Benchmarking Pointing Techniques with Distractors: Adding a Density Factor to Fitts' Pointing Paradigm

Renaud Blanch, Michael Ortega

Benchmarking Pointing Techniques with Distractors: Adding a Density Factor to Fitts' Pointing Paradigm

Renaud Blanch[1], **Michaël Ortega**[2]

[1]UJF-Grenoble 1 / UPMF-Grenoble 2 / Grenoble-INP / [2]CNRS / INRIA
Laboratoire d'Informatique de Grenoble, UMR 5217
F-38041, Grenoble, France
blanch@imag.fr, ortega@imag.fr

ABSTRACT

Fitts' pointing paradigm is widely used to conduct controlled experiments and to evaluate new interaction techniques enhancing target acquisition. Many of them change the behavior of the cursor according to various inputs, most notably the positions of potential targets. We propose to extend Fitts' paradigm in order to challenge those techniques with *distractors* (i.e., potential targets which are not the goal of the user) in a controlled manner. To reduce variability, we add a single new factor to the paradigm, the *distractor density*. We specify a distractors distribution, fully determined by this factor together with those of Fitts' task, aimed at reducing bias toward a specific technique. We also propose a preliminary extension of Fitts' law to take account of the sensitivity to the density of distractors as well as of the task difficulty. In an experiment, we compare five existing pointing techniques, and show that this extended protocol enables contrasted comparisons between them.

Author Keywords

Distractors, Fitts' law, Fitts' paradigm, pointing.

ACM Classification Keywords

H.5.2 Information Interfaces and Presentation (e.g., HCI): User Interfaces – *Evaluation/methodology, Input devices and strategies, Theory and methods*.

General Terms

Experimentation, performance, standardization, theory.

INTRODUCTION

Selecting a target by pointing is one of the most frequent tasks performed in graphical user interfaces (GUIs). This task is well modeled in the physical world by Fitts' law [12], which also holds in virtual worlds [13]. Many interaction techniques have been proposed to facilitate target acquisition in GUIs: Balakrishnan [5] offers a survey of such techniques proposed up to 2004, and new ways to enhance pointing are still regularly discovered.

http://doi.acm.org/10.1145/1978942.1979180

Most of those enhancements rely on the system knowledge of potential target positions. This knowledge is exploited to alter the cursor behavior, either by changing its activation area (e.g., [20, 21, 28, 15]); by expanding potential targets(e.g., [30, 26]); by adapting the control-display ratio to make it "sticky" (e.g., [28, 7, 6]); or by adding "force fields" attracting the cursor (e.g., [2, 3]).

Those techniques are theoretically appealing but difficult to fully evaluate. The sole established experimental protocol shared by the HCI community is derived from Fitts' experimental paradigm. As depicted in Figure 1, it consists in the presentation of 1D pointing tasks characterized by two parameters (its amplitude (A) and its target size (W)). This experimental paradigm is adapted to conduct controlled experimental studies because it introduces only two independent variables. One limitation of such evaluations is obvious: they do not consider distractors. As reported by Ahlström *et al.* [3], in the absence of distractors, techniques such as *semantic pointing* [7] or *object pointing* [19] are tested at their optima.

The solution adopted by most researchers to evaluate the impact of distractors on their interaction technique is to conduct a less formalized experiment based on a more realistic interaction scenario. We will review some of those experiments in a later section. Let us highlight some limitations of this common practice:

- Since the design of experiments is geared towards mimicking the real world, too many independent variables are introduced and they can not be extensively explored. The analysis and the interpretation of the results are thus difficult to conduct.
- Such designs are often underspecified because they involve complex or random choices which are hard, if not impossible, to document using few words. This leads to experiments which are not reproducible.
- A corollary of the previous item is that a new experiment is often proposed for each new technique. The evaluations are then not comparable.
- The last shortcoming of such experiments is that they are designed specifically to evaluate the proposed technique and their design could be biased to favor the technique [4], either consciously or not.

Figure 1. Fitts' 1D pointing task. The task is characterized by its amplitude (A) and its target width (W).

Our main motivation in this paper is to propose a novel experimental paradigm to address the lack of an unified way to evaluate new pointing techniques. This experimental setup could then be used as a benchmark to compare those techniques in presence of distractors. We bootstrap the process by comparing various techniques from the literature.

We first review how pointing techniques have been evaluated regarding their sensitivity to distractors. We then present the proposed experimental paradigm and how it has been designed. We also propose an extension to Fitts' law that considers the influence of distractors on performance. In the last section, this paradigm is used as a benchmark to compare different pointing techniques from the litterature.

RELATED WORK

As mentioned in the introduction, new interaction techniques enhancing target acquisition have been proposed for a while. We focus only on a set of techniques that are representative of how distractors have been handled in the past.

First, it should be noted that some consequent efforts have already been invested to unify the evaluation methodology of pointing devices or techniques and to spread those best practices into the community (e.g., the ISO standard for evaluating pointing devices [11], the recommendations of Soukoreff and MacKenzie [27] and of Zhai [29]). We do not propose "yet another framework" to evaluate pointing techniques. We propose a way to challenge techniques with distractors in a controlled manner, and this extension to the Fitts' pointing task can then be easily integrated into existing evaluation frameworks that do not currently handle the issue of distractors. The results of the experiments conducted using this paradigm can of course be evaluated using established methodologies.

Some authors have simply ignored the problem of distractors and have stuck to the established Fitts' protocol to evaluate their techniques (e.g., *semantic pointing* [7]). In this case, the technique is then evaluated in the optimal configuration: a single target known by the system. In a first experiment, the *object pointing* technique [19] is also evaluated using Fitts' paradigm. Since the cursor jumps to the closest target, it leads to a quasi-constant pointing time (i.e., the time needed to press the button). In a second experiment, the technique is evaluated in a more realistic 2D setup including distractors. Its performance is then comparable with the performance of normal pointing. Unfortunately, the 2D setup is weakly described and is impossible to reproduce.

Some authors (e.g., Keyson [22] or Worden *et al.* [28]) have addressed the issue of distractors by stressing their techniques with a worst case scenario. This illustrates that having the technique and the experimental protocol designed by the same person can introduce a bias [4], either against or in favor of the technique. While being particularly fair, this methodology has a drawback: it tests the technique for a single configuration of distractors and thus does not provide insights on its sensitivity to the amount of distractors.

Finally, other authors tested their techniques with varying distributions of distractors. The setup used to evaluate *ninja cursor* [23] is based on a "pseudo-random distribution" of distractors, presumably uniform, across the screen. They tested their technique in the presence of 1, 100 and 400 targets. Chapuis *et al.* evaluated their *DynaSpot* technique [10] with a setup proposed by Grossman and Balakrishnan for the evaluation of their *bubble cursor* technique [15]. This layout is not totally documented but its idea is to control the density of the distractors while distributing them pseudo-randomly on the path between the start of the movement toward the target and around the target itself. In those papers, the sensitivity to distractors is evaluated using this density as a factor.

We also believe that the density of distractors is the relevant factor to consider when testing the sensitivity of interaction techniques to the presence of distractors. We back up this claim in the next section.

POINTING TASK WITH DISTRACTORS

Our goal is to provide a pointing task that will challenge pointing techniques with distractors in a less biased as possible manner, and in a way that allow reproducible experiments. We want to characterize the distractors distribution by as few independent variables as possible, in order to ease the statistical analysis of the results.

In this section, we first formalize why the distractor density is the factor that should be added to the task parameterization and to the analysis of experimental results. We then present the pointing task composed of a target and several distractors, give its parametrization, and provide its design rationale.

Density as a Significant Factor

As seen in the Related Work section, researchers are aware that the presence of distractors impacts pointing techniques [23, 10, 15]. When they try to study this impact, there is a consensus among them to use the distractor density as the relevant factor to analyze how the performance is impacted by the presence of distractors.

This can be justified in an *ad hoc* manner for some techniques. For the *bubble cursor* and *object pointing*, a density of 1 (i.e., no empty space between distractors) makes the techniques identical to a normal pointing. On the other hand, a density tending towards 0 (i.e., no distractor at all) makes the target selection almost instantaneous regardless of the difficulty of the task. For *semantic pointing*, a density of 1 means that the motor scale is modified uniformly along the path toward the target and inside the target itself, and so no improvement should be observed (the index of difficulty in motor and visual space are equal). On the other hand, a density of 0 should lead to the improvements reported by Blanch *et al.* [7]. A careful analysis of other techniques should lead to similar conclusions.

A more general justification of the role that the distractor density plays can be suggested by analyzing the pointing task from an information theory point of view. The "room for improvement" in pointing tasks has been previously analyzed (e.g., by Blanch *et al.* [7]) as the mismatch between the abstract selection task —e.g., selecting 1 icon among the 32 icons on the desktop— and the actual pointing task —pointing at one of the 32×32 pixels of that icon among the 1600×1200 pixels of the screen. The former involves providing $\log_2(32) = 5$ bits to the system whereas the later consists in providing $\log_2\left(\frac{1600\times1200}{32\times32}\right) \approx 11$ bits[1]. Finding a way for not providing those $(11-5=)$ 6 extra bits is roughly the goal of all recent pointing techniques.

The *density of distractors* (ρ) gives a measure of this mismatch: $\rho = 0$ means that only the target is present —the abstract task is then a 0 bit task and the mismatch is maximal; on the other hand, $\rho = 1$ means that the space is paved with targets —the abstract and actual tasks are reunited and there is no room left for improvement using only the knowledge of the potential target positions.

The *de facto* manner authors have expressed sensitivity to distractors, the quick analysis of particular techniques, and finally the thought experiment proposed above' lead to the same conclusion: the *distractor density* is a factor that should be used to test the efficiency of pointing techniques.

Task Parameterization

A pointing task consists in a target whose size and position make the task more or less difficult. To add distractors to this task in a reproducible manner, their layout has to be specified as a function of the target geometry and of the *distractor density*. We present our proposed distribution of the distractors, explain how it is parametrized, and document the choices made while designing it.

Target Parameterization

Fitts' experimental protocol parametrizes a pointing task with 2 independent variables. The *amplitude* (A) of the task and its *width* (W) (see Figure 1 for a 1D pointing task) have long been used as factors. However, performance in *movement time* (MT) is always analyzed as function of the *index of difficulty* (ID) using Fitts' law [12]:

$$MT = a + b \times ID, \tag{1}$$

with ID expressed using Shannon formulation as advocated by MacKenzie [24]):

$$ID = \log_2\left(\frac{A}{W} + 1\right). \tag{2}$$

[1] This is not the Fitts' index of difficulty of an actual pointing task, but an upper bound of it.

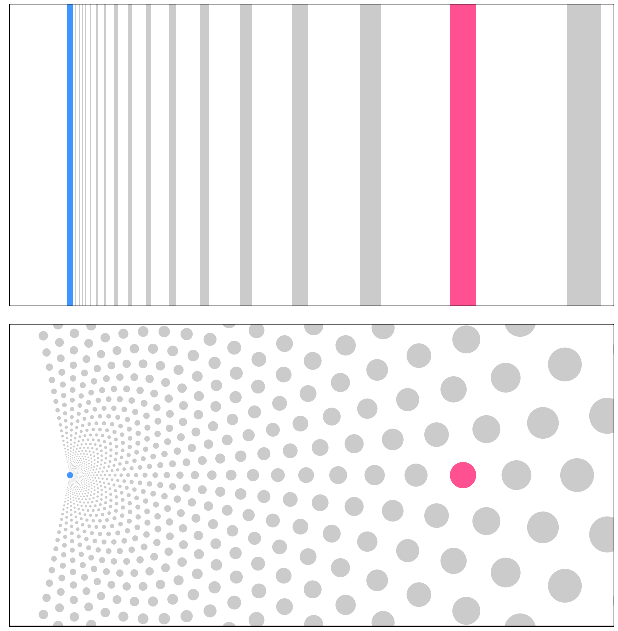

Figure 2. Self-similar tasks with distractors ($ID = 4, \rho = 0.25$). 1D (top) and 2D (bottom) tasks; distractors are grey, home is blue (left), target is red (right).

Guiard advocates for a parametrization of the task by its "form" and "scale", i.e., ID and A [17, 18]. This parametrization is formally equivalent to the use of A and W because ID, A and W are linked by Equation (2). Once ID and A are chosen, W can be computed by expressing W as a function of A and ID:

$$W = \frac{A}{2^{ID} - 1}. \tag{3}$$

As explained by Guiard, using ID and A as factors suppresses a bias widespread in the pointing experiments (namely, that high IDs are correlated with large As). We decided to use this parameterization of the target, but the distractor parameterization presented below still holds as soon as the target is specified by whichever means.

Distractors Parametrization

A question remains: given a *distractor density* (ρ), how to choose the layout of the distractors, i.e., their position and size? There are a lot of possible answers to this question. Adding as few new factors as possible is a strong requirement: the more factors are used, the more experimental trials will be needed to cover the cross product of their variation ranges. Our goal is to add the minimal number of factors that makes it possible to study the effect of distractors density. Since ρ should at least be added, we propose to stick with this minimal addition. The layout of the distractors should then be totally determined by the three factors: ID, A and ρ.

We propose a 1D layout, and since Fitts' law 2D generalization [25, 1, 16] is often used to evaluate pointing techniques in GUIs, we also provide a 2D generalization of this layout. Figure 2 illustrates those layouts with $ID = 4$ and $\rho = 0.25$

for 1D (top) and 2D (bottom) tasks. The main characteristics of the distractors are: a uniform index of difficulty equal to the target *ID* (they all have a form similar to the target modulo a scale and a rotation centered at the origin); and a uniform spatial distribution (the scaling factor between to successive distractors is constant, which makes the task self-similar). The actual 1D and 2D layout specifications are given in Appendixes.

Design Rationale

The design space of distractors layout for a given density is very large. We had to make several choices to propose the layouts presented above. We discuss here the main alternatives we faced and the rationale behind the choices we made.

"Realistic" vs. Random vs. Uniform Distribution

The first question is *should the distribution of distractors be "realistic"?*, i.e., try to mimic layouts commonly found on the desktop (e.g., as done by Ahlström *et al.* to evaluate their pointing techniques [3]). We excluded this approach because it is hard to design a "realistic" layout that can be parametrized with a single factor ρ. Choosing a single kind of layout would certainly introduce a bias toward particular techniques. Conversely, multiplying the situations to account for the diversity of reality would introduce too many variables that would make the results analysis harder.

The next question is *should some kind of randomness be introduced in the layout?*, like others have done (e.g, [15, 23]). The first problem with this approach is that real randomness leads to very disparate situations that can be properly compensated only by increasing the number of trials. For this reason, authors are often tempted to "control" the randomness (examples of such manipulation found in the literature include: ensuring that distractors do not overlap; ensuring that distractors are at a given distance from the target; ensuring a kind of uniformity of spatial distribution). The goal behind the introduction of controlled randomness is often to recreate "realistic" layouts. The produced layouts share their drawbacks (potential bias, hard to control). They are likely to be difficult to reproduce because the way randomness is introduced and controlled is hard to document.

Our final choice for a uniform distribution of the distractors is mainly backed up by the drawbacks of the alternatives but also because it is fully determinist, thus easy to reproduce. This distribution does not look "realistic" because it is very regular, but real settings often present regularities.

Equally Spaced in Space vs. in Scale

The most obvious way to distribute distractors uniformly is to use distractors having a *constant size* identical to the target and to choose a *constant step in space* to distribute them. Figure 3 presents such layouts in 1D. We have rejected this idea in favor of a *constant ID* identical to the target and a *constant step in scale*. This choices emerged while trying to solve an issue of the constant size layout. This issue is visible in Figure 3: the task at the top has a global distractor density $\rho \approx 0.31$ wile the task at the bottom has a global

Figure 3. Evenly distributed equal-width distractors ($ID = 3$). Same local density (two distractors between the start and the target, $\rho_L \approx 0.36$) in spite of two very different global densities: $\rho \approx 0.31$ (top) and $\rho = 0.4$ (bottom).

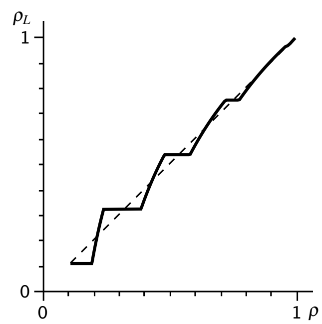

Figure 4. Local density ρ_L vs. nominal density ρ ($ID = 2.5$).

distractor density $\rho = 0.4$. However, on both paths between the start of the task and the middle of the target, exactly two distractors are present. On those paths, the proportion of space occupied by distractors is thus the same. This local density ρ_L is ≈ 0.36. The difference with the nominal ρ is $\approx +16.1\%$ for the first task and $\approx -11\%$ for the second one. Figure 4 illustrates the gap between ρ_L and ρ. The step-like shape of the curve is due to a quantization effect: different values of ρ lead to the same number of distractors on the path to the target and thus to the same ρ_L. It is likely that pointing techniques will mainly be sensitive to ρ_L rather than to ρ, and thus using ρ as a factor will probably lead to a substantial loss of statistical significance.

To confirm this hypothesis, we ran a pilot study in which a pointing technique presumably sensitive to ρ, namely *semantic pointing* [7], is tested using the constant size 1D layout. We used a single $ID = 3$, two differents $A \in \{511, 1023\}$ and 9 densities in the range $[2/15 \approx 0.13, 2/3 \approx 0.67]$ chosen at the angular points of the ρ_L vs. ρ function in order to maximize the differences between ρ_L and ρ. Two subjects performed a total of 360 trials (20 times each combination of factors). Figure 5 shows the movement time *MT* vs. the nominal density ρ on the left and vs. the local density ρ_L on the right. The similarity between the *MT* vs. ρ plot (Figure 5 left) and the ρ_L vs. ρ plot (Figure 4) is not an accident. Modeling *MT* by a linear combination of ρ and ρ_L confirms that the ρ_L parameter explains significantly the *MT* variations ($t = 3.09, p = 0.0022$) whereas ρ is rejected as a significant parameter ($t = 0,08$).

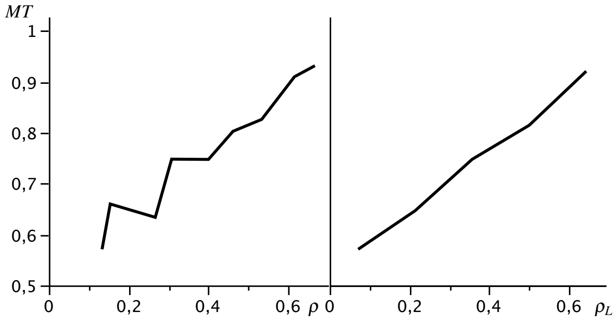

Figure 5. Semantic pointing movement time (MT in seconds) vs. nominal density ρ (left) and local density ρ_L (right) ($ID = 3$).

This pilot study shows that the naïve *constant size* approach suffers from a serious drawback: the ρ factor does not match the parameter that explains the performance. Analyzing the results would then require to transform the ρ factor into the ρ_L parameter using a non-trivial function which also depends on the *ID*. This would result in the introduction of correlated factors, which is not desirable. We have conducted this pilot study with a 1D layout, but the 2D version would probably lead to similar results since it is subject to a similar quantization at the origin.

Using the *constant ID* layout is a solution to this quantization artifact. Since it is self-similar, the distractors size tends toward zero at the origin of the task, and the local density of distractors is theoretically equal to the nominal ρ. The drawback is that it theoretically introduces an infinity of distractors. Stopping the construction when their size becomes smaller than 1 pixel (the smallest visible target on a screen) solves this issue in practice.

Other side effects of the self-similarity of the layout is that a task presents a single *ID* to the subject, and that distractor size distribution is not farther to the reality than the *constant size* layout (lots of small targets, few big targets [9]). Still, the main property of this distractor layout is the ability to accurately control the distractor density on the path of the pointing movement.

TOWARDS AN EXTENSION TO FITTS' LAW

Once a new factor is added to the parametrization of the pointing task, one can wonder how it will affect the movement time and how to account for it in Fitts' law. It obviously depends on how each specific pointing technique will take advantage of the sparseness of the task to ease the pointing but we can try some reasoning here.

As stated in the previous section, the *density of distractors* characterizes the mismatch between the abstract task —one icon among 32— and the actual pointing task —one of the pixels of the icon among the whole screen. Having a task that paves the space with distractors ($\rho = 1$) leaves no room for improvement, but if the space was only half full ($\rho = .5$), all potential targets could be made twice as big without overlapping. That would make the task easier and reduce its *ID* by roughly one bit[2]. If we reduce further the density of the task, e.g., $\rho = .25$, targets could be made four time bigger and *ID* would then decrease by 2 bits. If we define the *index of sparseness*[3] (*IS*) by:

$$IS = \log_2 \frac{1}{\rho}, \tag{4}$$

we see that it gives an approximation of the number of bits that could be gained on *ID* (e.g., $IS = 0$ for $\rho = 1$, $IS = 1$ for $\rho = .5$, $IS = 2$ for $\rho = .25$, etc.) If we hypothesize that an optimal technique could reduce *ID* by *IS* bits, the *movement time* for that optimal technique would then be:

$$MT_{opt} = a + b \times (ID - IS). \tag{5}$$

In practice however, it is more likely that different techniques will take advantage of the sparseness of the task differently. This could be measured by introducing a third parameter in Fitts' law:

$$MT = a + b \times ID - c \times IS, \tag{6}$$

where, as advocated by Zhai [29], a (resp. b) reflects the non-informational (resp. informational) aspect of input performance; and c quantifies the sensitivity of the technique to the presence of distractors. In this extended law, a low value for c means that the technique does not behave differently in the presence or in the absence of distractors (e.g., presumably the plain old mouse) and then Equation (6) falls back to the standard Fitts' law given by Equation (1). On the other hand, a value for c close to b means that the technique takes as much as possible advantage of the sparseness of the task and the law falls back to the limit case given in Equation (5).

Given the extended law formalized by Equation (6), pointing techniques could then be compared according the 2 parameters recommended by Zhai, together with a third parameter:

- a, the *intercept*, that measures the non-informational part of the performance;
- $1/b$, the *throughput*, that measures its sensitivity to the form of the task characterized by its *ID*; and
- c/b, the *sensitivity to distractors*, that measure its ability to make the most of the sparseness of the task, and to reduce *ID* by that proportion of *IS*.

We are aware that the extension of Fitts' law proposed in Equation (6) is only backed by reasoning and not by any actual fit of data. We acknowledge that *ID* and *IS* do probably interact, and that distractors play a role more complex than simply perturbing the pointing techniques by also modifying the perception of the task by the user. That is why we refer to Equation (6) as a *first step towards an extension to Fitts' law*, and leave further investigation in this direction for future works.

---

[2] The more A/W is bigger than 1 (i.e., the larger the *ID* is), the more exact is this approximation.

[3] We choose *index of sparseness* rather than *index of density* (its opposite) to avoid a possible confusion of the acronyms (*IS* vs. *ID*).

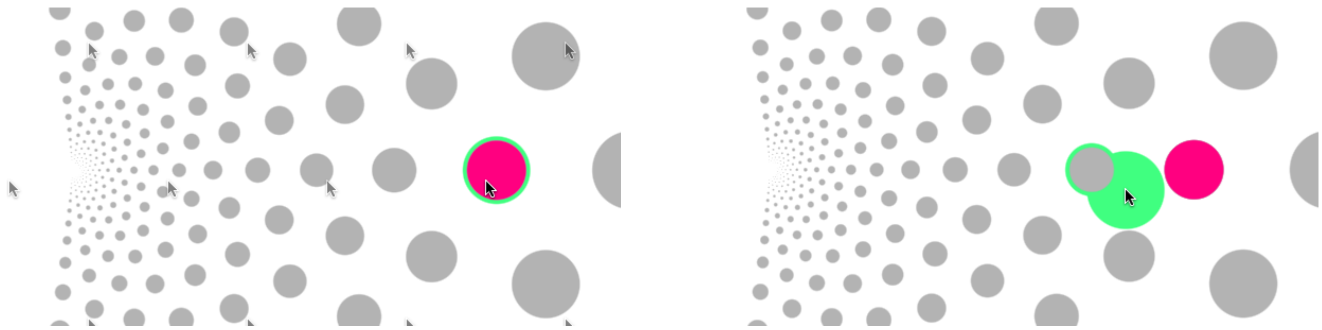

Figure 6. 2D feedbacks of *rake cursor* (left) and *Bubble cursor* (right) for the task $ID = 3$, $\rho = 0.25$.

BENCHMARKING POINTING TECHNIQUES

Having defined a pointing task from the three factors: index of difficulty (ID), task amplitude (A), and distractors density (ρ), we can use it to test the sensitivity of various techniques to those factors. Here we compare five interaction techniques: the *raw pointing* (RP) with a cursor, the *semantic pointing* (SP) technique [7], the *bubble cursor* (BC) technique [15], the *DynaSpot* (DS) technique [10], and the *rake cursor* (RC) technique [8].

This set of techniques is rather arbitrary, but our goal is to cover various strategies used to facilitate pointing: modifying the control-display ratio (SP), using a supplemental input channel (RC), or modifying the cursor activation area (BC and DS). A criterion that also motivated our selection is that those techniques are described in their respective papers with enough details to allow their precise reproduction, which is unfortunately not as common as it should be.

Pointing Technique Definition

Testing various pointing techniques made explicit the lack of definition of what is a pointing technique. We propose the following one: a pointing technique consists of handling *inputs*, providing a *picking function*, and updating a *feedback*.

Receiving relative movements from the physical pointing device and transforming it to a cursor position is the minimal *input handling*. It can involve more complex processing (e.g., for RC, mixing the mouse input and the gaze input) to maintain an internal state used by the other parts of the pointing techniques.

The *picking function* uses this internal state to compute the pixel activated when the user depresses the button of the pointing device. The picking function can be as simple as returning the cursor position (e.g., RP and SP). It can also do more complex processing such as triggering a spatial search to find the target closest to the cursor and returning the coordinates of a pixel located inside (e.g., BC).

The *feedback* usually serves the role of making the internal state of the technique and its picking function observable. It should make its behavior predictable. The minimal feedback consist of displaying a cursor arrow on the screen. It can also display the cursor activation area (BC, DS) or several arrow cursors (RC). The feedback of the BC and RC techniques is illustrated in Figure 6 for a 2D pointing task setup. Since the cursor can be outside the target that will be selected, some techniques also provide a highlight to make it observable to the user (BC, DS). In our implementation, we added this highlight to every technique because we think that its presence can affect the performance of pointing.

Implementation

The implementation we used to conduct our study is made available to the community[4] and can either serve as a reference implementation, or it can be reused as is. It includes the implementation of the techniques compared in the experiment, the implementation of the distractors layouts and the framework that drives the experiment.

Experiment

Tasks

The participants had to perform successive 2D pointing tasks. They had to move the standard cursor inside a start area (a blue disc on the middle-left of the screen), rest there for about 0.5 s. After this delay, a trial consisting of grey distractors and a red target (Figure 2 bottom) was presented and the participant had to click the target. The direction of the movement was always from the left to the right, and the participants had to come back to the start area after each trial. After each block, their error rates were displayed and they were encouraged to conform to a nominal 4% error rate by speeding up or slowing down.

Conditions and Procedure

The five *techniques* (RP, SP, BC, DS, RC) were used by each participant one after another. They always started with the *raw pointing* and the order of the four other techniques was balanced between the subjects using a latin square.

Three IDs (3, 4, 5), two As (511 and 1023 pixels), and four ISs ($\approx$ 0.74, 1.74, 2.74, 3.74) —corresponding to four ρs (0.6, 0.3, 0.15, 0.075)— were used. A randomized series of the 24 combinations was presented four times for each technique to each participant. Those four series were preceded by 12 training trials during which they were familiarized with the techniques. Each participant performed 540 pointing tasks, 480 of them being recorded.

[4] The Python/OpenGL source code is available at: http://iihm.imag.fr/blanch/projects/distractors/.

	ER	RT	MT	TT
RP	5.41	$.397 \pm .129$	$.846 \pm .193$	$1.243 \pm .268$
SP	3.28	$.376 \pm .120$	$1.021 \pm .285$	$1.397 \pm .348$
BC	3.14	$.392 \pm .135$	$.716 \pm .168$	$1.108 \pm .258$
DS	3.70	$.367 \pm .119$	$.749 \pm .177$	$1.116 \pm .243$
RC	4.89	$.470 \pm .161$	$.725 \pm .349$	$1.195 \pm .378$

Table 1. Error rate (ER in %), reaction time (RT in seconds), movement time (MT in seconds) and total time ($TT = RT + MT$ in seconds) by technique.

Subjects and Apparatus

Eight right handed unpaid adult volunteers (1 female, 7 male) served in the experiment. We used a Logitech MX 400 mouse as a pointing device, the system acceleration being discarded by reading the low level (HID) motion events. The control-display ratio was globally set to 2. For SP, it was 1 outside the distractors and 4 inside. The gaze position needed by the RC technique was acquired using a Tobii ET-17 eye tracker (17-inch 1280×800 monitor). The same monitor was used for the other techniques. The step of the RC cursor grid was fixed at 300 pixels so that the target would never be directly under a cursor at the start of a trial.

Statistical Results

The effects of the factors on the performances are explored by analyzing three dependent variables: *error rate* (ER), *reaction time* (RT) and *movement time* (MT). The *total time* (TT) of the pointing —defined as the sum of RT and MT— is also considered since for the RC technique RT is significantly different from the other ones [8]. Table 1 sums up the values[5] of those variables for the five techniques considered. The analysis of variance below use the participant as random factor and the *technique*, ID, A and IS as factors. Means are compared using Tukey's HSD tests. A total of 17 obvious outliers (among 3840 observations) caused by technical problems were removed before performing the statistics.

Error Rate

ER is 4.08% on average (very close to the 4% value users were instructed to aim for). The main significant effect on ER is contributed by ID ($F_{1,3776} = 18.74*$[6]). A second effect is found for the interaction $IS \times technique$ ($F_{4,3776} = 2.38, p = .0492$), but no other effect is present. Comparing average ERs by techniques gives no statistically significant differences. An interesting result is then that the distractor density by itself does not significantly impact the error rate.

Reaction Time

The most significant effects on RT are, from the strongest to the weakest: ID ($F_{1,3776} = 4088.25*$), IS ($F_{1,3776} = 1783.95*$), the interaction $ID \times IS$ ($F_{1,3776} = 543.65*$), and finally the technique ($F_{4,3776} = 160.61*$). Observing an effect of the technique on RT is not common in pointing experiments. Since in this case this effect is rather strong, it needs to be explained. Comparing the means of the RT by technique, confirms that *rake cursor* is significantly slower than any of the other techniques. This difference was already reported and explained for RC and RP [8], but we show here that the difference is also significant when compared to other sophisticated techniques like DS or BC. The reason is probably that with RC, the user needs to choose which cursor she will use prior to start the movement.

Movement Time

All factors do have an effect on MT, the strongest ones being contributed by ID ($F_{1,3776} = 842.00*$) and the technique ($F_{4,3776} = 254.22*$). The interactions between the technique and each of the other factors also give significant effects. Observing so many significant effects makes them difficult to interpret. Comparing the means of the MT by technique gives two simple but interesting results: *semantic pointing* is significantly slower than any other technique, including *raw pointing*; and DS, RC and BC are not significantly different but all significantly faster than RP (and thus than SP).

Looking at the rank of each technique per participant provides an interesting result: the pattern "SP slower than RP and RP slower than BC and DS" is mostly shared (in a single case DS is slower than RP, but not significantly). On the other hand, the rank of the *rake cursor* technique exhibits a lot of variability: for two of the participants, RC is slower than RP but faster than SP, and thus ranks second worst technique; for two other participants, RC ranks third (faster than SP and RP but slower than BC and DS); while for the four remaining participants, RC ranks as the fastest technique, being even significantly faster than any other technique for three of them. This analysis shows that the performance of *rake cursor* is strongly variable among participants.

Total Time

RT being particularly high for RC, their authors also study the *total time* ($TT = RT + MT$) to allow a more fair comparison of the techniques [8]. In our case, a whole analysis of TT gives qualitative results mostly equivalent to those reported above for MT. The only difference is the rank of RC: it performs significantly better than SP and RP but it is significantly slower than BC and DS (Table 1, last column).

Analysis using Extended Fitts' Law

Table 2 summarizes the parameters obtained when fitting the MT (top) and TT (bottom) data with the extended Fitts' law (EXT) proposed in Equation (6). The first column gives the intercept (a in seconds). The second one gives the throughput *à la* Zhai ($1/b$ in bits per second) of the technique The third column gives the ratio c/b which quantifies the sensitivity to distractors, i.e., how the sparseness of the task is exploited to reduce its difficulty: $c/b = 1$ meaning that each bit of IS reduces ID by one bit; and $c/b = 0$ meaning that the density has no impact on the performance.

The remaining columns of Table 2 characterize the goodness of fit of the model using two metrics: the adjusted coefficient of determination (adj. R^2) and the Akaike information criterion (AICc). Those metrics both take into account the number of degrees of freedom of the models and thus allow comparisons with the standard Fitts' law (STD, values given in parens). For modeling TT, EXT is always better (higher

[5] $m \pm \sigma$ gives the mean m and standard deviation σ.

[6] $*$ denotes $p < .0001$.

MT	*a*	$1/b$	c/b	adj. R^2		AICc	
RP	.371	7.87	.07	.65	(.65)	-206.55	(-207.24)
SP	.513	5.49	.48	.59	(.39)	-102.87	(-69.30)
BC	.423	9.62	.46	.54	(.37)	-185.04	(-158.25)
DS	.426	10.87	.17	.34	(.33)	-152.25	(-151.92)
RC	.232	7.25	.14	.28	(.27)	-55.76	(-56.62)

TT	*a*	$1/b$	c/b	adj. R^2		AICc	
RP	.481	4.35	.25	.68	(.61)	-112.78	(-95.68)
SP	.595	3.46	.48	.78	(.51)	-97.60	(-30.03)
BC	.489	4.57	.46	.70	(.48)	-113.02	(-66.17)
DS	.495	5.03	.33	.61	(.50)	-107.16	(-86.10)
RC	.395	4.03	.28	.57	(.49)	-56.19	(-42.78)

Table 2. Parameters of the extended Fitts' law given by Equation (6)**: intercept (*a* in seconds), througput (**$1/b$ **in bits per second), and sensitivity to distractors (**c/b**, dimensionless); adjusted coefficient of determination (adj.** R^2**); and the Akaike information criterion (AICc) for the movement time (top); and the total time (bottom) by technique. Numbers in parens give the corresponding values for the standard Fitts' law.**

adj. R^2, lower AICc) than STD, and for MT, EXT is better or comparable to STD. It is interesting to note that EXT and STD are comparable for *RP*, *DS* and *RC*, the techniques that are the less sensitive to distractors (lowest c/b ratios).

Regarding the techniques, some observations can be made: as expected, SP is quite sensitive to *IS*, and so is BC. This is not surprising since the bubble of BC uses the empty space surrounding the target to make it effectively larger. DS is less sensitive to *IS* than BC which is also logical since the size of its activation area is bounded, and thus do not benefit of very sparse distractors distributions. On the other hand, DS has a better throughput than BC, probably because its feedback is less disturbing. So while the two techniques perform mostly the same on average, *bubble cursor* is better suited to low *ID* or low density tasks, while *DynaSpot* is better suited to high *ID* or high density tasks. Overall, *rake cursor* has the best intercept and is the enhanced technique the least sensitive to *IS*, which makes it particularly adapted to dense tasks.

CONCLUSION AND FUTURE DIRECTIONS

We have proposed an extension to Fitts' pointing paradigm that enables to quantify the sensitivity of pointing techniques to the presence of distractors. This extension adds a single factor to the two specifying a pointing task: the target density, and derives the distractors location in 1D and 2D deterministically from those three parameters. This makes possible the fair comparison of pointing techniques. We have also proposed a first step toward an extension of Fitts' law by adding a term that accounts for the sensitivity the distractors density, namely the *index of sparseness*. With those tools at hand, we have compared five pointing techniques and shown that the extended protocol and model allow more contrasted comparison of the techniques.

We believe that the use of this protocol will allow more fair comparisons of pointing techniques. To encourage its use, we release to the community a reference implementation of the distractors layout generators, a reference implementation of the interaction techniques compared in this paper, and the framework used to conduct the experiment. We hope that this repository will grow in the future, fed by the community.

In the future, we will focus our attention on the model itself. Since the interaction of *ID* and *IS* is often significant, the model may have to consider this factor also. But this interaction has to be understood first.

Regarding the protocol, two extensions are envisioned. First, the distributions of distractors presented here is not incompatible with different experimental setups, like 1D reciprocal pointing or 2D multi-directional task following the ISO 9241-9 standard. We would like to extend our experiment framework to support those task setups. Then, providing a 3D generalization of distractors layout relying also on self-similarity and distractors density should also be doable. This would allow to extend the benchmark to 3D interaction techniques, and to test the effect of distractors on the 3D generalization of Fitts' law [14].

And finally, we will add more implementation of techniques to our framework, so that it can serve as an educational resource, as well as a benchmark that allows to compare new techniques to a whole corpus of existing techniques.

ACKNOWLEDGMENTS

The authors would like to thank Olivier Chapuis, Michel Beaudouin-Lafon, Brigitte Meillon and Laurence Nigay for their insightful comments and/or their practical help on this work. Part of this work was conducted within the NOMAD project (FUI Minalogic).

APPENDIXES

1D Distractors Layout Parameterization

Having the distractors similar to the target (i.e., having the same *ID*) modulo a constant step of scale (r) means that their widths (W_i) and amplitudes (A_i) follow a geometric progression. Having the target match the distractor W_0, A_0 leads to:

$$\forall i \in \mathbb{Z}, \quad \frac{A_{i+1}}{A_i} = \frac{W_{i+1}}{W_i} = r, \tag{7}$$

$$A_i = A \times r^i, \tag{8}$$

$$W_i = W \times r^i. \tag{9}$$

To fully define the task using our parametrization (ID, A, ρ), we need to express r using those factors.

If we consider the space separating the center of to successive tasks (dotted surface in Figure 7), we can compute a local distractor density (ρ_i) in this area by dividing the length covered by the distractors by the distance separating the distractors centers:

$$\begin{aligned} \rho_i &= \frac{W_{i+1}/2 + W_i/2}{A_{i+1} - A_i} \\ &= \frac{W_i/2}{A_i} \times \frac{r+1}{r-1} \quad \text{using Equation (7)} \\ &= \frac{W/2}{A} \times \frac{r+1}{r-1} \quad \text{using Equations (8, 9).} \end{aligned} \tag{10}$$

In Equation (10), ρ_i is not anymore a function of i, which means it is the same between each successive pair of targets, i.e., that $\rho_i = \rho$ for each i. This property is a consequence of the self-similarity we have introduced in the distribution of the targets, and confirms that the global distractor density ρ is also locally respected.

Equation (10) gives:

$$\rho = \frac{r+1}{r-1} \times \rho_{min}$$
$$\iff \quad r = \frac{\rho + \rho_{min}}{\rho - \rho_{min}}, \tag{11}$$

provided we define ρ_{min}, the lower bound for the distractor density[7], as:

$$\begin{aligned}\rho_{min} &= \frac{W/2}{A} \\ &= \frac{1}{2} \times \frac{1}{2^{ID}-1} \quad \text{using Equation (3).}\end{aligned} \tag{12}$$

Finally, using Equations (8, 11, 12) and (3, 9), a full parameterization of the distractors depending only on the factors (ID, A, ρ) can be derived:

$$\forall i \in \mathbb{Z}, \quad A_i = A \times \left(1 + \frac{1}{(2^{ID}-1)\rho - 1/2}\right)^i,$$
$$W_i = \frac{A_i}{2^{ID}-1}.$$

Those equations give an infinity of targets. In practice, constraints on a minimal size for the distractor (not less than a pixel) and a maximal distance (not more than the screen width) can be used to bound the range of i.

2D Distractor Layout Parameterization

The construction of the 2D set of distractors is a bit more complex but relies on the same principle of linking the local density to the global ρ. The global density ρ is first divided into two contributions: a tangential density ρ_t and a radial density ρ_r, such as the product of those contribution remains equal to the whole distractors density:

$$\rho_t = \sqrt{\frac{\rho}{k}},$$
$$\rho_r = \sqrt{\rho \times k} \quad \text{with } k = \frac{\pi}{4} \sin \frac{\pi}{3}. \tag{13}$$

The constant k is chosen to create an hexagonal packing of the distractors, which is desirable because the hexagonal packing is the densest circle packing and thus allow exploring distractors density up to $\frac{\pi}{6}\sqrt{3} \approx 0.91$[8].

The tangential density ρ_t is used to link the β angle between two rays of distractors to the α half-angle under which the

[7] ρ_{min} can be seen as the task density when there is no distractor: on the path from the start towards the center of the target (A), only the half of the target present before reaching the center is not empty space ($W/2$), thus a minimal density of $\frac{W/2}{A}$.

[8] E. W. Weisstein. Circle Packing. 2008.
http://mathworld.wolfram.com/CirclePacking.html

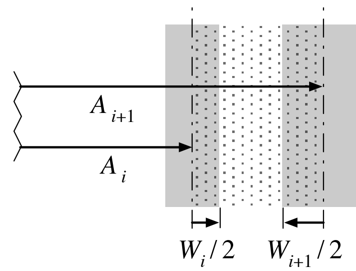

Figure 7. Computing local distractors density in 1D.

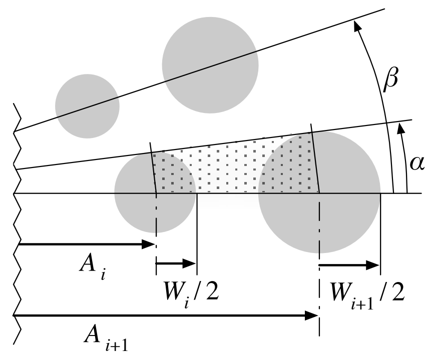

Figure 8. Computing local distractors density in 2D.

distractors are seen from the origin (see Figure 8):

$$\begin{aligned}\beta = \frac{2\alpha}{\rho_t} \quad \text{with } \alpha &= \sin^{-1}\frac{W/2}{A} \\ &= \sin^{-1}\frac{1}{2} \times \frac{1}{2^{ID}-1}.\end{aligned} \tag{14}$$

As for the 1D task, the scale linking the sizes of two consecutive distractors in the same ray (r) can be linked to the density by considering the dotted surface in Figure 8:

$$r = \sqrt{\frac{\pi}{\alpha - \pi/2 + \rho_r/\tan\alpha} + 1}. \tag{15}$$

The distractors can then be defined by:

$$\begin{aligned}&\forall j \in \mathbb{Z} / j \times \beta \in [-\pi/2, \pi/2], \quad \text{(each ray)} \\ &\qquad \forall i \in \mathbb{Z}, \quad \text{(each distractor on the ray)} \\ &\qquad A_{i,j} = \begin{cases} A \times r^i & \text{if } j \text{ is even} \\ A \times r^{i+1/2} & \text{if } j \text{ is odd} \end{cases}, \\ &\qquad W_{i,j} = \frac{A_{i,j}}{2^{ID}-1}, \\ &\qquad \begin{pmatrix} x \\ y \end{pmatrix}_{i,j} = A_{i,j} \times \begin{pmatrix} \cos\beta j \\ \sin\beta j \end{pmatrix}.\end{aligned} \tag{16}$$

Equations (3, 13, 14, 15) makes it possible to express $A_{i,j}$ and $W_{i,j}$ using only the factors (ID, A, ρ) as parameters. Equation (16) gives the positions of the centers of the distractors in traditional cartesian coordinates.

REFERENCES

1. Accot, J., and Zhai, S. Refining Fitts' law models for bivariate pointing. In *Proc. CHI'03*, ACM Press (2003), 193–200.
2. Ahlström, D. Modeling and improving selection in cascading pull-down menus using Fitts' law, the steering law and force fields. In *Proc. CHI'05*, ACM Press (2005), 61–70.
3. Ahlström, D., Hitz, M., and Leitner, G. An evaluation of sticky and force enhanced targets in multi target situations. In *Proc. NordiCHI'06* (2006), 58–67.
4. Appert, C., Beaudouin-Lafon, M., and Mackay, W. Context matters: Evaluating interaction techniques with the CIS model. In *Proc. HCI'04* (2004), 279–295.
5. Balakrishnan, R. "Beating" Fitts' law: virtual enhancements for pointing facilitation. *Int. J. Hum.-Comput. Stud. 61* (2004), 857–874.
6. Baudisch, P., Cutrell, E., Hinckley, K., and Eversole, A. Snap-and-go: helping users align objects without the modality of traditional snapping. In *Proc. CHI'05*, ACM Press (2005), 301–310.
7. Blanch, R., Guiard, Y., and Beaudouin-Lafon, M. Semantic pointing: improving target acquisition with control-display ratio adaptation. In *Proc. CHI'04*, ACM Press (2004), 519–526.
8. Blanch, R., and Ortega, M. Rake cursor: Improving pointing performance with concurrent input channels. In *Proc. CHI'09*, ACM Press (2009), 1415–1418.
9. Chapuis, O., Blanch, R., and Beaudouin-Lafon, M. Fitts' law in the wild: A field study of aimed movements. Tech. Rep. 1480, 2007. Université Paris-Sud XI.
10. Chapuis, O., Labrune, J.-B., and Pietriga, E. Dynaspot: speed-dependent area cursor. In *Proc. CHI'09*, ACM Press (2009), 1391–1400.
11. Douglas, S. A., Kirkpatrick, A. E., and MacKenzie, I. S. Testing pointing device performance and user assessment with the ISO 9241, Part 9 standard. In *Proc. CHI'99*, ACM Press (1999), 215–222.
12. Fitts, P. M. The information capacity of the human motor system in controlling the amplitude of movement. *J. of Exp. Psycho. 47* (1954), 381–391.
13. Graham, E. D., and MacKenzie, C. L. Physical versus virtual pointing. In *Proc. CHI'96*, ACM Press (1996), 292–299.
14. Grossman, T., and Balakrishnan, R. Pointing at trivariate targets in 3D environments. In *Proc. CHI'04*, ACM Press (2004), 447–454.
15. Grossman, T., and Balakrishnan, R. The bubble cursor: Enhancing target acquisition by dynamic resizing of the cursor's activation area. In *Proc. CHI'05*, ACM Press (2005), 281–290.
16. Grossman, T., and Balakrishnan, R. A probabilistic approach to modeling two-dimensional pointing. *ACM Trans. Comput.-Hum. Interact. 12*, 3 (2005), 435–459.
17. Guiard, Y. Disentangling relative from absolute amplitude in Fitts' law experiments. In *Proc. CHI'01*, ACM Press (2001), 315–316.
18. Guiard, Y. The problem of consistency in the design of Fitts' law experiments: consider either target distance and width or movement form and scale. In *Proc. CHI'09*, ACM Press (2009), 1809–1818.
19. Guiard, Y., Blanch, R., and Beaudouin-Lafon, M. Object pointing: a complement to bitmap pointing in GUIs. In *Proc. GI'04* (2004), 9–16.
20. Hoffmann, E. R. Effective target tolerance in an inverted Fitts task. *Ergonomics 38*, 4 (1995), 828–836.
21. Kabbash, P., and Buxton, W. The "Prince" technique: Fitts' law and selection using area cursors. In *Proc. CHI'95*, ACM Press (1995), 273–279.
22. Keyson, D. V. Dynamic control gain and tactile feedback in the capture of cursor movements. *Ergonomics 40*, 12 (1997), 1287–1298.
23. Kobayashi, M., and Igarashi, T. Ninja cursors: using multiple cursors to assist target acquisition on large screens. In *Proc. CHI'08*, ACM Press (2008), 949–958.
24. MacKenzie, I. S. A note on the information-theoretic basis for Fitts' law. *J. of Mot. Behav. 21* (1989), 323–330.
25. MacKenzie, I. S., and Buxton, W. Extending Fitts' law to two-dimensional tasks. In *Proc. CHI'92*, ACM Press (1992), 219–226.
26. McGuffin, M. J., and Balakrishnan, R. Fitts' law and expanding targets: Experimental studies and designs for user interfaces. *ACM Trans. Comput.-Hum. Interact. 12*, 4 (2005), 388–422.
27. Soukoreff, R. W., and MacKenzie, I. S. Towards a standard for pointing device evaluation, perspectives on 27 years of Fitts' law research in HCI. *Int. J. Hum.-Comput. Stud. 61*, 6 (2004), 751–789.
28. Worden, A., Walker, N., Bharat, K., and Hudson, S. Making computers easier for older adults to use: area cursors and sticky icons. In *Proc. CHI'97*, ACM Press (1997), 266–271.
29. Zhai, S. Characterizing computer input with Fitts' law parameters — the information and non-information aspects of pointing. *Int. J. Hum.-Comput. Stud. 61*, 6 (2004), 791–809.
30. Zhai, S., Conversy, S., Beaudouin-Lafon, M., and Guiard, Y. Human on-line response to target expansion. In *Proc. CHI'03*, ACM Press (2003), 177–184.