Investigating Visual Feedforward for Target Expansion Techniques

Maxime Guillon[1,2], François Leitner[2], Laurence Nigay[1]
[1]Univ. Grenoble Alpes, LIG, CNRS
F-38000 Grenoble, France
{maxime.guillon, laurence.nigay}@imag.fr
[2]Aesculap SAS
F-38130 Échirolles, France
{maxime.guillon, francois.leitner}@bbraun.com

ABSTRACT
Target expansion techniques facilitate the pointing task by enlarging the effective sizes of targets. When the target expansion is applied to both the motor and visual spaces, the visual feedforward mechanism is key: Indeed it provides a visual aid to the user on the effective expanded targets prior to the execution or completion of the pointing task, enabling the user to take full advantage of the target expansion technique. Focusing on feedforward mechanisms, we introduce a design space that allows us to describe, classify and design target expansion techniques. To do so we first introduce and characterize the concept of *atomic feedforward mechanism* along three design axes. We then describe a target expansion technique as a combination of atomic feedforward mechanisms using a matrix-based notation. We provide an analytical exploration of the design space by classifying existing techniques and by designing six new techniques. We also provide a first experimental exploration of the design space in the context of distant pointing. The experimental protocol includes an innovative target layout for handling non-centroidal target expansion. The results show that feedforward dynamicity increases movement time and decreases subjective usability, while explicit expansion observability efficiently supports error prevention for distant pointing.

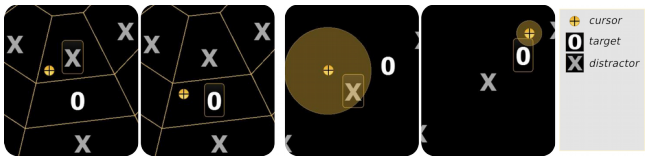
Author Keywords
Target expansion; feedforward; pointing; distant pointing.

ACM Classification Keywords
H.5.2. Information interfaces and presentation: User Interfaces

INTRODUCTION
Pointing to targets is an elementary task universally present in graphical user interfaces (GUI). Thus researchers have proposed many targeting assistance techniques [1, 3, 4, 6, 8, 10, 11, 12, 15, 16, 17, 18] to optimize pointing. Pointing can be accurately modeled using Fitts' law [4, 9, 13]. 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). We focus on targeting assistance techniques that enlarge the effective sizes of targets (W) both in the motor and visual spaces. Such target expansion techniques rely on a partitioning of the space. An example of space decomposition is the Voronoi tessellation [3, 7, 8, 10, 11] that maximizes the use of empty space and is unambiguous since only one target is contained in each Voronoi cell. Several target expansion techniques implement a target expansion algorithm based on the Voronoi tessellation [3, 6, 8, 10, 11, 15]: The expanded targets then correspond to the Voronoi cells and the user can point anywhere inside the target Voronoi cell instead of pointing at the target. For target expansion both in the motor and visual spaces, the design challenge of target expansion techniques is to make the motor target expansion visually observable through a feedforward mechanism. The visual feedforward mechanism is key since it provides a visual aid to the user on the effective expanded targets prior to the execution or completion of the pointing task, enabling the user to take full advantage of the target expansion technique. Recent studies [10, 16] confirm the key role that the feedforward mechanism plays in the performance of a target expansion technique.

Focusing on feedforward mechanisms for target expansion, this paper proposes a design space for describing, classifying and designing target expansion techniques. Three design axes characterize the concept of *atomic feedforward mechanism*, an elementary unit that can be combined for designing a target expansion technique. We present a matrix-based notation for describing target expansion techniques as a combination of *atomic feedforward mechanisms*. We provide an analytical exploration of the design space by classifying existing target expansion techniques and by designing six new target expansion techniques that involve *atomic and combined feedforward mechanisms*. We also provide a first experimental exploration of the design space. To do so the experimental protocol includes an innovative target layout that can handle non-centroidal target expansion for controlling various shifts between a target's centroid and its expanded target's centroid.

CHI 2015, April 18 - 23 2015, Seoul, Republic of Korea
Copyright 2015 ACM 978-1-4503-3145-6/15/04…$15.00
http://dx.doi.org/10.1145/2702123.2702375



(a) (b)

Figure 1. (a) VTE: The visual feedforward is the display of all the Voronoi cells. The goal target ("0") can be selected when the cursor is within the Voronoi cell of the "0" target. (b) Bubble Cursor: The visual feedforward is a bubble with the cursor at the center. The goal target ("0") can be selected when the bubble reaches the "0" target.

We note that the experimental exploration of the design space is conducted in the context of distant pointing because: (1) Techniques enhancing target selection by expanding the targets demonstrated their efficiency in distant pointing for various contexts, including virtual 3D environments [18], large displays [17] and operating rooms [10]. (2) Our application domain is augmented surgery, where efficient distant pointing techniques are important when considering the cognitive workload of the surgeons. Moreover the GUIs involved in augmented surgery are particularly suited to target expansion techniques. Indeed, for target enlargement, dividing the motor space into areas is beneficial only if the motor space contains empty space not used for interaction (the effective sizes of the targets being extended into the adjacent empty space). In the augmented surgery GUIs that we study, classical desktop-based functions (e.g., selection of a group of items, of an empty space) are not supported so that, the entire motor space can be altered without any functionality loss.

MOTIVATION: VARIETY OF FEEDFORWARD MECHANISMS FOR TARGET EXPANSION TECHNIQUES

Existing target expansion techniques exhibit different characteristics of visual feedforward that motivated us to establish a design space. For illustrating this variety, we consider two radically different target expansion techniques that employ the same target expansion algorithm, based on a Voronoi tessellation for partitioning the space into Voronoi cells (i.e. the expanded targets). First the Voronoi-based Target Expansion (VTE) technique [10] provides an explicit static feedforward mechanism by displaying the entire space partitioning (i.e. the Voronoi tessellation) before and during the pointing task (Figure 1-a). Second, Bubble Cursor [8] provides a dynamic feedforward by visually augmenting the cursor: a round bubble that expands to constantly reach the closest target (Figure 1-b). The bubble feedforward implicitly conveys the target expansion algorithm to the user, since the target's expanded shape (i.e. the target's Voronoi cell) is not displayed.

DESIGN SPACE

Since a target expansion technique can implement several visual feedforward mechanisms, we introduce the concept of *atomic feedforward mechanism* to designate an elementary mechanism that can be combined within a technique.

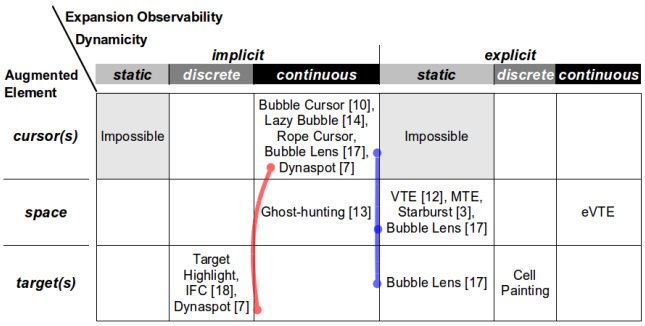


Figure 2. Atomic feedforward mechanisms characterized along three axes. Target expansion techniques are located within the design space. Empty squares in the design space highlight unexplored design possibilities.

This section first examines design axes for characterizing an atomic feedforward mechanism, independently from its visual form (e.g., color, shape) and from the underlying expansion algorithm. We then describe an entire target expansion technique using a matrix notation by considering the combination of atomic feedforward mechanisms provided during a pointing movement.

Atomic Feedforward Mechanisms: Three Axes

Figure 2 presents three axes for an atomic feedforward mechanism.

Axis Dynamicity: As defined in [2, 10, 19], a feedforward mechanism can be static or dynamic. Some of the dynamic feedforward mechanisms are discrete, while others are continuous [2, 19]. While continuous feedforward mechanisms constantly change when the cursor moves, discrete ones intermittently change at discrete points in time (e.g., when the hovered target changes). For example Bubble Cursor [8] is a case of continuous feedforward, the bubble being updated continuously. VTE [10] is static, permanently displaying the complete Voronoi tessellation.

Axis Expansion Observability: The *Expansion Observability* axis characterizes the visual observability of the motor target expansion. A feedforward mechanism makes observable the current target cell, the cells of several targets (for instance the ones close to the cursor) or the space partitioning (i.e. all the cells). Existing mechanisms provide spatial information. No symbolic approach (e.g., letters) has been studied for target expansion feedforward since the pointing task is spatial. We further describe an atomic feedforward mechanism by distinguishing explicit or implicit expansion observability.

- Explicit feedforward: When the cell of a target or the cells of several targets are directly observable (with different visual clues including lines or colors), the feedforward mechanism is explicit: This is a spatial information that informs the user where to click. By showing one screenshot of the technique, an observer can directly observe the target's cell (Figure 1-a: VTE).

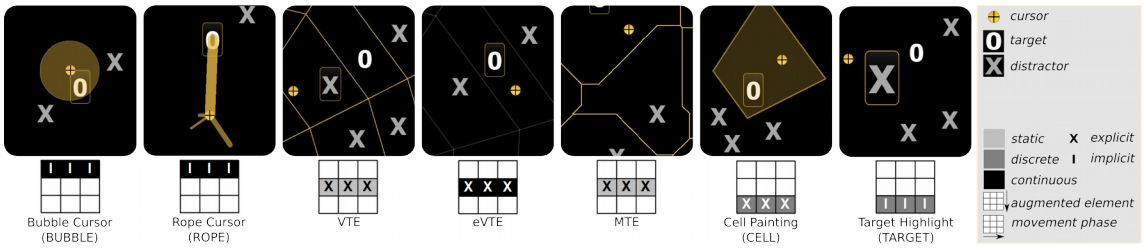


Figure 3. Developed and experimentally compared target expansion techniques involving one type of atomic feedforward mechanism (for each technique: a screenshot and its matrix-based representation).

- Implicit feedforward: If the user has to use indirect means (e.g., cursor responses to various positions) in order to estimate the target's cell, then the feedforward is implicit. An implicit feedforward mechanism provides spatial clues relative to its previous state: The mechanism is therefore dynamic (discrete or continuous) and informs the user when to click. For example Bubble Cursor is implicit since the targets' cells are not displayed: When the dynamic visual bubble reaches the target, the user can select it by clicking (Figure 1-b: Bubble Cursor).

Axis Augmented Element: Various visual clues (e.g., color, shape, size) characterize the feedforward visual form. In addition for target expansion, a discriminating characteristic of the visual form is the *element* of the pointing task that is visually *augmented*: the cursor, the space and the target. Since the cursor is moving, cursor-based feedforward is by definition dynamic. A space-based visual feedfoward, partly or fully, modifies the available empty space while a target-based feedforward modifies the visual appearance of a target. For instance with VTE, the visual appearance of the targets is not modified: It is a space-based feedforward mechanism, as opposed to a target-based mechanism that doubles the size of the current selected target.

The resulting three-axes design space (Figure 2) defines 16 types of atomic visual feedforward mechanisms that serve as a basis for the design of target expansion techniques.

Combined Atomic Feedforward Mechanisms

Target expansion techniques combine different types of atomic feedforward mechanisms. To characterize the combination of feedforward mechanisms, we consider *when* the mechanisms are used according to the three phases of the pointing movements that the Optimized Initial Impulse Model defined [14]: (1) The preliminary phase precedes any pointing movement. It includes the time of target spotting and anticipation and neuromotor programming of the movement. (2) The ballistic phase designates the first reaching movement which is fast and imprecise. (3) The corrective phase only happens if the ballistic movement does not land on the target. Indeed because of neuromotor noise, corrective small and precise submovements are often needed for performing difficult pointing tasks.

If a technique uses several atomic feedforward mechanisms simultaneously (i.e. during the same phase), the mechanisms are combined in a *parallel* way. *Sequential* combination then describes feedforward mechanisms used during consecutive phases. Atomic feedforward mechanisms are therefore described at the granularity of the movement phase [14].

Matrix-based Notation for Describing a Technique

An atomic mechanism corresponds to one and only one square within the classification scheme of Figure 2. Contrastingly, a target expansion technique combining different feedforward mechanisms corresponds to different types (and therefore squares) in the classification scheme of Figure 2 (e.g., Dynaspot and Bubble Lens described in the following section). The comparison of techniques is then not straightforward. We therefore define a compact visual notation for describing an entire target expansion technique based on the characteristics of the involved feedforward mechanisms and their combination scheme (matrices of Figures 3 and 4). A 3x3 matrix represents a target expansion technique. Columns represent the 3 movement phases along a horizontal time axis, in chronological order from left to right. Lines correspond to the *augmented element* axis of Figure 2: from the *cursor* (top) to the *target* (bottom) through the *space* (middle). Each cell of the matrix then represents one type of augmented element during one movement phase. A blackness code represents the *dynamicity* axis: light gray for *static*, dark gray for dynamic *discrete* and black for dynamic *continuous*. Finally, a "X" in the cell is used for an *explicit* feedforward along the *expansion observability* axis (e.g., VTE matrix in Figure 3) and a "I" for an *implicit* one (e.g., BUBBLE matrix in Figure 3).

ANALYTICAL EXPLORATION OF THE DESIGN SPACE: EXISTING TARGET EXPANSION TECHNIQUES

Figure 2 classifies target expansion techniques from the literature according to their underlying feedforward mechanisms. Figures 3 and 4 represent the described target expansion techniques using our matrix-based notation.

Cursor-Based Mechanisms: Bubble Cursor, Lazy Bubble and Cone Cursor

The Bubble Cursor [8] feedforward mechanism illustrates well the intensively studied type of *implicit continuous* augmentation of the *cursor* (BUBBLE matrix in Figure 3). By limiting its size to reach only one target [8], the bubble potentially shows which target is on the point of being

designated: This de facto constitutes an error prevention mechanism. Lazy Bubble and Cone Cursor [12] are variations along the continuum from a traditional point cursor to Bubble Cursor. These techniques implement *implicit continuous* feedforward by visually augmenting the *cursor*: Their users know when to click to select a target rather than where to click.

Space-Based Mechanisms: Voronoi-based Target Expansion (VTE), Starburst and Ghost-hunting

As previously described, VTE [10] displays the entire Voronoi diagram once, before the movement begins. VTE illustrates a type of feedforward that make the target expansion mechanism explicitly observable (VTE matrix in Figure 3). Starburst [3] proposes a new partitioning of the space, especially suited to clusters of targets. Like VTE, its feedforward mechanism is an *explicit static* augmentation of the *space*. Ghost-hunting [11] creates an avatar for each target (a *ghost*) which constantly indicates the optimal trajectory to reach the target depending on the cursor's position. Its expansion algorithm is also based on a Voronoi tessellation. This technique uses an *implicit continuous* augmentation of the *space* (Ghost-hunting matrix in Figure 4), even if the avatars could also modify the targets' appearance by intersecting them.

Target-Based Mechanisms: Visual Target Expansions and Implicit Fan Cursor (IFC)

Feedforward mechanisms that highlight the target have been intensively studied [13, 20]. Amongst them, Cell Painting (Figure 3) expands the target to its cell by painting the full expanded target shape with a semi-transparent color. The feedforward mechanism is therefore an *explicit discrete* augmentation of the *target* (CELL matrix in Figure 3). When the target is scaled, namely TARGET in Figure 3, or highlighted by changing its color [13, 20], the feedforward mechanism becomes an *implicit discrete target* augmentation (TARGET matrix in Figure 3). In terms of *expansion observability*, it only makes observable the current target and not the cell (in contrast to Cell Painting).

Implicit Fan Cursor (IFC) [16] was originally designed as a *continuous cursor*-based feedforward mechanism. With this first design, namely the Fan Cursor technique, a fan-shaped cursor's activation area dynamically adjusts its spanning angle and orientation to minimize the cursor's movements. In [16] Fan Cursor is compared with a design alternative, namely Implicit Fan Cursor (IFC), based on highlighting the target: Instead of the cursor fan-shaped activation area, the current target is highlighted and slightly expanded. Results show that the *discrete* augmentation of the *target* (IFC) (IFC matrix in Figure 4) is more efficient than the initial Fan Cursor technique.

Combined Mechanisms: Dynaspot and Bubble Lens

Dynaspot [6] extends Bubble Cursor [8] with an implicit mode-switching enabling desktop-based interaction. Above a cursor speed threshold of 150 pixels/s, targets are

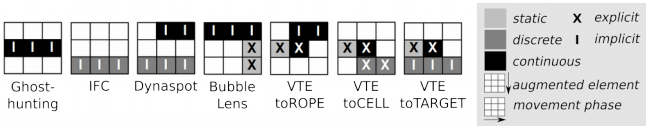


Figure 4. Matrix-based representation of existing target expansion techniques and of designed techniques combining different types of atomic feedforward.

expanded according to a Voronoi tessellation. The bubble, contrary to Bubble Cursor, grows to a predefined maximum width and shrinks back when the cursor slows down under the threshold. The expansion phase nevertheless lasts for about 300 ms after the cursor speed falls under the threshold. During the expansion phase, the closest target is highlighted. Thus this technique combines two complementary feedforward mechanisms. First, the bubble, an *implicit continuous cursor*-based feedforward mechanism, informs about the target expansion activation state. Second the target highlight is an *implicit discrete target*-based feedforward mechanism that indicates the designated target (Dynaspot matrix in Figure 4).

Bubble Lens [15] behaves like Bubble Cursor [8], with the addition of a local magnification when the targets are small and dense. Thanks to kinematic triggering, magnification is triggered on the downward slope of the first corrective submovement [14, 15] in a smoothed velocity profile. Thus, the magnification happens only during the corrective phase of a pointing movement [14]. During the magnification phase, the targets and the space are expanded in a round area of interest (a different expansion algorithm than the one based on a Voronoi diagram). Since the targets and the space are represented in their expanded forms, independently from the cursor's movements, the magnification combines two *explicit static* feedforward mechanisms: a visual *space* expansion and a visual *target* expansion. To sum up, since Bubble Cursor is still active during the magnification, two expansion algorithms and three feedforward mechanisms are therefore active simultaneously (Bubble Lens matrix in Figure 4). A matrix having several filled cells in a column (Figure 4) represents the parallel combination of atomic feedforward mechanisms supported by Dynaspot and Bubble Lens. We note that the matrix-based notation allows us to describe the parallel combinations of mechanisms based on different *augmented elements* only.

ANALYTICAL EXPLORATION OF THE DESIGN SPACE: DESIGNING NEW TARGET EXPANSION TECHNIQUES

Exploiting the design space, we designed six new target expansion techniques. While BUBBLE is well established, VTE outperformed it [11]. So we started from these two techniques whose characteristics are different along the design axes. Based on BUBBLE and the *expansion observability* axis we designed ROPE. Based on VTE and the *dynamicity* axis we designed eVTE and MTE. Finally studying feedforward for each movement phase prompted us to combine them.

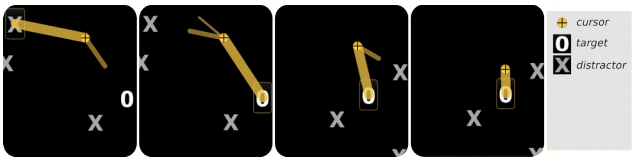


Figure 5. Rope Cursor: Dynamic main/mini ropes indicating the current target and the neighboring targets.

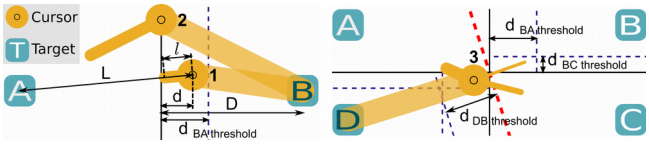


Figure 6. Rope Cursor: Parameters for computing the length and width of the main/mini ropes.

Technique 1: Rope Cursor

Our design goal was to increase the *expansion observability* of Bubble Cursor while minimizing the visual disturbance of the *dynamicity* of the bubble area. To do so we use lines instead of a bubble area as shown in Figure 5: A main line (i.e. the rope) joins the cursor to the closest target and supplementary smaller lines (i.e. the mini-ropes) dynamically indicate the proximity of neighboring targets.

When the cursor is closer to a border of the hovered Voronoi cell than a distance threshold (Figure 6: cases 1 and 2, where $d < d_{BA\ threshold}$), a mini rope appears on the cursor. We arbitrarily define this distance threshold at one third of the distance from the designated target to the corresponding border (Figure 6: $d_{BA\ threshold} = D/3$). Thus, distance thresholds can be different in a cell, as shown in Figure 6 ($d_{BA\ threshold} \neq d_{BC\ threshold}$). When a target shares only one corner with the hovered cell (Figure 6: case 3, targets D and B), we define a distance threshold equal to one third of the distance from the designated target to the corresponding corner (Figure 6: the widest dashed line represents the bisection of a segment DB). The mini-rope is oriented towards the direction of the target whose cell's border is close (distance threshold) to the cursor. Thus, the mini-ropes directly designate the approaching neighboring targets. For instance in Figure 6 cases 1 and 2, the mini-rope indicates target A. While the cursor is approaching the border, the mini rope linearly grows within a range of *0* to *L/2* in Figure 6. Thus, when the cursor is about to cross the border (Figure 6: case 2), the mini-rope is as long as half the distance separating the cursor from the designated neighboring target, making the border crossing transition smooth. This mini-rope behavior represents the remaining distance to the neighboring cell (Figure 6: *d*). Indeed, at any time the ratio between *l* and *L/2* is equal to the ratio of $d_{BA\ threshold} - d$ and $d_{BA\ threshold}$. Complementary to its length, the mini-rope's width indicates the proximity order of neighboring targets: The wider the mini-rope is, the closer the cursor is to the target. Each mini-rope's width is equal to the immediately wider (mini-)rope's width divided by 2 (Figure 3, 5 and 6). Rope Cursor is a new technique based on the *cursor* that is *implicit* and *continuous* (ROPE matrix in Figure 3). Rope Cursor provides a better *expansion observability* than Bubble Cursor, while still being *implicit.* Indeed the 3 mini-ropes (but more is possible) show the proximity of 3 neighboring targets simultaneously (Figure 3 and 6), while the bubble is constrained to adjust its round geometry to only two closest targets [8] (Figure 1-b: Bubble Cursor).

Technique 2: Erasable Voronoi-based Target Expansion (eVTE)

The *static* display of the Voronoi diagram provided by VTE may disturb the visual perception of the underlying GUI and hide useful information. To address this issue, we explored the *dynamicity* axis. To make VTE *dynamic*, the transparency of the Voronoi diagram is coupled to the cursor's speed. The diagram's transparency follows a sigmoidal function of the cursor's speed. Thus, at high speed the Voronoi diagram disappears in a continuous and progressive way to reveal the GUI. The technique, namely Erasable VTE (eVTE), also supports the possibility of temporarily erasing the diagram by shaking the cursor. eVTE is a new technique that is (*explicit*, *continuous*, *space*-based) (eVTE matrix in Figure 3). eVTE then fills an empty square in Figure 2: a design possibility so far unexplored.

Technique 3: Manhattan Target Expansion (MTE)

MTE is based on a partitioning of the space among the targets according to a Voronoi diagram computed in Manhattan distance (Figure 7). Contrary to the Euclidean distance (e.g., VTE), the Manhattan distance between two points (also called rectilinear distance) is the sum of the absolute differences of the Cartesian coordinates of the two points. Voronoi diagrams in such a geometry include only vertical, horizontal and 45°-oriented lines (MTE in Figure 3 and Figure 7). The resulting partitioning on screen could then be more aesthetic and consistent with GUI. In using this innovative space partitioning, we propose a technique with the same characteristics as VTE according to our design axes (*explicit*, *static*, *space*-based) (MTE matrix in Figure 3) but with a different expansion algorithm.

Techniques 4-5-6: Combined Feedforward Mechanisms

Our design goal for these three techniques addressed the same issue as for eVTE: the VTE's Voronoi diagram permanent display visual overload. As opposed to the design of eVTE, this time we explored the *combination* of feedforward mechanisms. Since VTE enhances the movement anticipation [10], we kept VTE during the preliminary phase. We then explored the combinations of VTE with three different feedforward mechanisms for the ballistic and corrective phases: ROPE (*implicit*, *continuous*, *cursor*-based), CELL (*explicit*, *discrete*, *target*-based) and TARGET (*implicit*, *discrete*, *target*-based) (Figure 3). The resulting techniques are respectively called VTEtoROPE, VTEtoCELL and VTEtoTARGET and are represented using our matrix-based notation in Figure 4. First, like Dynaspot

[6], the feedforward mechanism switching occurs at the early ballistic phase, as soon as the anticipation time is over: The VTE's Voronoi diagram progressively disappears with a visual fade-out effect, its transparency quadratically increasing from full opacity to full transparency in 300 ms. Thus a dynamic continuous version of VTE is still present during the ballistic phase for 300 ms (as represented by the black cell in the matrices of Figure 4), in parallel with another feedforward mechanism (ROPE, CELL or TARGET). Second, while TARGET is active during the entire movement (three phases) in VTEtoTARGET, we explored another type of combination with VTEtoROPE and VTEtoCELL. Indeed ROPE and CELL start to be active at the early ballistic phase and remain active until the end of the movement (two phases only).

Summary: As illustrated by the three techniques, VTEtoROPE, VTEtoCELL and VTEtoTARGET, the combination of feedforward mechanisms defines numerous possibilities. The resulting design space defined by the three axes of an atomic feedforward mechanism and by the combination of atomic mechanisms therefore opens a vast set of design possibilities. We started the analytical exploration of the design space by classifying existing target expansion techniques and by designing six new ones. In the rest of the paper we embark on its experimental exploration.

EXPERIMENTAL EXPLORATION OF THE DESIGN SPACE

The goals of the experiment were to lead a first experimental exploration of the design space and to evaluate the 6 new techniques. In the experiment, in addition to comparing the 6 above techniques, we included Bubble Cursor and VTE as baselines. We also included TARGET as an atomic *target*-based feedforward. To sum up, we have therefore compared the following 9 techniques: VTE, BUBBLE, ROPE, eVTE, MTE, VTEtoROPE, VTEtoCELL, VTEtoTARGET and TARGET. We thus compared at least one technique per line of the design space (Figure 2) and one technique per column (except the column *implicit-static*). Our hypotheses are as follows:

- (H1) Within a cell, continuously changing feedforward produces noise and redundancy (Shannon's information theory) while the selected target does not change. We expect these ineffective stimuli to slow down the selection task.
- (H2) The atomic and combined techniques with explicit feedforward at movement start will be faster than the others. Users will optimize their ballistic gesture [14] according to a larger target size (Fitts' law).
- (H3) The atomic and combined techniques with explicit feedforward at movement end will be more precise. Users will obtain more powerful error anticipation means by evaluating the distance between the cursor and the target boundaries.

New Target Layout for Non-Centroidal Tessellations

This section presents the new target layout that we propose to be able to measure the shifting of the pointing movement endpoints induced by the visual feedforward mechanism.

Shifting of the pointing movement endpoints

The users optimize their pointing gestures depending on the perceived size of the target they are aiming at [14]. As a consequence, the target's centroid (its center of mass) is a good predictor of the movement endpoint, as demonstrated in [9] with targets of arbitrary shapes. By comparing the mean distance from the movement endpoints to the targets' centroids ($d_{toTarget}$) with the mean distance from the movement endpoints to the target cells' centroids (i.e. expanded targets' centroids) (d_{toCell}), we identify three cases for a feedforward mechanism:

- If $d_{toCell} < d_{toTarget}$, "cell-centered mechanism".
- If $d_{toCell} > d_{toTarget}$, "target-centered mechanism".
- If $d_{toCell} \approx d_{toTarget}$, "mixed-centered mechanism", as the pointing movement endpoints have no clear tendency.

The fact that a feedforward mechanism is cell-centered is experimental proof that the expanded target's shape has been taken into account by the users during the pointing movement. While we can expect that an explicit feedforward mechanism will provoke such a shifting from the target's centroid to the cell's centroid, the timing of such an explicit feedforward is of importance. Indeed, it determines the moment the users will perceive the expanded targets and therefore the remaining time for the optimization of the pointing movements.

For the case of Voronoi target expansion, existing target layouts make it difficult to observe the endpoints according to our three cases (i.e. cell, target and mixed). Indeed as in [6, 8, 15, 16, 17], a square Voronoi cell has a centroid matching that of the target: This is mathematically called a centroidal Voronoi tessellation [7]. The geometric layout described in [4] and used in [10] creates a pseudo-hexagonal Voronoi cell, whose centroid is very close to the corresponding target's centroid. In [1], cells' centroids are shifted from targets' centroids along a unique radial direction.

New target layout

We therefore propose a new target layout that shifts Voronoi cells' centroids from the targets' centroids along different controllable directions (Figure 7). The layout is fully reproducible and based on an ISO 9241-9 task [1]. The ISO task places a set of goal targets [1, 16] on a virtual circle. We add one distractor target at the center of this circle. We further define three constraints for the other distractors: (1) Distractors are placed on concentric circles. (2) Each concentric circle contains the same number of distractors. (3) The distractors of distinct circles are aligned along the radii. These constraints define trapezoidal Voronoi cells (Figure 7).

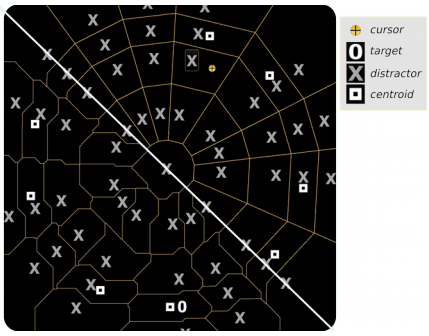


Figure 7. New target layout for non-centroidal Voronoi cells along with VTE (top) and MTE (bottom). The outlined squares are the centroids of the targets' cells (targets from the ISO 9241-9 task). The goal target (i.e. the white zero) is shifted from its Voronoi cell centroid in an anti-clockwise direction.

In order to create non-centroidal trapezoidal Voronoi cells, we then place the distractors (1) on circles with aperiodic diameters or (2) along radii in aperiodic directions (Figure 7). The cells' centroids are therefore shifted from the targets' centroids, respectively along (1) a radial direction or (2) a tangential direction (the outlined squares in Figure 7). In the context of an ISO 9241-9 task, targets are placed along periodic directions. To obtain distractors along radii in aperiodic directions, the distractors are added on the bisector of every two slices (i.e. angle between two radii) (Figure 7).

Apparatus and Participants

Our application domain is intra-operative distant interaction for computer-assisted surgery. By using orthopedic hardware and software certified material, we reproduced in part surgical conditions. Eighteen unpaid volunteers (6 women and 12 men) were all right-handed, ranging in age from 22 to 51 years. They stood in front of a table, 1.8 m away from the screen. They selected the target with a foot-switch three button medical pedal and controlled a cursor with an Aesculap virtual pointer. This ray-casting system [1] projects on the screen the axis of a metallic pointer on which an IR reflective rigid-body is clipped. An NDI Polaris IR tracker provides pointer positions and orientations. The software application was running on a 2.3 GHz Quadcore PC with Windows 7. We used a 24" LCD display at 1920 x 1200 resolution.

Implemented Techniques

Figure 3 gives screenshots of all the atomic feedforward mechanisms we implemented. The behavior of Bubble Cursor was the same as in [8, 10, 16]. The Rope Cursor main rope's and mini-ropes' widths were respectively 24, 12, 6 and 3 pixels. The transparency of the eVTE's diagram varied from full opacity to full transparency when the cursor speed varied from 0 to 1500 pixels/s, according to a sigmoid function. These values were fixed during pre-tests by conducting an analysis of pointing movements that were performed with the Aesculap's pointing system. With TARGET, the target was scaled by 1.618.

For the three combined techniques, we detected the ballistic phase using a cursor speed threshold, which we determined to be 500 pixels/s. This speed threshold was evaluated at around 150 pixels/s in [6, 18], but the settings were different than ours. For applying the threshold of 500 pixels/s we filtered the cursor's speed by computing the mean of the 4 last instant speeds. Thus, we excluded the speed peaks induced by the pedal presses and the noise induced by the hand jitters amplified by the distance between the users and the screen. This method clearly revealed one ballistic phase per pointing movement on smoothed velocity profiles.

Task and Procedure

Participants achieved ISO 9241-9 pointing tasks with 9 targets, the first one being the start target and 8 target selections being then recorded (Figure 7). The Aesculap's C++ software platform provided the workflow management functions, as well as the application *black-cockpit* look and feel (Figures 1, 3, 5 and 7). The targets included a white "0" for goal-targets and a gray "X" for distractor targets. A thin border highlighted the designated target. This target highlight corresponds to a minimal feedforward mechanism (*implicit*, *discrete, target*-based), present in all the techniques. A short sound confirmed successful selections. When participants selected a distractor target, we displayed the current error number during 100 ms in place of the "X" of the corresponding distractor. Mistaken trials were removed from selection time measures. We collected selection times, cursor trajectories, velocity profiles, errors and movement endpoints.

The participants were instructed to be as fast and as accurate as possible. They answered a System Usability Scale (SUS) [5] on the technique they just experienced before starting the next one. To do so, we provided an imaginary concrete context of distant interaction with a TV. Finally, the participants were asked to rank the 3 techniques they appreciated the most, having access to visual recall on paper sheet, similar to the screenshots of Figure 3. The techniques were counter-balanced across participants using a Latin square. The experiment lasted approximately 40 minutes and started with a 5-minute training session.

Design

The experiment used a 9 techniques x (2 x 2 x 2) target layouts within-participant design. We used the above described target layout including two distractor circles with the diameters of 400 and 1200 pixels. We varied the goal target circle diameters (600 or 1000 pixels). For each of these two movement amplitude conditions, we also considered the presence or not of an additional circle of distractors with a diameter of 800 pixels. The last condition is related to the shift of the centroids along the tangential direction: Distractors are either added to the bisector of every two slices or are not added. We therefore obtained 2 x 2 x 2 target layouts, randomly ordered for each participant.

Technique	Representation	Classification	Selection Time	Error Rate
MTE		Cell-centered ***	1308 ms	5.13 %
VTE		Cell-centered ***	1310 ms	6.34 %
TARGET		Target-centered ***	1394 ms	8.09 %
ROPE		Mixed-centered	1424 ms	6.03 %
VTEtoTARGET		Cell-centered ***	1431 ms	6.78 %
BUBBLE		Target-centered **	1440 ms	7.41 %
VTEtoCELL		Cell-centered ***	1456 ms	5.93 %
eVTE		Cell-centered ***	1477 ms	5.40 %
VTEtoROPE		Cell-centered **	1527 ms	7.61 %

x explicit, I implicit; static, discrete, continuous; augmented element, movement phase

Table 1. For each technique: (1) its matrix-based representation (2) its classification based on movement endpoints (3) its mean selection time (4) its mean error rate.

64 selections (8 layouts x 8 selections) per technique and per participant were recorded. A total of 10368 selections (9 techniques x 18 participants x 64 selections) were recorded.

Results

In this section, we use the following code for test significance: * denotes $p < .05$; ** denotes $p < .01$; *** denotes $p < .0001$. We removed 149 outliers (1,44 % of the data) due to double clicks or tracking freeze. For studying the movement endpoints, we computed $d_{toTarget}$ and d_{toCell} for each technique: We then compared them using pairwise Wilcoxon tests. One-way Repeated-Measures ANOVA revealed a significant effect of techniques on selection time ($F_{8,9496} = 36.8$ ***). We ran pairwise t tests using the Holm-Bonferroni method to compare selection time between techniques. For error rate, we performed a Pearson's Chi-squared independence test between success of target selection and the 9 techniques.

Table 1 shows the classification of the techniques based on movement endpoints. MTE, VTE, VTEtoTARGET, VTEtoCELL, eVTE and VTEtoROPE were measured cell-centered. BUBBLE and TARGET were target-centered, while ROPE was mixed-centered. The techniques are ordered in increasing order of mean selection time: in Table 1 from the top to the bottom and in Figure 8-left from the left to the right. MTE was the best performing technique, faster than the other techniques (*** with all, except VTE no significant difference). In second position, VTE was faster than every other technique (*** with all). In third position, TARGET was faster than VTEtoCELL**, eVTE*** and VTEtoROPE***. The difference with BUBBLE was not significant ($p = .08$). ROPE was faster than eVTE*. VTEtoROPE was by far the slowest technique (* with eVTE, ** with VTEtoCELL, *** with all the other techniques).

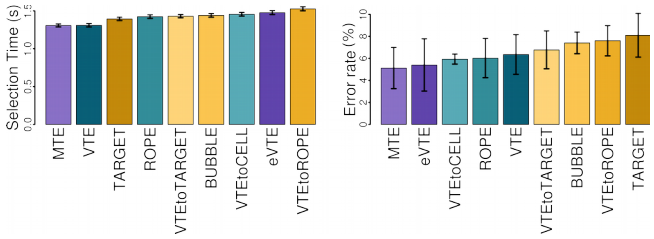
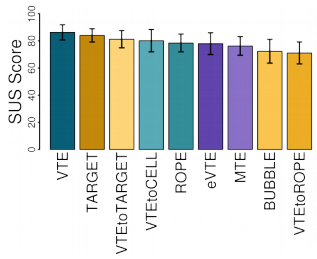


Figure 8. (left) Mean selection time, (right) Error rate per technique, with 95% confidence intervals.



Technique	Preference Ranking Score	#1	#2	#3
VTEtoCELL	36	5	3	2
TARGET	28	4	2	2
VTEtoTARGET	27	3	3	3
MTE	18	1	4	1
VTEtoRope	15	2	1	2
ROPE	14	1	2	3
VTE	10	0	3	1
BUBBLE	8	1	0	3
eVTE	6	1	0	1

Figure 9. (left) Mean SUS scores with 95% confidence intervals. (right) Preference ranking scores and the number of times the techniques were ranked 1st, 2d or 3rd.

In Figure 8-right, the techniques are ordered from the left to the right with increasing order of error rate. MTE, contrary to VTE, was significantly less error-prone than BUBBLE*, VTEtoROPE* and TARGET**. eVTE was less error-prone than VTEtoROPE* and TARGET*.

SUS Scores and Qualitative Results

In Figure 9-left, the techniques are ordered from the left to the right with decreasing order of mean SUS score. VTE and TARGET were perceived the most usable, with a mean SUS score above 84. BUBBLE and VTEtoROPE were perceived the less usable techniques with a mean SUS score of about 72.

Based on the qualitative data we collected, we computed a preference ranking score for each technique. The table of Figure 9-right shows them from the top to the bottom in decreasing order. We computed the preference ranking score (S) of a technique using the formula: $S = 5 \times 1^{st} + 3 \times 2^{d} + 3^{rd}$, where 1^{st}, 2^{d} and 3^{rd} were its rankings at the corresponding place. We checked that with lower coefficients (respectively 3, 2 and 1), the results remained similar. Participants preferred VTEtoCELL, then TARGET and VTEtoTARGET. VTE, BUBBLE and eVTE obtained the lowest preference ranking scores.

Discussion

In terms of target expansion algorithm, the Voronoi tessellation in Manhattan distance is promising. Indeed, MTE is significantly the fastest and the less error-prone technique. In Manhattan distances, Voronoi diagram lines are oriented along only 3 different directions: This implies more symmetrical and periodic patterns. This visual characteristic (related to human cognition) explains this good performance of MTE. MTE however suffered a lack

of learning effects (being the only technique based on Manhattan distances) as compared to the other techniques: This explains its bad mean SUS score. We discuss the other significant observed results in light of the design axes.

Dynamicity to Be Mininized

Selection time: The experiment shows that *the selection time increases with the dynamicity of the feedforward mechanisms* (H1). First, the three less dynamic techniques are the fastest ones. MTE and VTE, the two static techniques, are significantly the two fastest techniques. TARGET, an implicit dynamic-discrete technique, is faster than all the techniques involving a dynamic-continuous feedforward mechanism including those using implicit feedforward. However the differences between TARGET and ROPE, VTEtoTARGET and BUBBLE are not significant. Second, eVTE and VTEtoCELL, the two explicit dynamic techniques, are significantly slower than VTE and MTE, the two explicit static techniques.

Error rate: Contrary to VTE, eVTE is significantly less error-prone than TARGET and VTEtoROPE. One explanation is that continuous feedforward mechanisms constantly provide new information and attract the user's gaze. On the one hand, the resulting extra cognitive load makes the technique slower. On the other hand, *driving the user's attention can make the technique more accurate when useful information is provided at the right time of the pointing movement* (i.e. corrective phase for eVTE).

Subjective usability: Going one step further than static versus dynamic, we define the degree of dynamicity of a visual feedforward mechanism as the quantity of pixels that the mechanism modifies (or as the flow of modified pixels, which is the temporal derivative of the quantity of pixels the mechanism modifies). Based on this measure we classify the techniques in increasing degree of dynamicity as follows: VTE, MTE, TARGET, VTEtoTARGET, VTEtoCELL, then the techniques involving only dynamic-continuous mechanisms. Thus except for the special case of MTE explained above, there is a direct mapping between the mean SUS score and the degree of dynamicity. One explanation is that *dynamicity of visual feedforward increases user's cognitive workload and therefore decreases the perceived usability*. Such an additional cognitive workload is notable when using ray-casting pointing (with 5 degrees of freedom) that already causes a significant cognitive workload.

User preference: The users have a clear preference for discrete feedforward mechanisms. VTEtoCELL, VTEtoTARGET and TARGET, the only three techniques involving discrete feedforward, obtained the three best preference scores. This is consistent with previous work [16, 18], where a discrete target highlight mechanism was reported to perform better than a continuous cursor-based mechanism. We therefore confirmed these time performance results for the case of distant pointing and reinforced them with qualitative results as well as subjective usability results.

Explicit Expansion Observability

Shifting of the pointing movement endpoints: The results show a direct mapping between the explicit/implicit nature of the feedforward mechanisms and the shift of observed movement endpoints (Table 1). Indeed, the techniques involving explicit feedforward mechanisms (i.e. all except TARGET, BUBBLE and ROPE) are cell-centered techniques. The implicit techniques, TARGET and BUBBLE, are target-centered, while ROPE is mixed-centered, because the mini-ropes increase the observability of the cell boundaries without explicitly presenting them. If the users have no information on target expanded shapes, they rely on the target shapes. When providing explicit information, we allow the users to aim at the expanded target shape (cell-centered): This is fully consistent with the Optimized Initial Impulse Model [14]. *Explicit feedforward mechanisms thus have a direct impact on the optimization of the pointing task* (H2). As a consequence, the two fastest techniques (MTE and VTE) are cell-centered.

Error rate: When focusing on the movement landing phase (the ballistic phase ending and the potential corrective phase), the techniques providing explicit feedforward are the less error-prone ones (even if the differences are not significant) along with Rope Cursor. *An explicit feedforward allows the users to directly know the remaining distance to the cells' borders by providing a spatial information* and therefore to anticipate potential errors (H3). For the case of Rope Cursor, the mini-ropes' length also provides this information by an implicit continuous feedforward mechanism. Indeed Rope Cursor displays the distance of the cursor to the borders of the expanded cells but not the expanded cells themselves.

Compatibility of Augmented Elements for Combination of Feedforward Mechanisms

The quantitative results indicate that the combination of a space-based feedforward mechanism with a cursor-based feedforward mechanism of VTEtoROPE is not suitable. VTEtoROPE involves a too cognitively demanding switching between augmented elements. Contrastingly, *the combination of a space-based mechanism with a space-based or target-based mechanism is promising* and requires further studies. Moreover it is important to test these combined mechanisms as well as eVTE in the context of a GUI since their goal is to reduce the visual overload of a Voronoi diagram displayed on top of a GUI.

CONCLUSION

The key concept we introduce for the design of target expansion techniques is the atomic feedforward mechanism. The concept is key because it provides a visual feedforward aid to the user by employing a target expansion algorithm. The atomic feedforward mechanism has three design axes, namely dynamicity, expansion observability and augmented element. These design axes allow us now to classify existing target expansion techniques by studying the combination of atomic mechanisms according to the

phases of the pointing movement. We further define a matrix-based notation for concisely representing a target expansion technique as a combination of atomic feedforward mechanisms. The design space directs us to design six novel target expansion techniques with atomic/combined feedforward, thus demonstrating the generative power of this design space.

As a first experimental exploration of the design space we ran a controlled comparative experiment. To do so we define a new target layout that can handle non-centroidal target expansion. The experimental results show that: (1) The dynamicity of the feedforward mechanism increases the selection time and reduces user acceptance; (2) Explicit feedforward increases subjective usability and reduces the error rate. Moreover, in terms of our new techniques, MTE performs better than techniques in euclidean distance. Amongst the implicit feedforward techniques including Bubble Cursor, Rope Cursor supports an efficient error prevention mechanism, with an error rate comparable to the one of the explicit feedforward techniques.

Our analytical and experimental exploration of the design space was broad, considering all three axes. This exploration allowed us to identify specific areas in the design space that must be further explored in a systematic way (by varying parameters along one axis only) as part of laboratory and field experiments. Our long-term goal is to define a predictive model of the performance of a target expansion technique by modeling the factors related to the visual feedforward mechanisms.

ACKNOWLEDGEMENTS

We would like to thank Gilles Bailly, Marcos Serrano and the CHI reviewers for their insightful comments. Special thanks to George Serghiou for proof-reading the final version. This work has been partially supported by the LabEx PERSYVAL-Lab (ANR-11-LABX-0025-01).

REFERENCES

1. Bateman, S., Mandryk, R. L., Gutwin, C. and Xiao, R. Analysis and comparison of target assistance techniques for relative ray-cast pointing. *IJHCS 71*, 5 (2013), 511-532.
2. Bau, O. and Mackay, W. OctoPocus: a dynamic guide for learning gesture-based command sets. In *Proc. UIST 2008*, (2008), 37-46.
3. Baudisch, P., Zotov, A., Cutrell, E. and Hinckley, K. Starburst: a target expansion algorithm for non-uniform target distributions. In *Proc. AVI 2008*, (2008), 129-137.
4. Blanch, R. and Ortega, M. Benchmarking pointing techniques with distractors: adding a density factor to Fitts' pointing paradigm. In *Proc. CHI 2011*, (2011), 1629-1638.
5. Brooke, J. SUS: A Quick and Dirty Usability Scale. *Usability Evaluation in Industry* (1996), 189-194.
6. Chapuis, O., Labrune, J. and Pietriga, E. DynaSpot: speed-dependent area cursor. In *Proc. CHI 2009*, (2009), 1391-1400.
7. Du, Q., Faber, V. and Gunzburger, M. Centroidal Voronoi tessellations: Applications and algorithms. *SIAM Review 41,* 4 (1999), 637-676.
8. Grossman, T. and Balakrishnan, R. The bubble cursor: enhancing target acquisition by dynamic resizing of the cursor's activation area. In *Proc. CHI 2005*, (2005), 281-290.
9. Grossman, T., Kong, N. and Balakrishnan, R. Modeling pointing at targets of arbitrary shapes. In *Proc. CHI 2007*, (2007), 463-472.
10. Guillon, M., Leitner, F. and Nigay, L. Static Voronoi-Based Target Expansion Technique for Distant Pointing. In *Proc. AVI 2014*, (2014), 41-48.
11. Kuwabara, C., Yamamoto, K., Kuramoto, I., Tsujino, Y. and Minakuchi, M. Ghost-hunting: a cursor-based pointing technique with picture guide indication of the shortest path. In *Proc. IUI 2013*, (2013), 85-86.
12. Laukkanen, J., Isokoski, P. and Räihä, K. The cone and the lazy bubble: two efficient alternatives between the point cursor and the bubble cursor. In *Proc. CHI 2008*, (2008), 309-312.
13. McGuffin, M. and Balakrishnan, R. Fitts' law and expanding targets: Experimental studies and designs for user interfaces. *ACM TOCHI 12,* 4 (2005), 388-422.
14. Meyer, D. E., Abrams, R. A., Kornblum, S., Wright, C. E. and Smith, J. E. Optimality in human motor performance: ideal control of rapid aimed movements. *Psychological Review 95*, 3 (1988), 340-70.
15. Mott, M. and Wobbrock, J. Beating the bubble: using kinematic triggering in the bubble lens for acquiring small, dense targets. In *Proc. CHI 2014*, (2014), 733-742.
16. Su, X., Au, O. and Lau, R. The Implicit Fan Cursor: A Velocity Dependent Area Cursor. In *Proc. CHI 2014*, (2014), 753-762.
17. Tse, E., Hancock, M. and Greenberg, S. Speech-filtered bubble ray: improving target acquisition on display walls. In *Proc. ICMI 2007*, (2007), 307-314.
18. Vanacken, L., Grossman, T. and Coninx, K. Multimodal selection techniques for dense and occluded 3D virtual environments. *IJHCS 67*, 3 (2009), 237-255.
19. Vermeulen, J., Luyten, K., Van den Hoven, E. and Coninx, K. Crossing the bridge over Norman's gulf of execution: revealing feedforward's true identity. In *Proc. CHI 2013*, (2013), 1931-1940.
20. Zhai, S., Conversy, S., Beaudouin-Lafon, M. and Guiard, Y. Human on-line response to target expansion. In *Proc. CHI 2003*, (2003), 177-184.