Static Voronoi-Based Target Expansion Technique for Distant Pointing

Maxime Guillon [1, 2] François Leitner [2] Laurence Nigay [1]

[1] Univ. Grenoble Alpes, LIG,
F-38000 Grenoble, France
CNRS, LIG, F-38000 Grenoble, France
{maxime.guillon, laurence.nigay}@imag.fr

[2] Aesculap SAS
1 place du Verseau,
F-38130 Échirolles, France
{maxime.guillon, francois.leitner}@bbraun.com

ABSTRACT

Addressing the challenges of distant pointing, we present the feedforward static targeting assistance technique VTE: Voronoi-based Target Expansion. VTE statically displays all the activation areas by dividing the total screen space into areas such that there is only one target inside each area, also called Voronoi tessellation. The key benefit of VTE is in providing the user with an immediate understanding of the targets' activation boundaries before the pointing task even begins: VTE then provides static targeting assistance for both phases of a pointing task, the ballistic motion and the corrective phase. With the goal of making the environment visually uncluttered, we present a first user study to explore the visual parameters of VTE that affect the performance of the technique. In a second user study focusing on static versus dynamic assistance, we compare VTE with Bubble Ray, a dynamic Voronoi-based targeting assistance technique for distant pointing. Results show that VTE significantly outperforms the dynamic assistance technique and is preferred by users both for ray-casting pointing and relative pointing with a hand-controlled cursor.

Categories and Subject Descriptors

H.5.2 [**Information interfaces and presentation**]: User Interfaces– *graphical user interfaces, input devices and strategies.*

General Terms

Design, Human Factors, Experimentation

Keywords

Distant pointing; Target assistance; Target expansion; Voronoi; Ray-casting pointing; Relative pointing

1. INTRODUCTION

Pointing at displays from a distance is more and more common in various contexts: public/private, domestic/professional. Distant pointing techniques are therefore increasingly important but suffer from several problems including accuracy, amplification of small movements for distant targets, hand jitter and fatigue. To address these challenges of distant pointing, several targeting assistance techniques [1] [10] [15] [20] have been studied for improving performance. These targeting assistance techniques are dynamic and provide feedback during the pointing task.

In this paper we investigate feedforward techniques, and we present a technique, VTE (Voronoi-based Target Expansion) that divides the total screen space into areas such that there is only one target inside each area, also called Voronoi tessellation (polygons displayed on screen in Figure 1). By displaying the activation area of each target, VTE admittedly modifies the on-screen appearance of the targets. The key benefit of VTE however, is to provide the user with an immediate understanding of the targets' activation boundaries before the pointing task even begins.

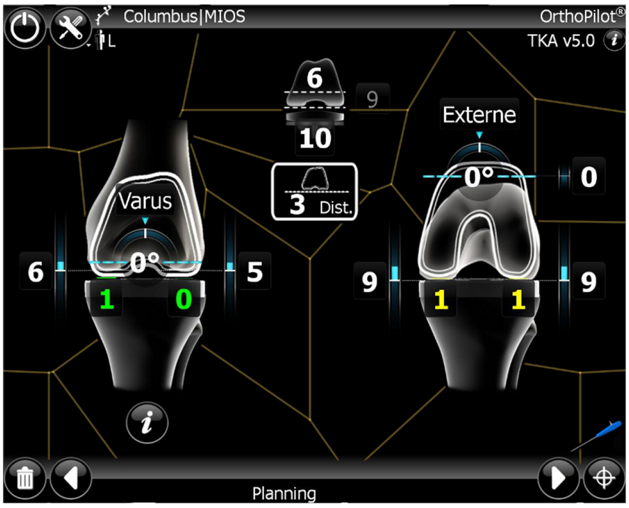

Figure 1: VTE, a Voronoi-based Target Expansion technique for improving distant pointing: VTE statically displays the expanded targets defined by their Voronoi cells (orange polygons on the screen of the Aesculap's OrthoPilot® Navigation System).

We first introduce our application domain and then review previous efforts at facilitating distant pointing by emphasizing the static versus dynamic aspect of the proposed techniques. We then present the design of our static Voronoi-based target expansion technique VTE and compare the performance of our technique with that of a dynamic target expansion technique [7] [18] with varying target densities. We conclude by discussing implications for the design of user interfaces and in particular for our application domain, namely augmented surgery.

AVI '14, May 27 - 30, 2014, Como, Italy
Copyright 2014 ACM 978-1-4503-2775-6/14/05…$15.00.
http://dx.doi.org/10.1145/2598153.2598178

2. APPLICATION DOMAIN: AUGMENTED SURGERY

During a surgery, we study how the surgeon can point at a distance to a graphical element on screen. The key constraints are the limited space for the screen in operating theaters, the strict boundary between what is sterile and what is not [16], and the goal of minimizing the movement of the surgeon away from the patient. Such surgical settings constraints make distant touchless interaction with the interactive system mandatory. In this context we focus on navigation systems for the orthopedic surgery specialty (e.g., Aesculap's OrthoPilot® Navigation System): they are systems that enhance the surgical workflow by helping the surgeon to measure and plan the surgery for an optimal implant alignment. As illustrated by Figure 2, the surgeon must remotely interact with widgets of approximately 80 pixels large from a distance of between 1 to 3 meters. To do so, the surgeon remotely points at a widget by relative ray-casting using a surgical instrument that is tracked in 3D space (e.g., Aseculap's pointer in Figure 2). For instance in Figure 1, the surgeon will point at the white numbers that specify the sizes of the knee prosthesis. S/he will then validate the selection of the number using a pedal.

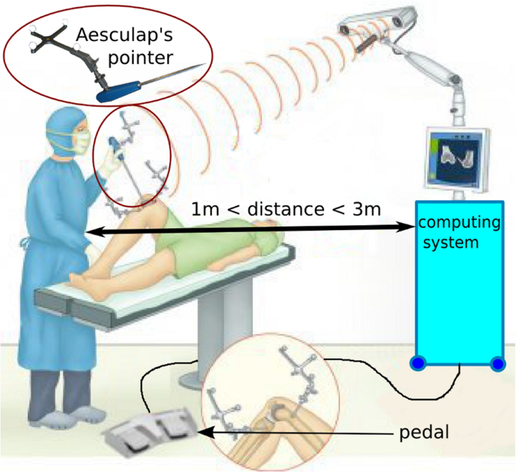

Figure 2: Setting for distant interaction during surgery: the case of Aesculap's orthopedic navigation system.

Distant interaction techniques [16] including voice and gesture have been studied in the context of augmented surgery. On the one hand, some studies focus on the tracking and recognizer systems such as the finger gesture recognizer using the Kinect depth camera [8]. On the other hand, other studies focus on the design of the command languages such as the one- and two-handed gesture interaction described in [16] for vascular surgery. We adopt a complementary approach by focusing on distant pointing techniques and in particular on targeting assistance techniques in order to improve the performance (accuracy and movement time) for both ray-casting pointing and relative pointing with a hand-controlled cursor [20]. By doing so, on the one hand our work is grounded in improving distant pointing for the case of augmented surgery: several design choices have been taken having this medical context in mind and the experimental setting reproduces the one of augmented surgery (Figure 2). On the other hand, since we explore targeting assistance techniques, our work is general.

3. RELATED WORK

Targeting assistance techniques aim at facilitating the pointing task. Pointing at a target being a fundamental task, assistance techniques have been studied in various settings: pointing at targets on desktop using a mouse, 2D/3D pointing in virtual and augmented reality and pointing at a distance. We first review target assistance techniques in general, before presenting target assistance techniques for distant pointing.

3.1 Targeting Assistance Techniques: Static or Dynamic?

As described by the optimal initial impulse model [14], the process of pointing at a target includes two phases: a first ballistic motion (large and rapid movement) and if necessary (i.e. target not reached by the ballistic movement) a corrective phase with slower movements implying closed-loop feedback control. The Fitts' law models the performance of this two-phase process [1]. The Fitts' law states that two non-exclusive ways to reduce the difficulty of a pointing task are by reducing the distance from the starting point (A-Amplitude) to the target and/or by enlarging the target (W-Width). In [1] [7] [18] a review of targeting assistance techniques is based on these two parameters A and W. Moreover the two parameters can be applied to the visual, the motor or both the visual and motor spaces. In [2] [3] [12], a review of techniques is based on these three possibilities to modify A and W. We focus on targeting assistance techniques that enlarge the effective size of targets (W) both in the visual and motor spaces.

For target enlargement, dividing the motor space into areas is beneficial only if the motor space contains empty space (i.e. on-screen space not used for interaction): indeed the effective sizes of the targets are extended into the adjacent empty space. To avoid ambiguity (i.e. several targets in the same area) a layout strategy is the Voronoi tessellation [9] that maximizes use of empty space. Several target assistance techniques [3] [7] [18] are based on the Voronoi tessellation as a static decomposition of the motor space. The Starburst algorithm [3] extends the Voronoi partitioning algorithm to the case of clusters of targets. The resulting technique is a motor and visual target expansion one. Starburst provides a static targeting assistance by directly displaying the expanded target boundaries and has been tested on a tablet. Bubble Cursor [7], another Voronoi-based target expansion, dynamically displays the visual representation of the expanded targets: indeed a semi-transparent bubble surrounding a crosshair representing the center of the cursor dynamically changes its size to enclose the nearest target based on the corresponding Voronoi cell. In the context of Voronoi-based target expansion, the design issue that we explore in this paper is the static versus dynamic aspect of the assistance.

The static/dynamic issue is related to when to provide assistance. On the one hand, since only the corrective phase of a pointing task involves closed-loop feedback control, many studies focused on dynamic target assistance based on cursor movements for this corrective phase. Moreover during this corrective phase, very late assistance by target expansion has been shown to be valuable even if the user has already moved 90% of the distance [13] and even if the user does not know whether the target will expand or not [21]. On the other hand, static assistance has the power to help the two phases of a pointing task, by assisting in both planning of the ballistic motion and performing the corrective phase. To our knowledge, this paper presents the first quantitative study on this static/dynamic aspect of the visual assistance that we study for pointing at a distance.

3.2 Targeting Assistance Techniques for Distant Pointing

For modeling performance of distant pointing, the Fitts' law has been extended to consider the angular amplitude of movement and the angular size of the target [11]. Some of the above described assistance techniques have been studied in the context of distant pointing. First for distant pointing on a tabletop display, three exiting techniques have been compared in [17]: expanding the cursor, expanding the target and snapping to the target. The Snap-to-Target technique was quickest and preferred by users. In [18] the bubble cursor technique [7] that adds a 'bubble' around the closest target has been implemented for distant pointing and called Bubble Ray. For the case of a dense space, Bubble Ray has been extended to Speech-Filtered Bubble Ray. Speech is used to filter clustered distractors. Speech-Filtered Bubble Ray outperforms Bubble Ray that in turn outperforms simple ray-casting pointing. Recently and more closely related to our distant pointing setting, a study [2] of target assistance techniques for improving accuracy in distant ray-casting pointing compares two motor-space techniques, sticky targets and target gravity. The study shows that motor-space assistance techniques are highly effective for distant ray-casting pointing. In our study and as opposed to [2], we focus on both visual and motor space assistance techniques. Moreover all the studied assistance techniques for distant pointing are dynamic. In the following section we present VTE, a static assistance technique for distant pointing.

4. VTE DESIGN

Figure 1 illustrates VTE (Voronoi-based Target Expansion). VTE is a static target expansion technique that displays a Voronoi tessellation to enlarge the effective size of the targets during a distant pointing task.

4.1 Target Expansion

We chose to implement a technique that expands targets in both visual and motor space, trusting the principle of observability: a direct mapping between the system state and its representation, between the perceived visual space and the effective motor space. However VTE does not transform the target's original appearance, it only adds an extended activation zone around it for the pointing mode. To aim for a target, the user can point anywhere inside the target Voronoi cell. Thus, the technique is simple to understand and the GUI ergonomics are preserved. Such an adopted approach assumes that the GUI has available empty space in both motor and visual space, making target expansion possible: this is the case for the augmented surgery systems we studied (Figure 1).

4.2 Static Visual Assistance

Like Starburst [3], VTE provides a static targeting assistance by directly displaying the expanded target boundaries. As opposed to Starburst, since our applications (e.g., Figure 1) do not include clusters of targets, VTE implements the original Voronoi tessellation algorithm that has been demonstrated to be asymptotically optimal [9]. Moreover Starburst has been tested on a tablet, and we designed VTE for distant pointing.

As previously stated, by displaying a static feedforward visual representation of the effective motor target expansion, we expect the users to take full advantage of this expansion, not only for the corrective phase of the pointing task (last 10% of the movement as explained in [13]) but also for the ballistic motion phase. The optimal initial impulse model [14] suggests that the standard deviation to the endpoint increases with the distance covered by the movement and decreases with the movement time. By providing the users the possibility to exploit a larger target width, we expect that they will accordingly adapt their ballistic movements and the speed-accuracy trade-off. It is therefore important that the users have a realistic knowledge of the exact target boundaries in order to optimize their movements from the beginning of the pointing task. This benefit is particularly important for distant pointing in order to effectively adapt the mid-air movement.

5. EXPERIMENT 1: TUNING THE VTE TECHNIQUE

To fully design VTE, the goal of our first experiment was to determine which form is the best to statically display the Voronoi diagram of VTE in order to make the environment visually simple and uncluttered. As a second purpose of the experiment and even if our choice was to trust a static feedforward, we wanted to examine the possibility that an additional dynamic feedback could further improve the performance of VTE.

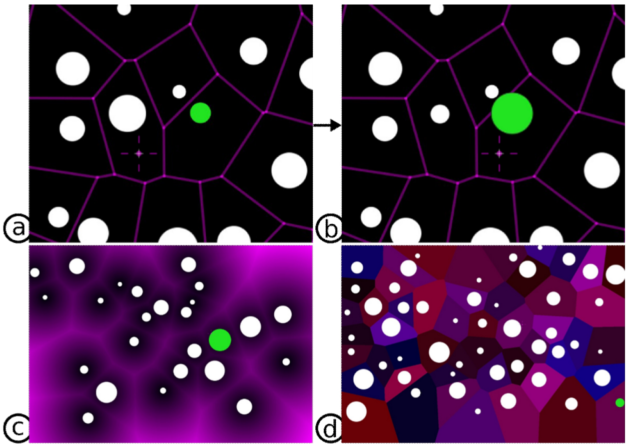

Figure 3: Evaluated visualization variants of VTE: (a → b): EDGE visualization and target enlargement mechanism. (c): DMAP visualization. (d): TILE visualization.

We considered three different static visualizations of the Voronoi diagram. The first visualization variant (EDGE) involved directly representing the full diagram with semi-transparent violet lines (Figures 3a and 3b). A second representation (DMAP for distance map) involved displaying of a distance map based on the target positions (Figure 3c). The distance of each violet pixel to the nearest target is coded by its transparency (on a black background). The longest distance separating a pixel and a target center was computed to define the most opaque pixel. Thus, the represented distances used the full transparency range. This representation visually presents the Voronoï tessellation in a more smooth and continuous way than the raw diagram (EDGE). The third variant (TILE) was a representation of Voronoï cells in different colors. A specific color theme was chosen, from pink to violet and blue. This representation defines a kind of paving of the GUI based on selectable element positions (Figure 3d).

As previously explained, we also wanted to explore the possibility that a dynamic feedback could improve performances or simply be better than a static feedforward. So, we added this feedback condition, called “Target” for target enlargement (Figures 3a and 3b), consisting of doubling the target size only when the cursor hovered over its Voronoi cell. In other words, only the target which is the closest to the cursor is doubled in size. This is a purely visual expansion to highlight the current target, that we combined with the three Voronoi visualizations (EDGE-Target, DMAP-Target, TILE-Target). We also considered a technique that implements this target enlargement dynamic feedback but with no representation of the Voronoi diagram (NONE-Target): Based on the same motor target expansion as the other techniques, the NONE-Target technique provides only a dynamic feedback (i.e. the current target based on the hovered Voronoi cell) to the user. The comparison of this NONE-Target technique with the other static techniques explores the possibility that a dynamic feedback could be better in performance than a static feedforward. In the experiment we have therefore compared the following 8 techniques:

- 3 techniques with a static visualization of the Voronoi diagram (EDGE, DMAP and TILE);
- 3 techniques with a static visualizations of the Voronoi diagram and with a dynamic feedback of the selected target (EDGE-Target, DMAP-Target, TILE-Target);
- 1 technique with no visual representation of the Voronoi diagram but with the dynamic feedback of the selected target (NONE-Target);
- A classical pointing technique with no target expansion (NOEXPANSION).

5.1 Apparatus and Participants

The display we used was a 20" LCD display at 1280x1024 resolution. For pointing at a distance, we used a hand-tracking technology, which provided a relative pointing system as defined in [20] : the movements of the hand are projected onto the vertical plane of the screen in order to control the cursor position on screen. The hand-tracking was executed with an Asus Xtion Pro Live device, using the OpenNi 2 default hand-tracking algorithm and the Nite 2 middleware. This device is easily portable (as opposed to the vision-based Aesculap’s solution) and offers a good vision angle of the scene. The hand-tracking hardware was simply put on the table, just in front of the screen. The movements of the user's hand were directly projected onto the screen to control the cursor position. The cursor then followed the hand movement with a control-display ratio at 1, without acceleration. For the selection phase, a foot-switch three button pedal was used. The Aesculap's software platform managed the interaction devices and provided the background as well as the cursor appearance. The software application of the experiment was running on a 2,3 GHz Quadcore PC with Windows 7.

Six unpaid volunteers (5 female, 1 male; 5 right-handed and 1 left-handed), ranging in age from 25 to 31 years, were recruited from our institution. All participants used their dominant hand to control the relative pointing system.

5.2 Procedure and Design

The participants stood 2 meters away from the screen. With this setting, we reproduced conditions that are common in the sterile context of an operating room. We asked the participants to acquire targets. The distractors were represented in white, and the current target highlighted in green (Figure 3). They controlled the cursor by moving their dominant hand in the space in front of them. The selection was made by pressing the right button of the foot pedal. The selection was triggered at the press of the button. When the cursor was out of the screen, selection events were not taken into account and a horn sound was played to give a feedback of an inappropriate action.

The controlled independent variables were the movement amplitude (630 and 840 pixels), the target width (40 in Figure 3a and 100 in Figure 3b), the distractor number (9, 24 in Figure 3b and 49 in Figure 3a) and the pointing technique: EDGE, EDGE-Target, DMAP, DMAP-Target, TILE, TILE-Target, NONE-Target and NOEXPANSION. Distractors (white targets) were randomly positioned, with the condition that a distractor must not overlap another one or the current target. Their sizes were randomly determined between 40 and 100 pixels. We used a within-subject fully crossed design of these variables, thus resulting in 12 combinations of conditions (2 amplitudes x 2 target widths x 3 distractor numbers) and 8 techniques.

The users first completed a training phase of 2 sets of targets (1 set corresponding to 12 selections in the 12 conditions) for each technique. Then, 3 blocks of 2 sets were completed for each technique. The order of the 12 conditions was randomly generated. Every set of 12 selections was preceded by a transition task consisting of selecting a target, which was 100 pixels wide and positioned at the left of the screen. Then, the position of the next target was calculated with random parameters to fit entirely in the screen space at the desired distance from the previous one. Thus, when performing a selection, the position of the next target could not be anticipated. At most, the user could understand that there was an alternation between both sides of the screen. All the visualizations with every target and distractor positions were generated before the test (background calculation). The total experiment lasted approximately 1 hour per participant. The participants were allowed to rest at every transition task.

5.3 Results

Performance was measured in selection times and error rates. In Figure 4 the techniques are ordered from left to right in increasing order of mean selection time (left) and error rate (right). For selection time, we performed Mann-Whitney-Wilcoxon tests between the 8 techniques. For error rate, we performed a Pearson’s Chi-squared independence test between success of target selection and the 8 techniques. As expected, all target expansions techniques largely and significantly outperformed the classical pointing technique (NOEXPANSION: mean selection time was 3.14s and error rate 28.7%) in both selection time and error rate ($p < 0.001$).

5.3.1 Error rate for the target expansion techniques

Except for NOEXPANSION, there was no significant dependence for error rate on technique. EDGE and TILE were the less error prone with an error rate of 4.86%. DMAP-Target had an error rate of 6.48% and TILE-Target an error rate of 7.11%, while the three other conditions had an equal error rate of 7.18%.

5.3.2 *Selection time for the target expansion techniques*

Without target enlargement, the differences between the mean selection times of the three visualization variants were all significant ($p < 0.01$). Furthermore, the NONE-Target condition (dynamic feedback) was significantly faster (2s) than DMAP and TILE conditions (static feedforward). EDGE was the fastest of the three visualization variants with a mean selection time of 1.98s ($p < 0.01$ with all others), followed by DMAP and TILE, with a mean selection time of 2.07s and 2.14s respectively. Moreover, EDGE was significantly the fastest of all conditions ($p < 0.01$).

DMAP was significantly faster than DMAP-Target, with a mean selection time of 2.07s and 2.16s respectively, while TILE and TILE-Target were not significantly different in selection time (2.139s). As EDGE-Target was also significantly slower than EDGE, we concluded that the dynamic feedback (i.e. target enlargement mechanism) did not lead to any improvement in performance when a static feedforward was already implemented.

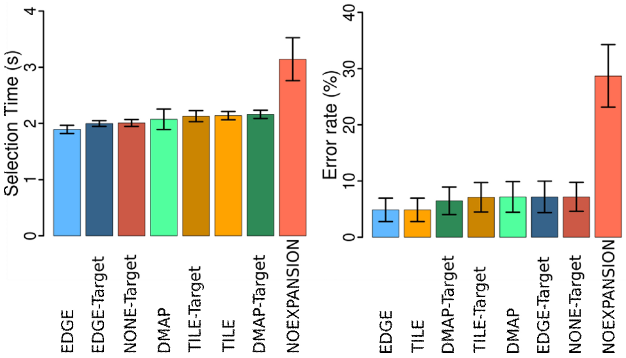

Figure 4: Mean Selection Time and Mean Error Rates per technique, with 95% confidence intervals.

5.4 Final Design

Because of its significant superiority in terms of performance, we selected the EDGE visualization for VTE. We also chose to keep only the static part of the target expansion, without any dynamic highlight of the current target. These choices based on performance, have been presented to professionals from the medical domain. A final version of VTE applied to Aesculap's software in Figure 1 represents Voronoi edges with semi-transparent orange 4 pixels wide lines only in empty visual space.

This first experiment confirms our design rationale that a static feedforward (EDGE visualization) is more efficient than a dynamic feedback that highlights the target according to the position of the cursor. Moreover, combinations of both are not faster or less error prone. In a second experiment, we further explore static feedforward versus dynamic feedback by comparing the final version of VTE (Figure 1) with Bubble Ray [18]. Bubble Ray is a Voronoi-based technique that dynamically enlarges the cursor activation area to reach only the closest target (Figure 5a). For controlling the cursor, we consider two poiting systems: first the original Bubble Ray technique based on ray-casting and second, a modified version of Bubble Ray based on relative pointing with hand-tracking instead of ray-casting.

6. EXPERIMENT 2

We compare VTE and Bubble Ray, two techniques relying on the same principle: a motor target expansion based on a Voronoi tessellation. We hypothesize that:

- H1: VTE performs better than the Bubble Ray in both selection speed and error rate.
- H2: VTE is perceived more usable and intuitive than the Bubble Ray.

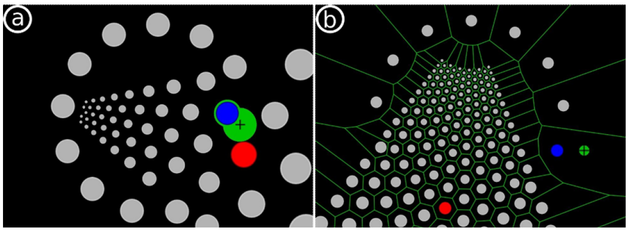

Figure 5: (a) Bubble Ray and (b) VTE implementations.

6.1 Target Assistance Techniques

The developed Bubble Ray technique was similar to the one in [18] and [7]: The algorithm governing the bubble's size is the one described in [7] and the diameter of the bubble was 12 pixels bigger than the target's diameter (Figure 5a). The VTE version represented Voronoi cells with green lines, wide enough to be seen at a distance of 2 meters (Figure 5b). Distractors were represented with grey disks, the target with a red one and the current hovered target in blue. We kept the black background from experiment 1. The crosshair was black and surrounded by a small green disc. If the selected target was not the right one, it became red during 100 ms (green in the opposite case) and the next target layout was displayed.

6.2 Apparatus and Participants

The experimental setting was identical to the first, except:

- Pedal selection: Because of the fatigue induced by the intensive use of the pedal during experiment 1, the selection of a target was triggered at the release of the pedal button in experiment 2.
- Relative ray-casting pointing: In addition to relative pointing, we considered relative ray-casting pointing [2], this technique being currently used in operating rooms. We used the Aesculap' s virtual pointer: an NDI Polaris Spectra IR camera is tracking a rigid-body, which is clipped on a surgical instrument (Figure 2, top left). This pointing system is rapid with a precision at the pointer tip of 1 mm and 1° in every direction.

Thirteen unpaid volunteers (6 females, 7 males; 12 right-handed, 1 left-handed), ranging from 25 to 67 years, were recruited from our laboratory. All participants were daily computer users and used their dominant hand to control the pointing devices. Two participants daily used a body motion tracking (Kinect device) for playing video games.

6.3 Procedure and Design

A 2D static target selection task, based on standard ISO 9241-9 extended with distractors [4], was used for this experiment. The participant had to select a set of targets placed on a circle, whose diameter was then the pointing movement amplitude. The circles contained 13 targets. The first target to be selected constituted the transition task, during which no distractor was displayed except those on the ISO 9241-9 circle. The 12 following target selections were recorded and constituted the measure. For these 12 selections, the distractors were generated according to the 2D version of the layout proposed in [4]. The distractor layout was centered on the previously selected target (i.e. the approximate start point of the movement) and rotated according to the movement direction. Targets smaller than 5 pixels or out of an angle of 60° (centered on the movement direction) were removed. Contrary to experiment 1, each target and distractor positions could be anticipated by the subjects.

The independent variables were the *Pointing System* (ray-casting RayC or relative pointing RelP), the assistance *Technique* (VTE or Bubble), the *Amplitude* (600 or 800 pixels), the index of difficulty *ID* (3, 3.5 or 4) of the pointing task, and the distractor *Density* (0.1, 0.3, 0.6 and 0.9). 0.91 is the maximal possible density for a polygonal packing of circular targets [4]. We used Shannon's *ID* formulation to compute target width. We used a repeated measures within-subject fully crossed design of all independent variables:

- one session per *Pointing System* x *Technique* per participant (resulting in 4 sessions, whose presentation order was counterbalanced across participants using a latin square design)
- one ISO 9241-9 trial block per *Amplitude* x *ID* x *Density* (resulting in 24 trial blocks, whose order was systematically randomized) per session

Participants performed one trial block for training before each session. Each trial block resulting in 1 transition task and 12 selection tasks, 1152 selections were recorded for each participant. A wrong target selection was counted as an error. The global error rate was calculated and updated for each trial. It was displayed during the transition tasks, with a message indicating if the participant should slow down or speed up to observe a target error rate of 4%. Participants could rest before starting each transition task. Between two consecutive sessions, the participants were asked to fill a questionnaire based on System Usability Scale [5] about the selection system they just experienced. They finally filled a comparative questionnaire concerning the entire experiment.

6.4 Results: Selection Time and Error Analysis

From 14976 selections, 40 obvious outliers were removed. We calculated an effective index of difficulty (*IDe*) based on a circular approximation of the effective target activation zone (i.e. the incircle centered on the target center). The distractor layout generated Voronoi cells in hexagonal shapes, slightly stretched in the movement direction [4]. So the effective target (*EW*) radius was the distance to the closest cell border, which was the border between the target and the movement starting point, perpendicular to the movement direction. *EW* used in *IDe* formulation was calculated using r value of appendixes in [4].

As expected, the differences between pointing devices (in latency and accuracy) were reflected in selection time and error rate. We used paired-t tests to compare selection times between the different *Pointing System* x *Technique* conditions. The overall mean selection times were 2.35 s for relative pointing and 1.92 s for ray-casting (t = 28.52*[1]). For error rate, we performed Pearson’s Chi-squared independence tests between success of target selection and all independent variables. The overall error rates were 7.36% for relative pointing and 4.30% for ray-casting ($\chi2$ = 63.2*). Thus for the selection time and error analysis, we separated the results between the two pointing systems because of their differences in accuracy, latency and induced fatigue.

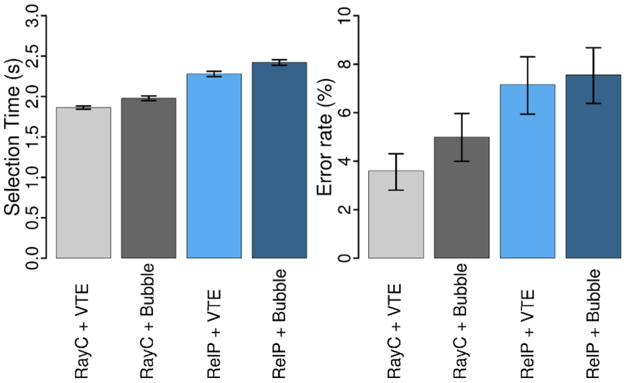

Figure 6: Mean Selection Time and Mean Error Rate per (Pointing System x Technique), with 95% confidence intervals.

Table 1: Fitts’ law coefficients (a for intercept in s and b for the slope in s/bit) for each (Pointing System + Technique)

	a	**b**	**Adj. R^2**
RayC + VTE	.164	.566	.94
RayC + Bubble	.253	.575	.89
RelP + VTE	-.425	.902	.94
RelP + Bubble	-.200	.874	.97

6.4.1 *Performance Results for Ray-Casting Pointing*

The overall mean selection times were 1.87 s for VTE and 1.98 s for Bubble Ray (Figure 6 left). Paired-t tests showed that this difference was significant (t = 6.528*), thus confirming H1. A repeated analysis of variance of variables (*Technique* x *Density* x *IDe* x *EW/W* x *ID* x *Amplitude*) on selection time showed significant effects for variables: *Density* ($F_{3,7467}$ = 343*), *IDe* ($F_{11,7467}$ = 294*), *EW/W* ($F_{9,7467}$ = 179*), *ID* ($F_{2,7467}$ = 162*) and *Technique* ($F_{1,7467}$ = 42.6*) but not for *Amplitude* (p = 0.13). No interaction effect with *Technique* was found. However and as expected, effects on selection time were significant for *Density* x *ID* ($F_{6,7467}$ = 22.9*) and *Density* x *Amplitude* ($F_{3,7467}$ = 10.7*). Although the pointing task is performed at a distance, the selection time can be accurately modeled as a linear function of *IDe* using Fitts’ law from a regression model (Table 1).

The error rates were 3.61% for VTE and 4.99% for Bubble Ray (Figure 6 right). This difference was significant ($\chi2$ = 8.44, $p < 0.01$), confirming H1. The variables *IDe* ($\chi2$ = 229*), *EW/W* ($\chi2$ = 173*), *Density* ($\chi2$ = 123*), *ID* ($\chi2$ = 63.3*) and *Amplitude* ($\chi2$ = 6.57, p = 0.01) had a significant effect on error rate.

6.4.2 *Performance Results for Relative Pointing*

The overall mean selection times were 2.28 s for VTE and 2.42 s for Bubble Ray (Figure 6 left). Paired-t tests showed that this difference was significant (t = 5.814*), thus confirming H1. A repeated analysis of variance of variables (*Technique* x *Density* x *IDe* x *EW/W* x *ID* x *Amplitude*) on selection time showed a

[1] * denotes $p < 0.001$

significant effect for all tested variables: *Density* ($F_{3,7467}$ = 434*), *ID* ($F_{2,7467}$ = 398*), *IDe* ($F_{11,7467}$ = 233*), *EW/W* ($F_{9,7467}$ = 219*), *Amplitude* ($F_{1,7467}$ = 37.6*) and *Technique* ($F_{1,7467}$ = 33.8*). Significant effects on selection time were found for the interactions *Technique* x *IDe* ($F_{11,7467}$ = 2.68, p = 0.002), *Technique* x *EW/W* ($F_{9,7467}$ = 2.23, p = 0.02), *Density* x *ID* ($F_{6,7467}$ = 17.5*) and *Density* x *Amplitude* ($F_{3,7467}$ = 4.38*). Based on the *IDe* effect, Figure 7 shows the mean index of performance: Below the threshold of *IDe* 3.5 (corresponding to an approximated effective target width *EW* of 63.1 pixels), VTE significantly outperforms Bubble Ray. Above the threshold of *IDe* 3.5, the two techniques are equivalent.

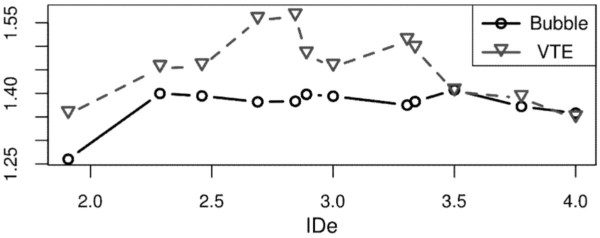

Figure 7: Relative Pointing: Mean Index of Performance (IP) as a function of the Effective Index of Difficulty (IDe) for relative pointing. IP = MT/IDe, MT being the movement time.

The error rates were 7.16% for VTE and 7.56% for Bubble Ray (Figure 6 right). However, this difference was not significant (χ2 = .384, p = 0.54). The variables *IDe* (χ2 = 475*), *EW/W* (χ2 = 329*), *Density* (χ2 = 255*), *ID* (χ2 = 142*) and *Amplitude* (χ2 = 32.1*) had a significant effect on error rate. While participants globally respected the target error rate (4% of selections) with ray-casting pointing, this was not the case for relative pointing. Moreover the difference in error rates below (4.93%) and above (19.3%) the threshold *IDe* 3.5 of Figure 7 was significant (χ2 = 313*), thus explaining the effect of *IDe* and *EW/W* on error rate. Such a threshold for movement speed and error rate was not observed for ray-casting pointing and we estimate that this is the limit of precision of this relative pointing system based on Asus Xtion Pro Live device.

6.5 Results: User Preferences

Questionnaires confirmed H2 about the perceived usability of the assistance techniques. Among 13 subjects, 6 subjects preferred RayC+VTE, 3 RayC+Bubble, 2 RelP+VTE and 2 ranked them identically. We asked participants to choose the most intuitive technique independently from the pointing system: 10 subjects selected VTE as the most intuitive one, 1 subject selected the Bubble Ray and 2 did not answer this question. The most common reasons for that choice were (1) the visibility and stability of the effective shapes (4 occurrences) and (2) that it was less disturbing than the Bubble Ray (3 occurrences). Finally, the SUS scores also confirmed H2. RayC+VTE obtained an average SUS score of 77.1, RayC+Bubble 74.6, RelP+VTE 71.5 and RelP+Bubble 66.5, which is a quite low mean score.

7. DISCUSSION

VTE outperformed the two techniques with dynamic feedback (target expansion in experiment 1, cursor expansion-Bubble Ray in experiment 2). This supports the rationale of the VTE design based on the two phases of a pointing task and on the reduction of the cognitive load.

Ballistic motion of the pointing task: By consciously knowing the exact activation zone, the users can plan a more opportunistic movement, which is adapted to the effective target shape. Indeed the nervous impulse starting from the central neuromotor system already programs an arm stiffness adapted to the effective target shape in both normal and tangential directions. Even if such a ballistic movement would not be necessarily faster, the confidence in its success is increased. The human neuromotor system has been proven to find the best speed/accuracy trade-off [19]: VTE offers a way to exploit these capabilities by displaying all the relevant information about the target.

Corrective phase of the pointing task: First, a potential (not measured in the experiments) benefit is to skip the corrective phase of the movement and to save 45% of the movement time [21]. Second, with helpful planning possibilities and a high confidence in the effective target shape, the users can optimize their corrective sub-movements. They can consciously and confidently exploit borders and corners of all expanded target shapes. This is maybe more difficult in the case of an indirect representation of the Voronoi cells as in Bubble Ray [20]. This could also lead to an early validation trigger, before stopping the motion. Moreover, we think that VTE offers the users a better feeling of control, which is key in usability.

Cognitive load: First, Voronoi cells are simple and regular shapes, easy to perceive and understand. Every point of the space is contained in the polygon surrounding its nearest target. The user can easily understand this characteristic: "wherever I click, the selected target is the closest one from the cursor". Second, with VTE the user perceives the effective boundaries of each target simultaneously. But since forgetting is a necessary mechanism in cognition, the cognitive system will then filter the needed information when focusing on a target. The rest is therefore forgotten. Finally with a static representation of the critical information, we shift the cognitive load to the perceptual system. This shift is particularly interesting in the critical context of surgery: the main task of the surgeon is complex and highly dynamic (subject to change according to the context) [6]. Moreover the major risks and the responsibilities inherent to surgery cause an important emotional load to the surgeons [6] and their staff and amplify cognitive biases. It is therefore important to minimize the surgeon's cognitive load during interaction with the augmented surgery system.

Further studies on VTE graphical appearance could improve its acceptability by users and its integration with existing interfaces. Inverted colors and transparency for displaying Voronoi edges are the two techniques that do not suppress any information on screen, while representing the full tessellation. Different Voronoi tessellations (as in [3] for the case of clusters of targets) can be studied including non euclidean or weighted Voronoi tessellations. However, the simplicity of polygonal Voronoi cells could be lost. A different approach could be to adapt the positions of the widgets (i.e. targets) on screen in order to obtain a centroidal Voronoi tessellation.

8. CONCLUSION AND FUTURE WORK

In this paper we presented VTE, a static feedforward target assistance technique, which distributes by Voronoi tessellation the available space of an interface among the selectable targets. We defined VTE as a plug-and-play technique that helps the target selection phases. Addressing the problem of the generated visual information added to the GUI, we suggested two main solutions to display Voronoi edges: inverted colors and transparency. Our two user studies support our claims that a static feedforward is more efficient in speed and error rate than a dynamic feedback for target expansion. Applied to an augmented surgery application, the first feedback on VTE by professionals of augmented surgery were positive. Our next step is to further experiment VTE in the context of the Aesculap's navigation system (Figure 1). As future work we plan to focus on the selection phase that follows the pointing phase. We also plan to study static and dynamic target expansion assistance for the case of classic pointing devices (e.g. mouse or touchscreen) in particular to understand if the gain is as important as for distant pointing.

9. ACKNOWLEDGMENTS

This work has been partially supported by the LabEx PERSYVAL-Lab (ANR-11-LABX-0025-01). We thank Thomas Vincent for his help with statistics, Renaud Blanch for his help in defining the experimental protocol, the colleagues of Aesculap for insightful comments on the technique in the context of augmented surgery and all the participants of the experiments. Special thanks to George Serghiou for proof-reading the final version.

10. REFERENCES

[1] Balakrishnan, R. 2004. ''Beating'' Fitts' law: virtual enhancements for pointing facilitation. *Int. J. of Human-Computer Studies*, 61 (2004), 857-874.

[2] Bateman, S., Mandryk, R.L., Gutwin, C., and Xiao, R. 2013. Analysis and comparison of target assistance techniques for relative ray-cast pointing. *Int. J. of Human-Computer Studies,* 71 (2013), 511-532.

[3] Baudisch, P., Zotov, A., Cutrell, E., and Hinckley, K. 2008. Starburst: a Target Expansion Algorithm for Non-Uniform Target Distributions. In *Proc. of the International Conference on Advanced Visual Interfaces* (2008). AVI'08. ACM, New York, NY, 129-137.

[4] Blanch, R., and Ortega, M. 2011. Benchmarking Pointing Techniques with Distractors: Adding a Density Factor to Fitts' Pointing Paradigm. In *Proc. of the SIGCHI Conference on Human Factors in Computing Systems* (2011). CHI'11. ACM, New York, NY, 1629-1638.

[5] Brooke, J. 1996. SUS: A Quick and Dirty Usability Scale. In *Usability Evaluation in Industry*. Taylor & Francis, 189-194.

[6] Cassell, J. 1987. On control, certitude, and the "paranoia" of surgeons. *Culture, Medicine and Psychiatry*, 11 (1987), 229-249.

[7] Grossman, T., and Balakrishnan, R. 2005. The Bubble Cursor: Enhancing Target Acquisition by Dynamic Resizing of the Cursor's Activation Area. In *Proc. of the SIGCHI Conference on Human Factors in Computing Systems* (2005). CHI'05. ACM, New York, NY, 281-290.

[8] Ebert, L.C., Hatch, G., Thali, M.J., and Ross, S. 2013. Invisible touch - Control of a DICOM viewer with finger gestures using the Kinect depth camera. *Journal of Forensic Radiology and Imaging,* 1, 1 (Jan. 2013), 10-14.

[9] Fortune, S. 1987. A sweepline algorithm for Voronoi diagrams. *Algorithmica* 2, 1 (1987), 153-174.

[10] Jota, R., Nacenta, M.A., Jorge, J.A., Carpendale, S., and Greenberg S. 2010. A comparison of ray pointing techniques for very large displays. In *Proc. of Graphics Interface* (2010). GI'10. 269-276.

[11] Kopper, R., Bowman, D.A., Silva, M.G., and McMahan, R.P. 2010. A human motor behavior model for distal pointing tasks. *Int. J. of Human-Computer Studies*, 68, 10 (2010), 603-615.

[12] Mandryk, R.L., and Gutwin, C. 2008. Perceptibility and Utility of Sticky targets. In *Proc. of Graphics Interface* (2008). GI'10, 65-72.

[13] McGuffin, M., and Balakrishnan, R. 2002. Acquisition of Expanding Targets. In *Proc. of the SIGCHI Conference on Human Factors in Computing Systems* (2002). CHI'02. ACM, New York, NY, 57-64.

[14] Meyer, D.E., Abrams, R.A., Kornblum, S., Charles E. Wright, C.E., and Smith, J.E.K. 1988. Optimality in human motor performance: Ideal control of rapid aimed movements. *Psychological Review,* 95, 3 (1988), 340-370.

[15] Myers, B., Bhatnagar, R., Nichols, J., Peck, C.H., Kong, D., Miller, R., and Long, A.C. 2002. Interacting at a Distance: Measuring the Performance of Laser Pointers and Other Devices. In *Proc. of the SIGCHI Conference on Human Factors in Computing Systems* (2002). CHI'02. ACM, New York, NY, 33-40.

[16] O'Hara, K., et al. 2014. Touchless Interaction Surgery. *Communications of the ACM*, 57, 1 (2014), 70-77.

[17] Parker, J.K., Mandryk, R.L., Nunes, M.N., and Inkpen, K.M. 2005. TractorBeam Selection Aids: Improving Target Acquisition for Pointing Input on Tabletop Displays. In Proc. of the IFIP International Conference on Human Computer Interaction (2005). INTERACT'05. Springer, 80–93.

[18] Tse, E., Hancock, M., and Greenberg, S. 2007. Speech-Filtered Bubble Ray: Improving Target Acquisition on Display Walls. In *Proc. of the International Conference on Multimodal Interaction* (2007). ICMI'07. ACM, New York, NY, 307-314.

[19] Van Galen, G.P., and De Jong, W. P. 1995. Fitts' law as the outcome of a dynamic noise filtering model of motor control. *Human Movement Science*, 14 (1995), 539-572.

[20] Vogel, D., and Balakrishnan, R.2005. Distant Freehand Pointing and Clicking on Very Large, High Resolution Displays. In *Proc. of the Annual ACM Symposium on User Interface Software and Technology* (2005). UIST'05. ACM, New York, NY, 33-42.

[21] Zhai, S., Conversy, S., Beaudouin-Lafon, M., and Guiard, Y. 2003.Human on-line response to target expansion. In *Proc. of the SIGCHI Conference on Human Factors in Computing Systems* (2003). CHI'03. ACM, New York, NY, 177-184.