Not Just Pointing: Shannon's Information Theory as a General Tool for Performance Evaluation of Input Techniques

Yves Guiard, Julien Gori, Quentin Roy, Olivier Rioul

Not Just Pointing: Shannon's Information Theory as a General Tool for Performance Evaluation of Input Techniques

Yves Guiard[1,2] **Julien Gori**[2] **Quentin Roy**[3] **Olivier Rioul**[2]

[1]LRI, Université Paris-Sud, CNRS, Université Paris-Saclay F-91400, Orsay, France

[2]LTCI, Telecom ParisTech Université Paris-Saclay F-75013, Paris, France

[3]Singapore Management University Singapore 188065

{yves.guiard, julien.gori, olivier.rioul}@telecom-paristech.fr, quentin@quentinroy.fr

ABSTRACT

Input techniques serving, quite literally, to allow users to send information to the computer, the information theoretic approach seems tailor-made for their quantitative evaluation. Shannon's framework makes it straightforward to measure the performance of any technique as an effective information transmission rate, in bits/s. Apart from pointing, however, evaluators of input techniques have generally ignored Shannon, contenting themselves with less rigorous methods of speed and accuracy measurements borrowed from psychology. We plead for a serious consideration in HCI of Shannon's information theory as a tool for the evaluation of all sorts of input techniques. We start with a primer on Shannon's basic quantities and the theoretical entities of his communication model. We then discuss how the concepts should be applied to the input techniques evaluation problem. Finally we outline two concrete methodologies, one focused on the discrete timing and the other on the continuous time course of information gain by the computer.

Author Keywords

Shannon's information theory; communication; entropy; input techniques; interaction techniques ; computer input ; Fitts' law ; dynamics of information gain.

ACM Classification Keywords

H.5.2. Information interfaces and presentation: Evaluation/methodology.

INTRODUCTION

Human-computer interaction (HCI) overlaps with many fields of the human and social science, most notably psychology. Nonetheless, since its birth that discipline has never ceased to be firmly grounded in computer science.

This is the first version, submitted to the ACM CHI conference on September 19, 2017, of a currently in progress article.

Then arises the puzzle: Claude Shannon being unanimously recognized today as one of the founding fathers of information science and technology, how can it be that his mathematical theory of communication [44], known today as information theory, has had so far so limited an impact in HCI?

HCI researchers who run evaluation experiments are generally aware of three famous applications of information theory in the psychology of the 1950's, G. Miller's magical number in absolute judgment tasks [36], the Hick-Hyman law [20, 23], and Fitts' law [8, 24, 47]. The fact is, however, none of hem is prone to routinely leverage Shannon's concepts. Researchers concerned with the evaluation of input techniques typically talk task completion times and error rates, not entropy, mutual information or information transmission rates.

One obvious exception is pointing, where an application of Shannon's model by psychologist Paul Fitts (1954) [8] has proven very useful to HCI [4, 33, 47]. Pointing is important to HCI because it constitutes a basic building block for interaction with most of our commercially available computing terminals, including wall displays, ATMs, desktops, laptops, tablets, smartphones, down to miniature devices like smartwatches or digital jewels. This well-known application of Shannon's theory, however, is not quite as successful as commonly believed. For one thing, the link between Fitts' law and information theory is indirect and contrived. Not only is the concept of movement difficulty central to Fitts' law problematic [16] and quite foreign to information theory, the definition of the information quantity crucial to Fitts' law—the so-called index of difficulty—has always remained controversial [6, 16, 21, 41]. It should also be borne in mind that Fitts' law is just about pointing—and pointing, notwithstanding its overwhelming use in commercial interfaces, is but one special case in the vast design space of interaction techniques, as we will see.

The goal of this paper is to suggest that Shannon's information theory [44] has a considerable potential for HCI, of which our field does not seem to have taken the full measure. We will illustrate the conceptual and practical benefits to be expected from Shannon's theory in one specific subfield of HCI—the evaluation of the performance of interaction techniques.

THE COMPUTER-INPUT PROBLEM IN HCI
To represent the research problem specific to HCI it is usual to use the diagram of a ring, or loop (Figure 1a).[1] The computer and its user are connected by two arrows pointing in opposite directions, thus indicating that each entity is a source of information for the other. While sharply distinguishing the computer-input channel from the computer-output channel, this representation makes it obvious that the system is interactive, meaning that, relative to the other, each channel can be considered a feedback channel, and that much complexity is likely to ensue.

HCI theorists seldom venture into more detailed diagrammatic representations of their research problem and this is not too surprisingly. With finer grains of analysis the complexity of the diagram, even with a single computer and a single user, would soon become deterring. Not only are there typically several parallel channels behind each arrow of Figure 1a (e.g., the computer output may consist of visual and/or auditory and/or vibratory messages), but some of these channels have their specific computer-supported proprioceptive feedback loops (e.g., the computer keeps the user uninterruptedly informed visually of pointing device motion).

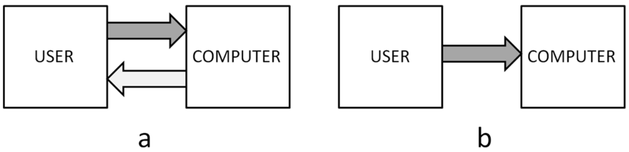

Figure 1. (a) : The general human-computer interaction problem; (b): the interaction-technique problem viewed as a computer-input problem.

An interaction technique can be prudently defined as a combination of hardware and software elements that provides users with a way to accomplish a certain class of tasks.[2] Such a definition is compatible with the field-defining, yet highly polysemous and hard to define notion of interaction [21]. Below, however, we will assume that in general what we call interaction techniques in HCI boil down in essence to computer-*input* techniques (Figure 1b). Of course not denying that interaction sequences constitute dialogues and that proprioceptive feedback loops are often involved, we will assume that the fundamental concern of interaction techniques evaluators is indeed the efficiency with which messages can be transmitted from the user (the source, in Shannon's terminology) to the computer (the destination).

There are various sorts of input events. In line with an established tradition of input techniques evaluation in HCI, the present paper mainly focuses on discrete input rather than more or less continuous flows of input information (e.g., text entry, drawing, or parameter setting). A command consisting of the choice of one discrete item among a countable finite set (the so-called menu), Shannon's discrete entropy applies most obviously. Surprisingly, however, so far input techniques evaluation has dispensed almost entirely with Shannon's tools.

USUAL METHODOLGY OF INPUT TECHNIQUES EVALUATION
The history of the particular field from which students of input techniques have borrowed their experimental methodologies, namely psychology, offers one explanation of Shannon's surprisingly limited impact on input techniques evaluation in HCI. In the early nineteen sixties, after ten years of considerable popularity [2, 39], the information-theoretic approach was abruptly abandoned by psychologists [32], who turned en masse to the then nascent cognitive approach [37]. Even Fitts' law—an empirical result no less important to psychologists than it is to HCI researchers—was soon reinterpreted in non information theoretic terms [5, 34].

State of the Art
Input techniques must be evaluated both subjectively and objectively. How a given technique is liked by users is an important question that HCI researchers often address, with more or less success,[3] using Likert scales. But here our main concern is objective performance. Since the beginnings of HCI [e.g., 4] the objective evaluation of input techniques has relied primarily on speed and accuracy measurements, in keeping with an old psychological tradition. In essence the method consists in evaluating the speed with which series of commands can be entered by a sample of participants, each being asked to use two or more techniques of interest to perform as fast as they can given the constraint that their error rate should not exceed some reasonable limit (typically on the order of 5%). The key dependent variable is the *response time*. In the context of input techniques evaluation this is the time elapsed from the onset of some stimulus—specifying which command the participant is asked to issue—to the end of the input act (e.g., a click, or the termination of the input gesture). The issue being the 'how', rather than the 'what' of command selection, for convenience most experimenters have recourse to sets of mockup menus composed of familiar concepts, such as animal names or color names.

Limitations of the Usual Methodology

[1] Our focus in this paper is on the simplest possible case, where one user interacts with one local terminal. More complex cases, including computer-supported collaborative work [1] will be left aside.

[2] This in essence is the Wikipedia definition of an interaction or input technique (September 8, 2017), inspired by that proposed by Foley et al. [9].

[3] as emphasized in [42], Likert scales are often misused in HCI, investigators tending to overlook their ordinal, non-metrical character.

The speed/accuracy method of performance evaluation we have just sketched rests on a well established tradition that traces back to the nineteenth century, and apparently it is as workable in HCI as it has always been in psychology. However, before introducing the possibility of another approach based on Shannon's information theory, it is useful to recall that the method of psychologists is certainly not flawless.

The most serious weakness of the classic methodology of performance evaluation arises from its incomplete control of the speed/accuracy trade-off. An old assumption in psychology, also widespread in HCI laboratories, is that if error rates are close to some reasonable minimum, then they can be ignored altogether and the performances can be characterize by just their speed [e.g., 48]. Unfortunately, however, this rests on wishful thinking as the shape of the trade-off function is known to be such that a tiny change in error rate (say, from 2% to 3%) is likely to entail substantial speed changes—simply because the slope of the trade-off is steepest in the high-accuracy region [17, 18, 31, 38, 50]. One correct solution is to compare trade-off functions, rather than response times and/or error rates, but this is a costly option and experimenters have persistently ignored it [31). Thus in general the usual methodology of performance measurement seems to lack an accurate and principled way of combining the speed and accuracy dimensions of performances, exposing researchers to the risk that speed and accuracy data may deliver contradictory conclusions.[4]

Another noteworthy limitation is that the familiar error rate is a rather coarse measure of accuracy. Insensitive to possible patterns, error rates convey no information on the sorts of errors users are prone to make [2].

The next two problems, which specifically concern HCI applications of classic methods, do not seem to have attracted much attention so far. One is that students of input techniques often fail in their experiments to safely decompose into successive non-overlapping stages the processes involved in computer input. In a typical experiment aimed at testing input techniques the trial starts with some experimenter-controlled stimulus serving to indicate to the participants what particular command they should issue. The duration of the response to that stimulus will be precious information on the efficiency of the technique under consideration; however, the latency of the response—i.e., the reaction time measured from stimulus onset to the start of input motion—is of little relevance, mostly reflecting characteristics of the stimulation procedure.[5]

But most HCI experimenters seem to lack a rationale for deciding whether or not their time measure should include the reaction time and as a matter of fact many authors find it necessary to report and analyze in parallel both the latencies and the durations of input motion. This exposes them to the extra risk of confusing contradictory findings in just the speed component of their performance analysis.

Finally, in the classic speed/accuracy methodology one tends to overlook the important dimension of the *size* of the available menu of commands. Consider for example smartwatches, often designed to offer a single binary choice on their touchscreen. Arguably such a design should allow users to make both fast and accurate selections, but is this the whole story? Since users generally need menus with more than two possibilities [25], such parsimony in menu width will have to be offset by an increase of menu depth. While we certainly want our input techniques to allow fast and accurate selections, we also want them to provide users with large enough vocabularies of commands.

SHANNON'S INFORMATION QUANTITIES

Information and Uncertainty: Shannon Entropy

Shannon's [44] stroke of genius was to provide a completely general model of the link between information and uncertainty. He designed a probabilistic measure of uncertainty which he called entropy. In the discrete case of relevance here the formula of entropy is

$$H(X) = \sum_{i=1}^{n} p(x_i) \log_2 \frac{1}{p(x_i)} \tag{1}$$

In our application context, X will denote the command, a chance event that takes values in the set $\{1, 2, \dots, n\}$, the integer n denoting the size of the vocabulary of available commands.

What Equation 1 shows is a weighted average, meaning the measure is determined not just by the number n of choices available in the menu, but also by the probability distribution of the possible choices. H is a bilaterally-bounded quantity, being non-negative and constrained by an upper bound:

$$0 \le H(X) \le \log_2 n \tag{2}$$

Twenty years before Shannon, Hartley [19] had proposed to use the log function to measure information. In fact $\log_2 n$ corresponds to the theoretical maximum of uncertainty obtainable with n possibilities—the maximum reached when all possibilities are equiprobable (uniform distribution). Shannon entropy is thus a generalized

[4] In this regard Fitts' law is again an exception. Based on information-theoretic considerations [47], Fitts' law experimenters are invited by the ISO standard [24] to compress their speed and accuracy data into a single throughput score.

[5] In the real world users receive no stimuli from experimenters. The input process that we want to understand starts at the initiation and ends at the cessation of input motion.

measure of uncertainty, which incorporates $\log_2 n$ as its upper limit. This point is illustrated in Figure 2, which considers a menu made up of 10 possible commands.

The central panel of Figure 2 shows an arbitrary distribution of probabilities, yielding H = 2.66 bits. It should be emphasized that knowledge of the probabilities of each possible command of a menu is a prerequisite to the calculation of the real—i.e., statistical—entropy of that menu. These probabilities, however, can only be determined empirically from frequency estimates, and so at some point we will have to recognize that a thorough information-theoretic tackle of input technique performance actually calls for field studies.

The lower bound H = 0 (lower panel of Figure 3) corresponds to the case where only one choice is possible, with probability p = 1. If this mathematically important limit, which mistreats the very idea of a menu, is of little practical interest, by contrast the maximum possible value of H given by $\log_2 n$ (upper panel) is very interesting indeed in the context of input research: this upper bound specifies a potential of entropy, and it makes a lot of sense to consider such a potential in the comparison of input techniques [30, 43]. The only thing to bear in mind is that that maximum is reached with a uniform distribution—a most implausible hypothesis when it comes to real-world menus, know to typically exhibit quasi-Zipfian distributions of probabilities [29].

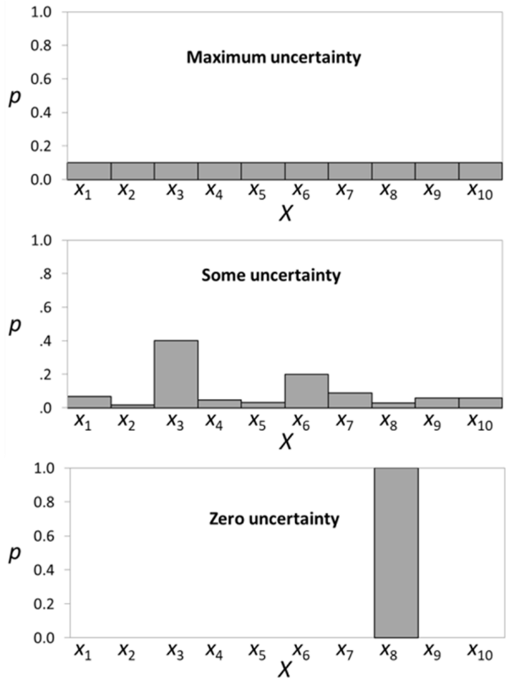

Figure 2. Illustration, with n = 10 possible commands, of the range of variation of Shannon's entropy H from $\log_2 10$ = 3.32 bits, the maximum obtained if all p_i = 1/10, down to 0 bit, the minimum obtained if p = 1 for one command and p = 0 for all others. With the arbitrary distribution shown in the middle panel, we obtain H = 2.66 bits.

Conditional Entropy and Transmitted information

To formulate the communication problem we need to introduce a second random variable Y to represent the channel output, which also takes values in the set $\{1, 2, \ldots, n\}$. From now on X will stand for the command sent by the user (U), and Y will stand for the command received by the computer (C).

Imagine an infallible input technique allowing perfectly noiseless (i.e., deterministic) communication between U and C. Given the command received by C with such a technique one would know with absolute certainty which command was sent by U.[6] Put differently, the conditional entropy $H(X|Y)$, defined as the conditional uncertainty about X given Y, would be zero. Simply subtracting that conditional, or remaining entropy from the initial entropy gives us the amount of transmitted (or mutual) information:

$$I(X;Y) = H(X) - H(X|Y) \qquad (3)$$

With an infallible technique allowing perfect transmission we would have $H(X \mid Y) = 0$ (i.e., no uncertainty whatsoever after reception of the command about which command was sent) and so the amount of transmitted information would equal the initial entropy of X. Now consider the opposite scenario of a totally unreliable input technique, which would fail to establish any degree of probabilistic dependency between X and Y. In such a case we would have $H(X \mid Y) = H(X)$, meaning that C would be as uncertain about which command was sent upon reception of the command as it was before. We can see from Equation 2 that this technique would allow on average the transmission of 0 bit of information per command from U to C.

Obviously the information-transmission performances of any real-world input technique will fall somewhere between the above two extremes. Where precisely the performance falls is easy to determine using Equations 1 and 3, whose practical utilization is explained in Figure 3.

For simplicity Figure 3 considers a menu composed of just two commands, called a and b, allowing four possible transmission outcomes, aa, ab, ba, and bb. It is supposed that a total of 1000 command transmissions have been logged, and that the four possible outcomes have been observed with the frequencies (counts) shown in the upper left table. Probabilities are estimated by dividing the counts by 1000, uncertainties are then obtained by multiplying the probabilities by the $\log_2$ of their reciprocals, and finally the sum of these uncertainties gives the entropy.

[6] Not necessarily the same command, however, because one could imagine a badly perverse design based on some arbitrary (though rigid) $X \rightarrow Y$ mapping (e.g., if x = COPY, then y =DELETE, etc.), thus opening the possibility of a perfectly noiseless transmission with a 100% error rate!

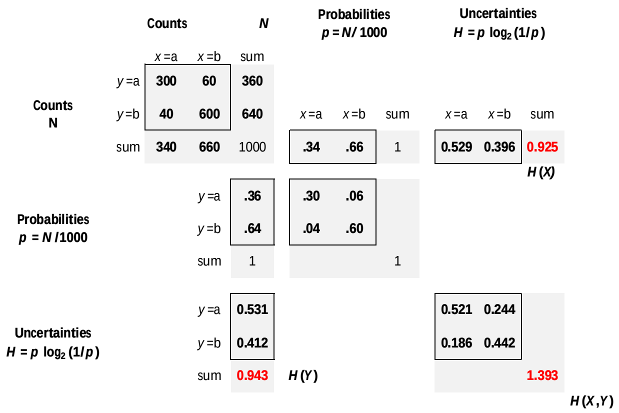

Figure 3. How to compute input entropy *H*(*X*), output entropy *H*(*Y*), and joint entropy *H*(*X*, *Y*) (arbitrary data).

In fact the calculation must progress in three directions. From left to right (i.e., ignoring the *Y* dimension) one obtains the input entropy *H*(*X*). From top to bottom (i.e., ignoring the *X* dimension), one obtains the output entropy *H*(*Y*). And along the diagonal (i.e., considering both dimensions), one obtains the joint entropy *H*(*X*, *Y*). Since it can be shown [] that $H(X \mid Y) = H(X, Y) - H(Y)$, Equation 3 can be rewritten as $I(X;Y) = H(X)+H(Y)-H(X, Y)$, and so we have our net measure of transmission: on average the imaginary technique of Figure 3 has allowed 0.925 + 0.943 – 1.393 = 0.47 bit of transmitted information per command.

In our example of Figure 3 the probability distribution of *X* is not uniform, as we let P(x = a) = .34 and P(x = b) = .66. Knowledge of the probability distribution of all the elements of the menu is needed to determine *H*(*X*). To dispense experimenters with the task of estimating, by means of costly field studies, realistic distributions, it is usual practice in HCI to confront experimental participants with equiprobable *X* alternatives. The simplification is also common in information-theory driven experimentation [30, 36, 43].

Additivity of Entropy Measures

Shannon's entropy of Equation 1 is a measure in the strict sense of measurement theory [35]. Entropy has a unit, the bit (also called the Shannon), and it is strictly positive. Entropy, pretty much like variance [12], can be decomposed additively, as is particularly easy to see in the useful representation, shown in Figure 4, due to Quastler [39].

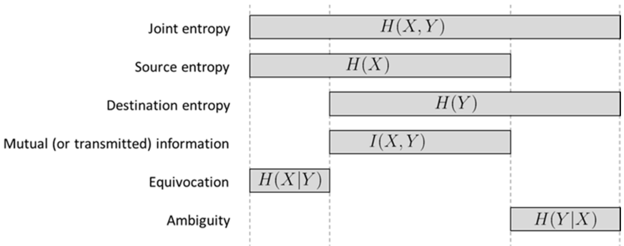

Figure 4. Additive decomposition of Shannon's entropy.

For example Equation 3, which defines the amount of transmitted information, can be recovered from just the visual inspection of this figure.

Measuring Transmission Times and Transmission Rates

As already explained, the evaluation of the speed of an input technique should in general rest on the measurement of the time *T* elapsed from the start to the end of the input motion. Any reaction time being excluded, what remains is what we usually call a movement time. This being recalled, that estimate of performance speed is all we need in the information-theoretic approach. Having measured, in the place of the conventional error rate, an amount of transmitted information (Equation 3), the calculation of a transmission rate *R*, in bits/s, just requires that the amount of transmitted information be divided by the time it takes to transmit that much information:

$$R = \frac{I(X;Y)}{T} \tag{4}$$

The rate of information transmission defined in Equation 4 may be called a throughput. It has the same dimension (bit/s) as the throughput of the ISO standard for the evaluation of pointing devices [24], and it is similar in content. There are two differences, however. One is that the derivation of Equation 4 being straightforward, our new definition is totally immune to the perplexities associated with the calculation of an index of difficulty. Second, Equation 4 holds for input technique in general, whereas the throughput of the ISO standard holds for just pointing.

Accuracy as a Relative Amount of Transmitted Information

Using Equation 3 it is straightforward to derive a *relative* measure of transmitted information. Reminiscent of the familiar error rate, the measure takes the form of a dimensionless percentage and can be considered a global index of accuracy (*A*):

$$A(X;Y) = \frac{I(X;Y)}{H(X)} \tag{5}$$

Such a measure is not redundant with error rate, however, because it tells us something about the structure of error transmission patterns to which error rates are totally blind [2, 13], meaning that it may certainly serve as a useful complement.

Technique Performance as a Function rather than a Data Point: The Optimax Plot

Equation 4 shows how speed and accuracy information can be elegantly compressed, with no loss, into a single performance measure. However integrative that measure, note that it may be insufficient to fully characterize an input technique, as a simple comparison with mechanical engineering may help to grasp.

Engineers asked to specify the power of an engine are likely to produce a curve rather than a numerical value. This is simply because the power of engines is strongly dependent on their regime, a typical power vs. regime function being concave with a global maximum (in watts) occurring at a certain optimal regime (in RPM). The situation is similar in the performance evaluation of input techniques, there being every reason to expect *R* to vary substantially with the entropy of the vocabulary of commands—a hypothesis discussed theoretically and verified experimentally by Roy et al. [43] and by Liu et al. [30]. To summarize their argument, as vocabulary entropy is increased transmitted information, the numerator of Equation 4, should level off at some point due to the user's capacity limitation [36]; on the other hand, there should be no levelling-off of transmission time, the denominator of Equation 4. Therefore the quotient *R* should exhibit a global maximum at some optimal value of vocabulary entropy. The data of both studies strongly supported that prediction.

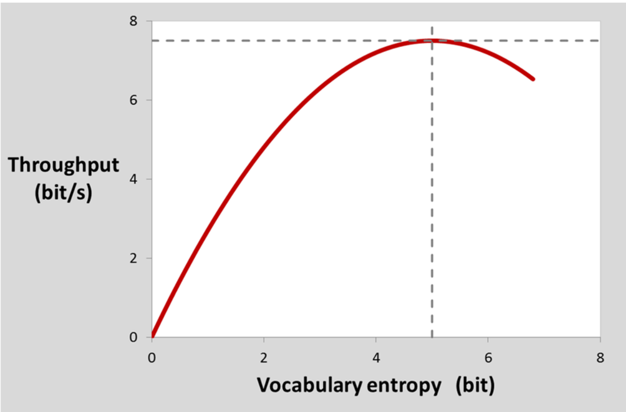

Figure 5. The optimax plot.

The plot of *R* (bits/s) vs. $H(X)$, which we call an *optimax* plot (Figure 5), delivers two pieces of empirical information both of relevance to input techniques research: an estimate of the vocabulary size that is optimal to a given technique and an estimate of the maximum rate of information transmission accessible with that technique.

MAPPING THE COMPONENTS OF SHANNON'S COMMUNICATION SCHEME TO THE INPUT PROBLEM

In this section we present preliminary comments on the way the different components of Shannon's general model should be mapped onto those of our specific HCI problem. The question is how our commands are initially coded into appropriate signals, how these signal are transmitted through some channel and ultimately decoded on the computer's end.

Shannon's theoretical entities are identified in his famous diagram (Figure 6), whose status in information science is reminiscent of the status of the structure of the atom in physics. It describes the structure of the irreducible building block of any conceivable information network. No matter the topology and the complexity of the network, every single link is a full-fledged Shannon communication system, with all its components—as one could check, had one the possibility to zoom in semantically, so to speak [11,15], on any element of any network diagram.

There is flexibility in the applicability of Shannon's model to real-world systems, the model applying at different levels of granularity. For example any two nodes of an information network, whether a simple local area network or the whole Internet, can be a source and a destination: no matter the positions of nodes A and B in the network, the network as a whole can be considered a Shannon channel linking A and B.

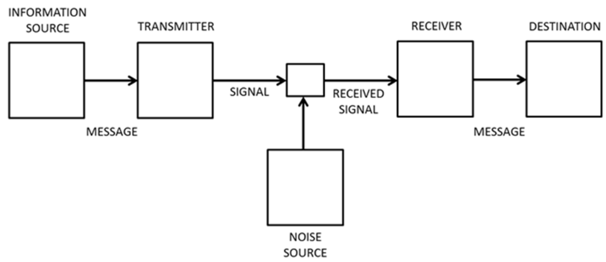

Figure 6. Shannon's diagram of a general communication system ([44], p. 380).

What we want to understand here is how Shannon's model can help us understand the structure of our input techniques evaluation problem. One detailed proposition is made in Figure 7, where every box and every arrow of Shannon's model is given a meaning of specific relevance to HCI.

Two important characteristics of Shannon's model must be emphasized. It is symmetrical with respect to the central channel box—i.e., all that must be done prior to signal transmission through the channel will have to be undone on the other side of the channel. Second, the diagram can be tackled at two different levels, marked graphically in Figure 7 by a vertical offset, the message level and the signal level.

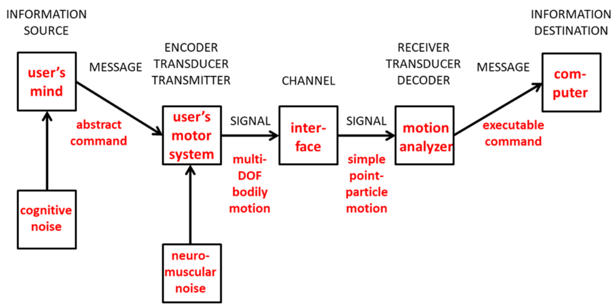

Figure 7. A tentative mapping of Shannon's communication scheme to the input techniques problem. Shannon's general concepts are recalled above the diagram, in uppercase characters.

The Input Interface is the Channel

The first explicit suggestion offered in Figure 7 is that Shannon's *channel* corresponds to the *input interface***.** Input techniques exploit a bewildering diversity of input devices, many of which require U to get physically in contact with some component of C (e.g., keyboards, mice, touchpads, touchscreens), while others involve remote sensing (e.g., microphones, hand or eye motion tracking). We have the further complication that sometimes C is made to sense its own motion (e.g., accelerometers onboard a hand-held device), in which case it is hard to interpret the interface as a region of 2D or 3D space. The general concept of an interface, also of great import to ergonomics and physics, is a tricky conceptual challenge faced by HCI theory. Apparently we must content ourselves with the prudent view that the input interface is the shared boundary between U and C [e.g., 9]. But the entity in question, however difficult to grasp, can be assigned a precise location in Shannon's chain.

Signals through the Interface: Point Particle Motion

One striking invariant of input techniques is that the *signal* circulating through the interface channel almost invariably takes the form of *point-particle motion*.[7] A point particle, an old abstraction of mechanics, is a body with no spatial extension, hence no orientation, whose motion can therefore be described using just its *positional* degrees of freedom (DOF). A clear illustration of the HCI exploitation of that abstraction is the familiar screen cursor, a graphical object of constant orientation whose location can be characterized by just the two spatial coordinates it precisely points to. The slider is a strict 1D equivalent that specifies just one coordinate. And input techniques that rely on 3D motion consider point-particle motion just as well, the role of the point particle being devoted for example to the whole hand (using, say, the Kinect) or, using more sophisticated motion-tracking technologies, to some optically- or magnetically-tracked marker. Whenever the pointer does have spatial extension (e.g., the 2D surface area occupied by the skin/glass contact on a touchscreen, or the volume occupied by the hand or a marker in 3D space), a barycenter is computed.

The observation that the point-particle model of physicists stands at the core of virtually all input techniques in HCI gives rise to the interesting possibility of a taxonomy of techniques based on elementary concepts of physical mechanics, as we will suggest shortly.

From Multi-DOF Movement to Simple-Particle Motion

Shannon's signal is the output of the transmitter, which is also a transducer (e.g., converting an acoustic vibration into an electrical current in telephony) as well as an encoder of the message to be sent. For that signal to take the form of an elementary motion event, a transformation must have taken place earlier in the chain—here we need to ask about the constraints imposed on our communication problem by the human nature of the information source. Recall that the bodies of macroscopic animals are highly complex machineries whose motion involves a huge number of highly-redundant musculoskeletal DOFs.[8] The multi-joint bodily *movements* of the user are far too complex to serve as input to the interface. Multi-DOF biological movement must be drastically reduced to 1-DOF simple-particle motion, a task automatically carried out by input devices such as mice and touchpads.

Noise Impact

The impact of noise is certainly not the same in our input techniques problem as it is in Shannon's [44] model. Primary concerned with remote communication such as telegraphy, telephony, or TV broadcasting, Shannon assumed all components of the communication chain other than the channel proper (i.e., the medium over which the signal circulates) to work deterministically. Unlike Shannon's remote communication channel, however, the input interface of HCI is typically a contact communication channel, and we may reasonably consider it deterministic. Stochastic errors, however, do affect input techniques and so the random variability issue needs to be addressed. It seem fairly obvious that input errors are caused by random variability occurring in the early components of the chain, which involve the human—the first two boxes of Figure 7. Somewhat ironically, psychological knowledge seems key to progress toward a rigorous application to input techniques of Shannon's detailed engineering model.

Based on abundant converging evidence, psychologists distinguish two markedly different kinds of errors, abstract *selection* errors and concrete *motor execution* errors.[9] Mistaking one command for another (an abstract, mental error) is clearly not the same as failing to shape properly the input act corresponding to an intended command (an error of the concrete, motoric kind). This line of argumentation suggests the view that input techniques suffer from two sources of random noise, one impacting the information source—i.e., the user's mind, from which we must suppose the commands to originate—and the other affecting the transmitter—the user's motor system.

The Input Technique Is the Code

As already noted, from the information-theoretic viewpoint the interface can be interpreted as the channel. Let us now propose the idea that, taken as a whole, the input technique is the code. In Shannon's communication framework the code is a mapping rule between sets of messages and sets of

[7] One obvious exception is brain-computer interaction [28], where C is asked to process many parallel EEG signals, with the help of AI algorithms.

[8] In human movement science and robotics this theme is generally known as the Bernstein DOF problem [27]. Note that in this paper we do not use the term *movement* interchangeably with the term *motion* borrowed from mechanics. Movement refer to the complex patterns of motion produced by living things.

[9] This dichotomy is classic for example in stimulus-response compatibility studies [14], which analyze reaction time as a series of information-processing stages [48]. There has been a stable consensus on the view that an abstract response selection stage precedes a motor preparation stage.

signals, and it serves twice: first at the encoding stage, where messages are translated into transmissible signals, and again at the decoding stage, where signals are translated back into messages comprehensible to the destination. In essence to design an input technique is to design a mapping rule thanks to which the user will be able to express every possible command of a menu in the coded form of a unique episode of motion.

This seems true in the case of techniques based on hand movement, but even the rather special example of vocal input is no exception. The user's vocal tract, a multi-DOF motor system, takes charge of the translation of the command intention (the source message) into waves of acoustic pressure. However complex the phonetic structure of human utterances, notice that it is again simple point-particle motion (more specifically, 1-DOF vibratory motion) that is being sensed by the microphone, the first component of the input interface, before being transduced into an electrical current and sent through the channel.

A Simple Taxonomy of Motion Coding

Point-particle motion being analyzable in many ways, a whole diversity of coding schemes can be contemplated.

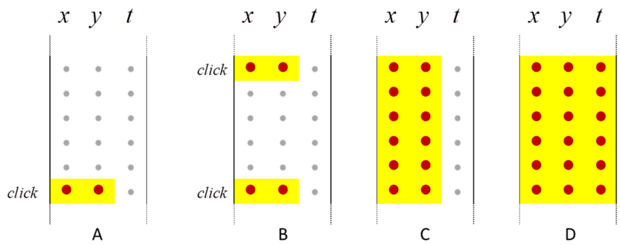

Figure 8. Different input coding possibilities available with input consisting of point-particle motion. Each dot stands for one number specifying a space or time coordinate. Highlighted is the data subset serving in input coding.

Figure 8 describes a simple taxonomy of input techniques. Given a logged series of triple $\{x, y, t\}$ coordinates describing the motion of the relevant point particle in some 2D layout, the code may be based on

- isolated spatial *locations* of the particle, as is the case in the vast diversity of pointing-based techniques,
- spatial *paths* formed by series of spatial coordinates of the particle, as in most gesture-based input techniques (e.g., see [51]),
- or, less frequently, space-time, or *kinematic trajectories* (e.g., as in [3, 7) followed by the particle.

The coding principle is simplest in the case of pointing with a screen cursor or, equivalently, tapping with a finger tip (Figure 8A). All menu items being displayed at non-overlapping locations in 2D space, the receiving algorithm infers the user's decisions just from the spatial coordinates $\{x, y\}$ of the *endpoint* of any episode of motion, identified by a discrete event like a click or a tap.

Notice that the coding scheme of Figure 8A, at work in the vast majority of our commercial interfaces, leaves unexploited most of the spatial and temporal information actually available in the motion produced by the user. Hence the interest of the case shown in Figure 8B, a variation on the pointing theme in which *two* pairs of $\{x, y\}$ coordinates are considered. For example MarkPad [10] codes command shortcuts based on the locations of both the start point and the endpoint of linear segments traced on a touchpad with the finger. Using this technique, which extracts twice more information from the logged motion data than does usual pointing, one could code hundreds of shortcut alternatives using the limited real estate of a standard touchpad.

In the case of Figure 8C the input code leverages the shape of *paths*, meaning that the decoding algorithm now considers series, rather than isolated pairs, of spatial coordinates. In this approach to the input coding problem users are invited to produce *stroke gestures*, an option which opens the way to a huge design space [51].

Finally, why not exploit the totality of the kinematic, or space-time information available in the logged data (Figure 8D), taking into consideration whole series of $\{x, y, t\}$ triplets? The crucial difference between (spatial) path analysis and kinematic trajectory analysis is that only the latter considers such quantities as velocity and acceleration. Examples of techniques exploiting this possibility are Flick-and-Brake [3] and Motion pointing [7] (for a recent review, see [49]).

POSSIBILITY OF AN INSTANT DYNAMICAL APPROACH TO INPUT TRANSMISSION

In both the classic cognitive approach and the optimax approach sketched above, time is construed as a sequence of non-overlapping intervals separated by discrete overt events. Many experimental studies of input techniques performance record a reaction time, measured from stimulus onset to response initiation, and a movement time measured from motion start to motion end. To this global chronometric approach we may contrast an instant dynamical approach, where time is considered continuous. Let us ask about the time course of information acquisition by the computer during input motion.

To tackle this issue, obviously $H(X|Y)$ must be computed, requiring that each input gesture be logged many times. We also need a decoding algorithm for analyzing the input uninterruptedly from the start to the end of input motion. Suppose we have the data we need, consisting of many logs of input motion gathered for each item of the vocabulary of commands. The beginning of motion serving as the origin of our time axis, for each input gesture we can compute conditional entropy at each time stamp, every 10 milliseconds or so, and thus obtain a quasi-continuous time profile of information gain by the computer.

To take an obvious example, how will such a time profile differ between a technique based on the Cartesian coordinates of a click or tap, as in regular pointing (Figure 8), and one based on angular selection, as in marking menus [26]? There is little risk to say that with pointing all the information is delivered in an all-or-none fashion, and very late indeed, whereas with the marking menu it is delivered gradually, and pretty early. For pointing it will be observed that the rate of information transmission is zero during the totality of input motion before explosion at the click, and that in the case of angular selection that rate is particularly high at the beginning of motion—a strong argument for the angular selection principle.

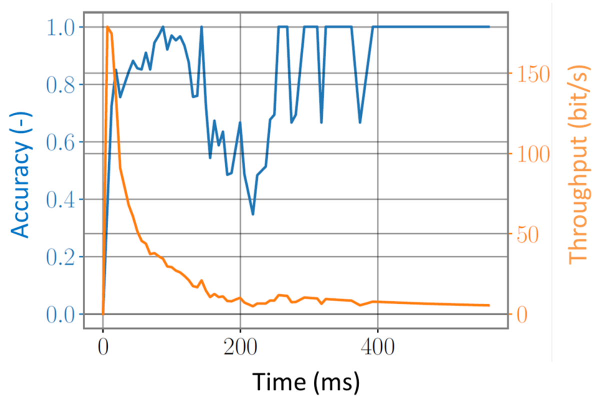

Figure 9. Empirical illustration of the time course of information transmission in the case of angular input.

Preliminary evidence about the efficacy of angular selection is shown in Figure 9, which plots the results of an exploratory experiment, run with one participants, in which we logged hundreds of simple linear mouse strokes oriented in eight different directions (the Compass8 case of [26]). Input decoding used the simplest imaginable memoryless real-time algorithm—namely, computing the arctangent from each sequence of two consecutive locations of the pointer. Two dependent variables are plotted in the figure. One is our relative measure of transmitted information (the accuracy of Equation 5), the other is the cumulated throughput—i.e., the current ratio of $I(X;Y)$ to the time elapsed at the time stamps plotted on the horizontal axis. The accuracy curve shows that a few tens of milliseconds suffice for the computer to be virtually certain about the identity of the command message. It is mostly at the beginning of the angular stroke that information is transmitted. This is confirmed by the instant throughput curve, which peaks sharply just after motion onset.

We tested on the same data a number of real-time decoding algorithms of various degrees of sophistication and found, quite interestingly, that the pattern remained essentially the same. This outcome supports the general idea that the bottleneck of information transmission in input techniques evaluation does not take place at the final stage of computer decoding, but rather at the early stage of message encoding by the user, as suggested above. If the input technique is a code, the efficiency of that code depends on its actual workability at the stage of human encoding.

DISCUSSION AND PERSPECTIVES

Perhaps the strongest argument for a serious consideration of the information-theoretic approach to the performance evaluation of input techniques is that it offers a principled and practically satisfactory solution to the trade-off problem, essentially intractable in the usual approach, namely by computing a throughput expressed in bits/s. The ISO standard does recommend such a computation, but only for pointing. In essence our argument is to generalize this recommendation to *all* sorts of input techniques, which is possible using the simple and straightforward definition of throughput given in Equation 4.

In this paper we have treated separately Shannon's quantities and the theoretical entities he identified with his famous diagram of Figure 6. It is interesting to note that if the psychologists of the 1950's [2, 12, 20, 23, 36, 39] used Shannon's information measures abundantly, they paid in general little attention to his diagram. Fitts [8] was no exception. Primarily interested in a measurement problem—how to express movement difficulty in bits—he remained vague on the functional location, in Shannon's transmission chain, of what he designated as the human motor system. Apparently the same can be said of most modern HCI research on Fitts law [21, 33, 47].

In this research we realized that the task of understanding how Shannon's structural concepts apply to the input problem is less trivial than may be thought, and that it is a prerequisite to any attempt to apply Shannon's ideas. One provisional conclusion reached in our attempt to apply Shannon's communication diagram to the input problem is that the input technique is the code. Such a conclusion is consistent with the idea that the fundamental stake of input techniques evaluation is the improvement of information transmission through a structure that interaction designers cannot change. The user's mind (the source) and the human motor system (the transmitter) are what they are, and the computer (the destination), even though its performances improve, is what it is. As for the decoding algorithms (the receiver), they almost always can be made to work to our satisfaction. The one component subject to revision and improvement is the channel (the interface), but its design is entirely dependent on the code being used.

Thus it seems that the problem faced by the designers and evaluators of input techniques ultimately boils down to a code optimization problem—obviously under constraints arising from both the human source (the mind) and the human encoder (the motor system). Such emphasis on the code—which, as we have seen, virtually always maps input messages onto characteristics of some sort of point particle motion—seems coherent with the fact that many practitioners of information theory take the code as their key concern, not to mention the fact that coding theory has become since Shannon's time an essentially autonomous research field, with numerous sub-specialties (e.g., [40]).

The paper called attention to the fact that so far most laboratory applications of Shannon's ideas in psychology and HCI have drastically simplified Shannon entropy by choosing, in practice, equiprobable alternatives, thus fixing the quantity at its maximum. Real-world menus being known to exhibit spectacularly non-uniform distributions of commands [29], there is reason to ask if recourse to more realistic non-uniform menus could not allow more faithful evaluations of input techniques. To evaluate that concern, among others, more research is needed.

REFERENCES

1. Abbott, K.R., Sarin, S.K. (1994). Experiences with workflow management: issues for the next generation. Proceedings of the 1994 ACM conference on Computer supported cooperative work, pp. 113–120. ACM.
2. Attneave, F. (1959). Applications of information theory to psychology: A summary of basic concepts, methods, and results. New York: Henry Holt and Company.
3. Baglioni, M., Malacria, S., Lecolinet, E., & Guiard, Y. (2011). Flick-and-brake: finger control over inertial/sustained scroll motion. In CHI'11 Extended Abstracts on Human Factors in Computing Systems, pp. 2281-2286. ACM.
4. Card, S. K., English, W. K., & Burr, B. J. (1978). Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys for text selection on a CRT. Ergonomics, 21(8), 601-613.
5. Crossman, E. R. F. W., & Goodeve, P. J. (1983). Feedback control of hand-movement and Fitts' law. The Quarterly Journal of Experimental Psychology, 35(2), 251-278.ve
6. H. Drewes (2010). "Only one Fitts' law formula please!" in CHI'2010 Extended Abstracts, pp. 2813-2822. ACM.
7. Fekete, J. D., Elmqvist, N., & Guiard, Y. (2009). Motion-pointing: target selection using elliptical motions. In Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 289-298. ACM.
8. Fitts, P.M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. Journal of experimental psychology, 47(6), 381.
9. J.D. Foley, A. van Dam, S.K. Feiner and J.F. Hughes (1990), Computer Graphics: Principles and Practice, Addison–Wesley.
10. Fruchard, B., Lecolinet, E., & Chapuis, O. (2017). MarkPad: Augmenting Touchpads for Command Selection. In Proceedings of the CHI 2017 Conference on Human Factors in Computing Systems, pp. 5630-5642. ACM.
11. Furnas, G.W., Bederson, B.B. (1995). Space-scale diagrams: Understanding multiscale interfaces. In Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 234-241. ACM.
12. Garner, W.R., & McGill, W.J. (1956). The relation between information and variance analyses. Psychometrika, 21(3), 219-228.
13. Gori, J., Rioul, O., & Guiard, Y. (2017). To Miss is Human: Information-Theoretic Rationale for Target Misses in Fitts' Law. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (pp. 260-264). ACM.
14. Guiard, Y. (1983). The lateral coding of rotations. Journal of Motor Behavior 15(4), 331-342.
15. Guiard, Y., & Beaudouin-Lafon, M. (2004). Target acquisition in multiscale electronic worlds. International Journal of Human-Computer Studies, 61(6), 875-905.
16. Guiard, Y., & Olafsdottir, H. B. (2011). On the measurement of movement difficulty in the standard approach to Fitts' law. PLoS one, 6(10), e24389.
17. Guiard, Y., Olafsdottir, H.B., & Perrault, S. (2011). Fitts' law as an explicit time/error trade‒off. In Proceedings of CHI 2011, ACM Conference on Human Factors in Computing Systems, pp. 1619-28. New York: Sheridan Press.
18. Guiard, Y., & Rioul, O. (2015). A mathematical description of the speed/accuracy trade-off of aimed movement. In Proceedings of the 2015 British HCI Conference, pp. 91-100. ACM.
19. Hartley, R.V. (1928). Transmission of information. Bell Labs Technical Journal, 7(3), 535-563.
20. Hick, W.E. (1952). On the rate of gain of information. Quarterly Journal of Experimental Psychology, 4(1), 11-26.
21. Hoffmann, E.R. (2013). Which version/variation of Fitts' law? A critique of information-theory models. Journal of motor behavior 45, 3 (2013), 205–215.
22. Hornbæk, K., & Oulasvirta, A. (2017). What Is Interaction? In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (pp. 5040-5052). ACM.
23. Hyman, R. (1953). Stimulus information as a determinant of reaction time. Journal of experimental psychology 45, 3, 188-196.
24. ISO. (2002). Reference Number: ISO 9241-9:2000(E):, Ergonomic requirements for office work with visual

display terminals (VDTs) - Part 9 - Requirements for non-keyboard input devices (ISO 9241-9) (Vol. February 15, 2002): International Organisation for Standardisation.

25. Kiger, J.I. (1984). The depth/breadth trade-off in the design of menu-driven user interfaces. International journal of man-machine studies, 20(2), 201-213.
26. Kurtenbach, G., & Buxton, W. (1994). User learning and performance with marking menus. In Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 258-264. ACM.
27. Latash, M. L. (1996). The Bernstein problem: how does the central nervous system make its choices. Dexterity and its development, 277-303.
28. Lebedev, M. A., & Nicolelis, M. A. (2006). Brain–machine interfaces: past, present and future. Trends in Neurosciences, 29(9), 536-546.
29. Liu, W., Bailly, G., & Howes, A. (2017). Effects of Frequency Distribution on Linear Menu Performance. In Proceedings of CHI'17, ACM.
30. Liu, W. Rioul, O., Beaudouin-Lafon, M., Guiard, Y. (2017). Information-Theoretic Analysis of Human Performance for Command Selection. In Proceedings of INTERACT'17, Mumbai, India.
31. Luce, R.D. (1986). Response time distributions in memory search: A caution. In F. Clix & H. Hagendorf (Eds.), Human Memory and Cognitive Capabilities – Mechanisms and Performances. Amsterdam: Elsevier Science Publishers (North Holland).
32. Luce, R.D. (2003). Whatever happened to information theory in psychology? Review of General Psychology, 7(2), 183-188.
33. MacKenzie, I. S. (1992). Fitts' law as a research and design tool in human-computer interaction. Human-Computer Interaction, 7, 91-139.
34. Meyer, D. E., Keith-Smith, J. E., Kornblum, S., Abrams, R. A., Wright, CE (1990). Speed-accuracy tradeoffs in aimed movements: Toward a theory of rapid voluntary action, in Jeannerod, M., Attention and Performance. Lawrence Erlbaum Associates, Inc, New Jersey.
35. Michell, J., & Ernst, C. (1996). The axioms of quantity and the theory of measurement, Part I. Journal of Mathematical Psychology, 40, 235-252.
36. Miller, G.A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. Psychological review, 63, 2, 81.
37. Neisser, U. (1967). Cognitive psychology. New York, NY: Appleton- Century- Crofts.
38. Pachella, R. G. (1973). The interpretation of reaction time in information processing research (No. TR-45). http://www.dtic.mil/get-tr-doc/pdf?AD=AD0787880
39. Quastler, H. (1955). Information theory in psychology. Glencoe, IL: Free Press.
40. Rioul, O. (2007). Théorie de l'Information et du Codage. Paris: Hermes.
41. Rioul, O., & Guiard, Y. (2012). Power vs. logarithmic model of Fitts' law: A mathematical analysis. Mathematics and social sciences 199, 85-96.
42. Robertson, J. (2012). Likert-type scales, statistical methods, and effect sizes. Communications of the ACM, 55(5), 6-7.
43. Roy, Q., Guiard, Y., Bailly, G., Lecolinet, É., Rioul, O. (2015). Glass+Skin: An Empirical Evaluation of the Added Value of Finger Identification to Basic Single-Touch Interaction on Touch Screens. Interact 2015.
44. Shannon, C. (1948). A mathematical Theory of Communication. Bell System Technical Journal, 27, 379–423.
45. Shannon, C. E. (1956). The bandwagon. IRE Transactions on Information Theory, 2(1), 3.
46. R. William Soukoreff, Jian Zhao & Xiangshi Ren (2011). The Entropy of a Rapid Aimed Movement: Fitts' Index of Difficulty versus Shannon's Entropy. Proceedings of the 13th IFIP TC13 Conference on Human-Computer Interaction - INTERACT 2011, Part 4, LNCS 6949, pp. 222-239.
47. Soukoreff & MacKenzie (2004). Towards a standard for pointing device evaluation: Perspectives on 27 years of Fitts' law research in HCI. International Journal of Human-Computer Studies, 61, 751-789.
48. Sternberg, S. (1969). The discovery of processing stages: Extensions of Donders' method. Acta psychologica, 30, 276-315.
49. Velloso, E., Carter, M., Newn, J., Esteves, A., & Gellersen, H. (2017). Motion Correlation: Selecting Objects by Matching Their Movement. ACM Transactions on Computer-Human Interaction (ToCHI), 24(3).
50. Wickelgren, W. A. (1977). Speed-accuracy tradeoff and information processing dynamics. Acta psychologica, 41(1), 67-85.
51. Zhai, S., Kristensson, P.O., Appert, C., Andersen, T.H., Cao, X. (2012). Foundational Issues in Touch-Surface Stroke Gesture Design — An Integrative Review. Foundations and Trends in Human–Computer Interaction 5 (2), 97–205.