics; it is an array of several such knobs, often provided with computer-controlled labels. Other input devices may also be used for task of entering a scalar value, through the use of interaction techniques. For example, a mouse may be used for this job via an on-screen slider; a keyboard may be used by typing in a numeric value.

Two-Dimensional Locator

Today, the mouse is the most widely used device for inputting 2-D positions, but it was not the first such device developed. It supplanted devices such as the joystick, trackball, light-pen, and arrow keys and, in an early example of the application of HCI research to practice, was demonstrated to give fastest performance and closest approximation to Fitts' Law compared to alternative devices at the time [Card, 1978a]. Despite its popularity, some specific, constrained situations call for alternative devices. For example, the Navy uses trackballs instead of mice on shipboard, because the rolling of the ship makes it difficult to keep a mouse in place. Portable computers use a small trackballs, touch-sensitive pads, or tiny joysticks because they are more compact.

The mouse and trackball are relative devices, they report only how far they move, not where they are. They typically generate an interrupt or a piece of serial data each time they move. The data tablet is an absolute locator device that is similar to the mouse in appearance, but the surface upon which it operates contains a grid of wires or other sensors that can measure the absolute position of a puck or stylus upon the tablet and report it when queried or else in a continuous stream. It is most often seen in graphics and CAD applications.

The joystick comes in several varieties. It can be used to control cursor position directly or it can control the rate of speed at which the cursor moves. Since its total range of motion is typically fairly small compared to a display screen, position control is imprecise; however, rate control requires a more complex relationship between the user's action and the result on the display and is therefore more difficult to operate. The joystick can move when it is pushed or, in an isometric joystick, it can remain nearly stationary and simply report the force being applied to it.

Direct input devices obviate the need to relate the position of the device to the position of the cursor on the screen. A touch screen is a device that fits over a CRT or other display and reports the location of finger or stylus touches on its surface. With it, the user can simply point to the desired item on the screen. This requires very little training, but it can be tiring if the user must hold his or her hand up to the screen for a long time. Precision of touchscreens is typically lower than that of other locator devices, though new strategies have been developed to improve the precision attainable with a finger-operated touch screen [Sears, 1991a]. A finger-operated touch screen can also be used to simulate a keyboard, and allows the ‘‘keys’’ to be relabelled under computer control. However, such a keyboard lacks the tactile feedback of a conventional keyboard, making it slower to operate and particularly poor choice for eyes-busy applications such as operating a car or airplane. The light pen was a technology used in early graphics systems that also allowed direct pointing on the screen with a light-sensitive stylus.

Modern pen-based systems use technology similar to that of a touchscreen or data tablet, but they typically use the pen as the sole input device. It is used both for location input and for character string input, and, more interestingly, it can also be used for interaction that more closely resembles the way a person would use a regular pen rather than a mouse, such as making circle and arrow gestures to move blocks of text. For entering text, full handwriting recognition is not yet achievable for all users. Block printing of capital letters is possible, but fairly cumbersome. For some users, a compromise works better: using an alphabet of characters specially designed to be easily distinguishable from one another to facilitate computer recognition. Such characters are also typically designed so that each can be drawn with a single stroke without lifting the pen, which makes it easier for the computer to find the boundaries between the letters. It also makes it possible to use a very small input area, in which the input letters are written in succession, on top of one another, for some applications.

Three-Dimensional Locator

The typical 3-D locator device functions like the three-dimensional equivalent of a data tablet in that it provides absolute position information along three axes in space, instead of two, either continuously in a stream or each time it is queried. Many such devices also report their orientation, in the form of angles of rotation about the three axes, or yaw, pitch, and roll. The most common such devices (Polhemus and Ascension) use a magnetic signal that is transmitted by a fixed source and received by a sensor held in the user's hand or attached to some object (see Figure 2). Ultrasonic ranging (Logitech) is also used for this purpose. It typically provides less precision, but is more robust in the face of magnetic interference, such as that from a CRT.

While often operated with the hand, the sensor of the 3-D tracker is typically a one-inch plastic cube, which can be used in a variety of ways. It can be held in the hand, or attached to a glove, foot, the user's head (as is typically done in virtual reality), or to passive props [Hinckley, 1994a] or other objects the user will manipulate. A hybrid form of 3-D tracker combines a mouse in a single package and allows it to be operated as a mouse while it is located on a table, but switches into 3-D operation when it is lifted into the air.

Today, all of these 3-D devices are still limited compared to a mouse or data tablet—in latency, precision, stability, susceptibility to interference, or number of available samples per second. In addition, they all require that the user hold or attach the small sensor and its trailing wire. Another approach is to use sensors that observe the user, without requiring him or her to hold or wear anything. Camera-based locator devices offer the promise of doing this, but today are still limited. A single-camera system is limited to its line of sight; more cameras can be added but full coverage of an area may require many cameras and a way to switch among them smoothly. This approach depends upon some type of image processing to interpret the picture of the user and extract the desired hand or body position. Small video cameras are beginning to appear as a standard component of graphics workstations; while they are intended for teleconferencing, they will also be useful for this type of 3-D input.

Another 3-D input device is the Spaceball, which is roughly the 3-D analogue of an isometric joystick (see Figure 3). It consists of a ball mounted on a fixed platform; the user holds the ball and pushes or twists it in the desired direction. Finally, note that 3-D input can also be achieved with a device referred to as a 3-D joystick, which is really a 2-D joystick with an additional input device attached to it, typically a knob that can be rotated on the end of the joystick, to provide the third input dimension.

Gesture

Hand gesture is a form of input that is still emerging. The devices used are the same 3-D trackers discussed, including magnetic and camera-based devices. However, rather than using them simply to designate a location in three-space, they can allow a user to make natural, continuous gestures in space. This requires not only a better, non-encumbering three-dimensional tracking technology but also a way to recognize human gestures occurring dynamically. Gestures are typically made with poor precision and repeatability, so a useful input technique would have to tolerate such imprecision and still glean the user's intended action. The same issues arise in using two-dimensional gestures on a surface, for pen-based interfaces.

Glove

Glove input devices report the configuration of the fingers of the user's hand, also called a hand ‘‘posture’’ in contrast to a ‘‘gesture,’’ which may involve motion or a sequence of different postures to convey meaning (see Figure 4). The Dataglove uses optical fibers, which attenuate light when bent. Other glove technologies use mechanical sensors. All of these devices typically report a vector containing the bend angle of each of the joints of each finger of the hand. Some also report abduction, the angles formed by the separation of the fingers from each other. Most glove devices combine a 3-D tracker, so that they can report the position and orientation of the hand as well as the angle of each finger. From these, it should in principle be possible to derive the exact position in space of each fingertip; however, the accuracy of today's glove device is does not always allow this.

Two-Handed Input

Aside from touch typing, most of the devices and modes of operation discussed thus far and in use today involve only one hand at a time. People are quite good at manipulating both hands in coordinated or separated tasks, as for example one does in driving a car, piloting an airplane, or performing surgery [Buxton, 1986a]. For example, a two-handed approach that simulates the use of a moveable, translucent stencil has been demonstrated to be effective for desktop tasks [Stone, 1994a].

Other Body Movements

Having considered input from the hand, we consider next other limbs and body movements that can be used as computer input, though, today, they are not nearly as widely used as manual input.

Foot

Simple foot controls are used in automobiles and musical instruments, and can readily be used as computer input for discrete or continuous scalar information, using simple input devices. The Mole is a more sophisticated foot-operated input device that provides locator input using a footrest suspended on two sets of pivots [Pearson, 1986a]. While control is less precise than a manually-operated mouse, it leaves the hands free for additional operations.

Head

Head movement can be measured with a 3-D tracker and can be used to control cursor position, though this can often require the neck to be held in an awkward fixed position. Another use of head movement is to perform a function more akin to the use of head movement in the natural world—panning and zooming over a display [Hix, 1995a]

Input for Virtual Reality

Most virtual reality systems rely on the same 3-D devices discussed above, used in combination. They use a 3-D magnetic tracker to sense head position and orientation, which then determines the position of the virtual camera, which generates the scene to be displayed in the user's head-mounted display, typically in stereo. The result is the illusion of a realistic, three-dimensional world that surrounds the user wherever he or she looks. The user can reach out into this world and touch the objects in it, using a second 3-D tracker attached to the hand (so the computer knows where the user's hand is relative to the displayed world) and, often, a glove (so the computer can detect grasping or other gestures). However, the user will not feel the object when his or her hand touches it. Mechanisms for providing computer-controlled force and tactile feedback are a topic of current research. An extension of this notion would be to provide virtual tools for input, where the user might first obtain a tool (by reaching for it in the virtual space) and then apply it to a three-dimensional virtual object. A virtual tool, can of course, metamorphose as needed for the job at hand and otherwise improve upon the properties of non-virtual tools. However, the latency and precision available from today's input devices still fall short of being able to support this smoothly.

Facial Expression

A less obvious form of muscle input is to use the facial expressions of the user. The device for doing this is simply a camera and frame grabber, but image understanding techniques for interpreting the images into meaningful facial expressions are still emerging. However much less subtle inputs are also possible. For example the computer can determine relatively easily from camera input whether the user is still sitting in the chair, facing toward the computer or not, using the telephone, or talking to another person in the room.

Myoelectric Inputs

Beyond physical measurement of limb motions, an emerging technology for muscle input is to measure myoelectric signals from electrode placed on the user's skin. While currently a research topic, this approach has the potential to provide a more compact, less cumbersome way to measure muscle movements. Such signals can also be detected slightly before the muscle actually begins moving, which can help to reduce overall system latency.

Voice

Another type of input comes from the user's speech. Carrying on a full conversation with a computer as one might do with another person is well beyond the state of the art today—and, even if possible, may be a naive goal. Nevertheless, speech can be used as input in several different ways: unrecognized speech, discrete word recognition, and continuous speech recognition.

Unrecognized Speech

Even without understanding the content of the speech, computers can digitize, store, edit, and replay segments of speech in useful ways. Conventional voice mail is an everyday example of this type of function, but far more sophisticated uses of this technology have been developed [Schmandt, 1993a].

Discrete Word

Understanding speech as input has been a long standing area of research. While progress is being made, it is slower than optimists originally predicted, and further work remains in this field. Although the goal of continuous speech recognition remains elusive, unnatural, isolated-word speech recognition can work reasonably well and is appropriate for some tasks. Discrete word recognition requires that the user pause briefly after saying each word. It is a highly unnatural way of speaking, though it can seem appropriate for giving computer commands. Some systems are speaker-dependent (they require each particular user to speak the words to be used into the system ahead of time to ‘‘train’’ the computer), while some are speaker-independent (they rely on a single set of training data for all users). Performance can also be enhanced by using a restricted grammar. For example, if the first word of each command must be a verb and the second must be a file name, the speech recognizer can use this information to limit the range of possibilities it must examine at each point and thereby provide more accurate results.

Continuous Speech

One of the most difficult aspects of recognizing continuous speech is simply finding the boundaries between the words. Research continues in the area of continuous speech recognition, with varying degrees of success found in both research and commercial systems. Improved performance can be obtained where the system can be tuned to a particular application domain and input grammar.

Even if the computer could recognize all the user's words, the problem of understanding natural language is a significant and unsolved one. It can be avoided by using an artificial language of special commands or even a fairly restricted subset of natural language. But, given the current state of the art, the closer the user moves toward full unrestricted natural language, the more difficulties will be encountered.

Multi-Mode Speech Input

Speech is often most useful in conjunction with other input media, providing an additional channel when the user is already occupied. (Driving a car and conducting a conversation is an everyday example.) If the user's hands, feet, and eyes are busy, speech may be the only reasonable choice for some input. However, more interesting cases begin with a collection of tasks in a user interface and then allocate them to the range of the user's communication modes. Another use for multiple modes is to combine otherwise ambiguous inputs from several modes (such as pointing and speaking) to yield an unambiguous interpretation of the user's input [Schmandt, 1982a].

Eye

While the main role of the eye in most human-computer interaction situations is to receive output from the computer, the movements of the user's eye can also be measured and used as input. A eye tracker can measure the visual line of gaze, that is, where the user's eye is pointing in space, and report it to a computer in real time (see Figure 5). Eye movement-based input, properly used, can provide an unusually fast and natural means of communication, because we move our eyes rapidly and almost unconsciously. However, eye tracking technology today is still only marginally adequate for use in applications; its prime application area is for disabled users, who cannot move their arms or legs.

Using eye movements as input also requires careful design of interaction techniques [Bolt, 1981a, Jacob, 1991a]. Eye movements, like other passive inputs discussed below, are often non-intentional or not conscious, so they must be interpreted carefully to avoid annoying the user with unwanted responses to his actions, the ‘‘Midas Touch’’ problem. People are not accustomed to operating devices simply by moving their eyes. They expect to be able to look at an item without having the look cause an action to occur. At first it is helpful to be able simply to look at what you want and have it occur without further action; soon, though, it becomes like the Midas Touch. Everywhere you look, another command is activated; you cannot look anywhere without issuing a command. Eye movements are an example of the ‘‘clutch’’ problem that arises in how many emerging passive or non-command forms of input (including speech, gesture, physiological measurement)—it requires a way to tell the computer when the device is ‘‘engaged’’ vs. when the user is using the same communication modality for some other purpose, but not ‘‘talking to’’ the computer.

Passive Measurements

User Behavior

Input may also be obtained from a user without explicit action on his or her part. Behavioral measurements can be made from changes the user’s typing speed, general response speed, manner of moving the cursor, frequency of low-level errors, or other patterns of use. A carefully designed user interface could make intelligent use of such information to modify its dialogue with the user, based on, for example, inferences about the user’s alertness or expertise (but note that there is also the potential for abuse of this information). These measures do not require additional input devices, but rather gleaning of additional, typically neglected information from the existing input stream.

Physiological Measurements

In a similar vein, passive measurements of the user's state may also be made with additional hardware devices. In addition to three-dimensional position tracking and eye tracking, a variety of other physiological characteristics of the user might be measured and the information used to modify the computer's dialogue with its user. Blood pressure, heart rate, respiration rate, eye pupil diameter, and galvanic skin response (the electrical resistance of the skin) are examples of measurements that are relatively easy and comfortable to make, although their accurate instantaneous interpretation within a user-computer dialogue is an open question.

A more difficult measure is an electro-encephalogram, although progress has been made in identifying specific evoked potential signals in real time [Wickens, 1983a]. The most accurate results are currently obtained with a somewhat unwieldy superconducting detector [Lewis, 1987a], rather than the conventional electrodes, but improvements in this technology can be envisioned.

Direct Connect

Looking well beyond the current state of the art, perhaps the final frontier in user input and output devices will be to measure and stimulate neurons directly, rather than relying on the body's transducers. This is unrealistic at present, but it may someday be a primary mode of high-performance user-computer interaction. If we view input in HCI as moving information from the brain of the user into the computer, we can see that all current methods require that this be done through the intermediary of some physical action. We strive to reduce the Gulf of Execution, the gap between what the user is thinking and the physical action he or she must make to communicate that thought. From this point of view, reducing or eliminating the intermediate physical action ought to improve the effectiveness of the communication. The long-term goal might be to see the computer as a sort of mental prosthesis, where the explicit input and output steps vanish and the communication is direct, from brain to computer.

Other Issues

Relationship to Output

While this chapter discusses input, it should be clear that many of the newer approaches here are intimately coupled with output. Input devices and their technologies are important, but increasingly are meaningful only in context of outputs, especially in more modern, highly-interactive forms of interaction. For example, while a keyboard makes sense as an isolated input device, a pop-up menu makes sense only when the mouse input and screen output are considered together. In a direct manipulation or graphical interface, the output objects on the display are the principal targets for subsequent input commands, which select and manipulate the displayed objects. Similarly, virtual reality makes sense only when the input from head and hand positions sensors controls the moment-to-moment output transmitted to the head-mounted display.

Device Interfaces

A mundane but nagging problem in the area of input is connecting new input devices to a computer. New devices often introduce new, slightly different hardware connections and software protocols for communication. Even superficially similar devices are not yet easily interchangeable and often require essential, but fundamentally trivial work to begin using a new device. The communication requirements of many of the input devices discussed here are sufficiently similar and undemanding that a standard physical interface and communication protocol is not a serious technical problem nor would it levy an unreasonable performance penalty. For example, the MIDI standard interface addresses this problem for both physical connection and simple logical protocol for keyboard-oriented musical instruments, and its dramatic success in expanding the usefulness of electronic musical instruments suggests the benefits.

RESEARCH ISSUES AND SUMMARY: FUTURE TRENDS

Interaction Style

A new style of interaction that is emerging is ''non-command-based'' interaction [Nielsen, 1993a]. While other interaction styles await, receive, and respond to explicit inputs from the user, in this approach the computer passively monitors the user and responds as appropriate. Its effect on the field of input is to move from providing objects for the user to actuate through specific commands to simply sensing the user's body. Jakob Nielsen describes this next generation interaction style:

> ''The fifth generation user interface paradigm seems to be centered around non-command-based dialogues. This term is a somewhat negative way of characterizing a new form of interaction but so far, the unifying concept does seem to be exactly the abandonment of the principle underlying all earlier paradigms: That a dialogue has to be controlled by specific and precise commands issued by the user and processed and replied to by the computer. The new interfaces are often not even dialogues in the traditional meaning of the word, even though they obviously can be analyzed as having some dialogue content at some level since they do involve the exchange of information between a user and a computer. The principles shown at CHI'90 which I am summarizing as being non-command-based interaction are eye tracking interfaces, artificial realities, play-along music accompaniment, and agents.'' [Nielsen, 1990a]

This new interaction style will require new devices, interaction techniques, and software approaches to deal with them. Unlike traditional inputs, such as keyboards and mice, the new inputs represent less the intentional actuation of a device or issuance of a command, but are more like passive monitoring of the user. This suggests a change from conventional devices to passive equipment that senses the user, such as unobtrusive three-dimensional trackers, hand-measuring devices, remote cameras (plus appropriate pattern recognition), range cameras, eye movement monitors, and physiological monitors.

Interaction Devices

One clear current need is for 3-D tracking that with greater accuracy and lower latency than current techniques. A method that freed the user from the wire would also be helpful. Camera-based techniques may solve this problem, or a new technology may be applied to it; both are areas of current research.

Beyond this, we might predict the future of input by looking at some of the characteristics of emerging new computers. The desktop workstation seems to be an artifact of past technology in display devices and in electronic hardware. In the future, it is likely that computers smaller and larger than today's workstation will appear, and the workstation-size machine may disappear. This will be a force driving the design and adoption of future input mechanisms. Small computers are already appearing—laptop and palmtop machines, personal digital assistants, wearable computers, and the like. These are often intended to blend more closely into the user's other daily activities. They will certainly require smaller input devices, and may also require more unobtrusive input mechanisms, if they are to be used in settings where the user is simultaneously engaged in other tasks, such as talking to people or repairing a piece of machinery.

At the same time, computers will be getting larger. As display technology improves, as more of the tasks one does become computer-based, and as people working in groups use computers for collaborative work, a office-sized computer can be envisioned, with a display that is as large as a desk or wall (and has resolution approaching that of a paper desk). Such a computer leaves considerable freedom for possible input means. If it is a large, fixed installation, then it could accommodate a special-purpose console or ‘‘cockpit’’ for high-performance interaction. It might also be used in a mode where the large display is fixed, but the user or users move about the room, interacting with each other and with other objects in the room. In that case, while the display may be very large, the input devices would be small and mobile.

Another trend seen in the emergence of virtual reality, is that computer input and output is becoming more like interacting with the real world. Instead of inputting strings of characters, users interact with a virtual reality in more natural and expressive ways—moving their heads, hands, or feet. Future input mechanisms might continue this trend toward naturalness and expressivity by allowing users to perform ‘‘natural’’ gestures or operations and transducing them for computer input. More parts or characteristics of the user’s body might be measured for this purpose and then interpreted as input. As a thought experiment along these lines, consider obtaining and interpreting input from the gestures and actions of an orchestra conductor.

Another way to predict the future of computer input devices is to examine the progression that begins with experimental devices used in the laboratory to measure some physical attribute of a person. As such devices become more robust, they may be used as practical medical instruments outside the laboratory. As they become convenient, non-invasive, and inexpensive, they may find use as future computer input devices. The eye tracker is such an example; the physiological monitoring devices discussed above may well also turn out to follow this progression.

Finally, in a more practical vein, it is important to remember that there has historically been a long time lag between invention and widespread use of new input or output technologies. Consider the mouse, one of the more successful innovations in input devices, first developed around 1968 [Engelbart, 1968a]. It took approximately ten years before it was found widely even in very many other research labs and perhaps twenty before it was widely used in applications outside the research world. The input mechanisms in use twenty years from now may well be chosen from some of the devices and approaches that today appear to be impractical laboratory curiosities.

DEFINING TERMS

Absolute input device: An input device that reports its actual position, rather than relative movement. A data tablet or Polhemus tracker operates this way (see Relative input device).

Control-display ratio: The ratio between the movement a user must make with an input device and the resulting movement obtained on the display. With a large control-display ratio, a large movement is required to affect a small change on the display, affording greater precision. A low ratio allows more rapid operation and takes less desk space.

Direct input device: A device that the user operates directly upon the screen or other display to be controlled, such as a touch screen or light pen (see Indirect input device).

Fitts' Law: A model that predicts time to move the hand or other limb to a target, based on the distance to be moved and the size of the target. The time is proportional to the logarithm of the distance divided by the target width, with constant terms that vary from one device to another.

Indirect input device: A device that the user operates by moving a control that is located away from the screen or other display to be controlled, such as a mouse or trackball (see Direct input device).

Interaction device: A hardware computer peripheral through which the user interacts with the computer.

Interaction task: A low-level primitive input to be obtained from the user, such as entering a text string or choosing a command.

Interaction technique: A particular way of using a physical device to perform a generic interaction task. For example, the pop-up menu is an interaction technique for choosing a command or other item from a small set, by means of a mouse and display.

Relative input device: An input device that reports its distance and direction of movement each time it is moved, but cannot report its absolute position. A mouse operates this way (see Absolute input device).

References

Bleser1990a.

T.W. Bleser and J.L. Sibert, ‘‘Toto: A Tool for Selecting Interaction Techniques,’’ *Proc. ACM UIST’90 Symposium on User Interface Software and Technology*, pp. 135-142, Addison-Wesley/ACM Press, Snowbird, Utah, 1990.

Bolt1981a.

R.A. Bolt, ‘‘Gaze-Orchestrated Dynamic Windows,’’ *Computer Graphics*, vol. 15, no. 3, pp. 109-119, August 1981.

Buxton1983a.

W. Buxton, ‘‘Lexical and Pragmatic Considerations of Input Structures,’’ *Computer Graphics*, vol. 17, no. 1, pp. 31-37, 1983.

Buxton1986a.

W. Buxton and B.A. Myers, ‘‘A Study in Two-Handed Input,’’ *Proc. ACM CHI’86 Human Factors in Computing Systems Conference*, pp. 321-326, 1986.

Card1978a.

S.K. Card, W.K. English, and B.J. Burr, ‘‘Evaluation of Mouse, Rate-controlled Isometric Joystick, Step Keys, and Text Keys for Text Selection on a CRT,’’ *Ergonomics*, vol. 21, no. 8, pp. 601-613, 1978.

Card1983a.

S.K. Card, T.P. Moran, and A. Newell, *The Psychology of Human-Computer Interaction*, Lawrence Erlbaum, Hillsdale, N.J., 1983.

Engelbart1968a.

D.C. Engelbart and W.K. English, ‘‘A Research Center for Augmenting Human Intellect,’’ *Proc. 1968 Fall Joint Computer Conference*, pp. 395-410, AFIPS, 1968.

Foley1984a.

J.D. Foley, V.L. Wallace, and P. Chan, ‘‘The Human Factors of Computer Graphics Interaction Techniques,’’ *IEEE Computer Graphics and Applications*, vol. 4, no. 11, pp. 13-48, 1984.

Foley1990a.

J.D. Foley, A. van Dam, S.K. Feiner, and J.F. Hughes, *Computer Graphics: Principles and Practice*, Addison-Wesley, Reading, Mass., 1990.

Hinckley1994a.

K. Hinckley, R. Pausch, J.C. Goble, and N.F. Kassell, ‘‘Passive Real-World Interface Props for Neurosurgical Visualization,’’ *Proc. ACM CHI’94 Human Factors in Computing Systems Conference*, pp. 452-458, Addison-Wesley/ACM Press, 1994.

Hix1995a.

D. Hix, J.N. Templeman, and R.J.K. Jacob, ‘‘Pre-Screen Projection: From Concept to Testing of a New Interaction Technique,’’ *Proc. ACM CHI’95 Human Factors in Computing Systems Conference*, pp. 226-233, Addison-Wesley/ACM Press, 1995. http://www.eecs.tufts.edu/~jacob/papers/chi95.html [HTML].

Hutchins1986a.

E.L. Hutchins, J.D. Hollan, and D.A. Norman, ‘‘Direct Manipulation Interfaces,’’ in *User Centered System Design: New Perspectives on Human-computer Interaction*, ed. by D.A. Norman and S.W. Draper, pp. 87-124, Lawrence Erlbaum, Hillsdale, N.J., 1986.

Jacob1991a.

R.J.K. Jacob, ‘‘The Use of Eye Movements in Human-Computer Interaction Techniques: What You Look At is What You Get,’’ *ACM Transactions on Information Systems*, vol. 9, no. 3, pp. 152-169, April 1991.

Jacob1994a.

R.J.K. Jacob, L.E. Sibert, D.C. McFarlane, and M.P. Mullen, Jr., ‘‘Integrality and Separability of Input Devices,’’ *ACM Transactions on Computer-Human Interaction*, vol. 1, no. 1, pp. 3-26, March 1994. http://www.eecs.tufts.edu/˜jacob/papers/tochi.txt [ASCII]; http://www.eecs.tufts.edu/˜jacob/papers/tochi.pdf [PDF].

Jellinek1990a.

H.D. Jellinek and S.K. Card, ‘‘Powermice and User Performance,’’ *Proc. ACM CHI’90 Human Factors in Computing Systems Conference*, pp. 213-220, Addison-Wesley/ACM Press, 1990.

Knowlton1977a.

K.C. Knowlton, ‘‘Computer Displays Optically Superimposed on Input Devices,’’ *Bell System Technical Journal*, vol. 56, no. 3, pp. 367-383, March 1977.

Lewis1987a.

G.W. Lewis, L.J. Trejo, P. Nunez, H. Weinberg, and P. Naitoh, ‘‘Evoked Neuromagnetic Fields: Implications for Indexing Performance,’’ *Biomagnetism 1987, Proc. of the Sixth International Conference on Biomagnetism*, Tokyo, 1987.

Mackinlay1990a.

J.D. Mackinlay, S.K. Card, and G.G. Robertson, ‘‘A Semantic Analysis of the Design Space of Input Devices,’’ *Human-Computer Interaction*, vol. 5, pp. 145-190, 1990.

Nielsen1990a.

J. Nielsen, ‘‘Trip Report: CHI’90,’’ *SIGCHI Bulletin*, vol. 22, no. 2, pp. 20-25, 1990.

Nielsen1993a.

J. Nielsen, ‘‘Noncommand User Interfaces,’’ *Comm. ACM*, vol. 36, no. 4, pp. 83-99, April 1993.

Pearson1986a.

G. Pearson and M. Weiser, ‘‘Of Moles and Men: The Design of Foot Control for Workstations,’’ *Proc. ACM CHI’86 Human Factors in Computing Systems Conference*, pp. 333-339, 1986.

Schmandt1982a.

C. Schmandt and E.A. Hulteen, ‘‘The Intelligent Voice-Interactive Interface,’’ *Proc. ACM Human Factors in Computer Systems Conference*, pp. 363-366, 1982.

Schmandt1993a.

C. Schmandt, ‘‘From Desktop Audio to Mobile Access: Opportunities for Voice in Computing,’’ in *Advances in Human-Computer Interaction, Vol. 4,* ed. by H.R. Hartson and D. Hix, pp. 251-283, Ablex Publishing Co., Norwood, N.J., 1993.

Sears1991a.

A. Sears and B. Shneiderman, ‘‘High Precision Touchscreens: Design Strategies and Comparison with a Mouse,’’ *International Journal of Man-Machine Studies*, vol. 43, no. 4, pp. 593-613, April 1991.

Shneiderman1983a.

B. Shneiderman, ‘‘Direct Manipulation: A Step Beyond Programming Languages,’’ *IEEE Computer*, vol. 16, no. 8, pp. 57-69, 1983.

Shneiderman1992a.

B. Shneiderman, *Designing the User Interface: Strategies for Effective Human-Computer Interaction, Second Edition,* Addison-Wesley, Reading, Mass., 1992.

Stone1994a.

M.C. Stone, K. Fishkin, and E.A. Bier, ‘‘The Movable Filter as a User Interface Tool,’’ *Proc. ACM CHI’94 Human Factors in Computing Systems Conference*, pp. 306-312, Addison-Wesley/ACM Press, 1994.

Wickens1983a.

C. Wickens, A. Kramer, L. Vanasse, and E. Donchin, ‘‘Performance of Concurrent Tasks: A Psychophysiological Analysis of the Reciprocity of Information-Processing Resources,’’ *Science*, vol. 221, pp. 1080-1082, 1983.

FURTHER INFORMATION

Input is usually seen as part of human-computer interaction, so information about this area is typically found in general books, journals, or conferences on HCI, rather than in a specialized more venue.

Good introductions to these issues are found in the respective chapters of two standard textbooks in this area, by Shneiderman and by Foley, van Dam, Feiner, and Hughes.

Research in input devices and techniques is covered in the proceedings of the annual ACM CHI Human Factors in Computing Systems Conference, published by the Association for Computing Machinery (ACM, New York City). Other relevant annual conference proceedings include the ACM UIST Symposium on User Interface Software and Technology (ACM), Graphics Interface (Canadian Human-Computer Communications Society, Toronto, Canada), and the Human Factors and Ergonomics Society (HFES, Santa Monica, Calif.).

The journal *ACM Transactions on Computer-Human Interaction* (ACM) includes work on input devices and techniques, as do *Human Factors* (published by HFES) and *Human-Computer Interaction* (Lawrence Erlbaum Associates, Inc., Hillsdale, N.J.). *ACM Transactions on Graphics* (ACM) publishes a series of articles titled ‘‘Interaction Techniques Notebook,’’ which report newly-invented interaction techniques.

FIGURE CAPTIONS

Figure 1. The Infogrip BAT™ one-handed chord keyboard. This keyboard is used with the left hand. The user places his or her fingers over the four keys on the left and presses one of the three keys on the right with the thumb. By pressing combinations of keys, different numbers, letters, and other symbols can be generated. *Source:* Photo courtesy of Infogrip, Inc., Ventura, Calif.

Figure 2. The Polhemus 3SPACE™ FASTRAK™ 3-D magnetic tracker. The device reports the position and orientation in 3-D of each of the four sensors (the small white cubes in the foreground), using a magnetic signal sent from the transmitter (the larger black cube on the right). *Source:* Photo courtesy of Polhemus, Inc., Colchester, Vt.

Figure 3. The Spaceball SpaceController™ 3D control device. This device operates like a 3-D isometric joystick: the user holds the ball and pushes or twists it in the desired direction. *Source:* Photo courtesy of Spacetec IMC Corporation, Lowell, Mass.

Figure 4. CyberGlove™ 18-sensor instrumented gloves. The gloves report the configuration of the fingers of the user's hand, Note the 3-D magnetic sensor incorporated into the wristband of the glove; it reports the position and angle of the hand itself. *Source:* Photo courtesy of Virtual Technologies, Inc., Palo Alto, Calif.

Figure 5. Applied Science Laboratories helmet-mounted eye tracker. This device measures visual line of gaze or where the user's eye is pointing in space. A tiny camera, located above the user's forehead, views the eye through the half-silvered mirror. A second camera, located near the user's chin, is optionally used to keep a record of what the user saw for later analysis. *Source:* Photo courtesy of Applied Science Laboratories, Bedford, Mass.

Figure 1.

(Insert black and white photograph of computer with 7-button keyboard)

Figure 2.

(Insert black and white photograph of "Polhemus" tracker, with two coils of wire in front of it)

Figure 3.

(Insert 35mm slide of blue ball mounted on black stand)

Figure 4.

(Insert black and white photograph of gloved hand in front of computer screen)

Figure 5.

(Insert black and white photograph of man wearing helmet)