



**HAL**  
open science

## Interaction multi-dispositifs pour tableurs sur tablette

Gary Perelman, Marcos Serrano, Célia Picard, Mustapha Derras, Emmanuel Dubois

► **To cite this version:**

Gary Perelman, Marcos Serrano, Célia Picard, Mustapha Derras, Emmanuel Dubois. Interaction multi-dispositifs pour tableurs sur tablette. 29ème Conférence francophone sur l'Interaction Homme-Machine (IHM 2017), AFIHM: Association Francophone de l'Interaction Homme-Machine, Aug 2017, Poitiers, France. pp.105-116, 10.1145/3132129.3132132 . hal-01578468

**HAL Id: hal-01578468**

**<https://hal.science/hal-01578468v1>**

Submitted on 29 Aug 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Interaction multi-dispositifs pour tableurs sur tablette

## *Multi-device interaction for spreadsheet on tablets*

Gary Perelman  
IRIT/Berger-Levrault  
Université de Toulouse – ELIPSE,  
31260, Toulouse, France  
gary.perelman@berger-levrault.com

Marcos Serrano  
IRIT  
Université de Toulouse – ELIPSE,  
31260, Toulouse, France  
marcos.serrano@irit.fr

Célia Picard  
Berger-Levrault  
31670, Labège, France  
celia.picard@berger-levrault.com

Mustapha Derras  
Berger-Levrault  
31670, Labège, France  
mustapha.derras@berger-levrault.com

Emmanuel Dubois  
IRIT – Université de Toulouse –  
ELIPSE, 31260, Toulouse, France  
emmanuel.dubois@berger-levrault.com

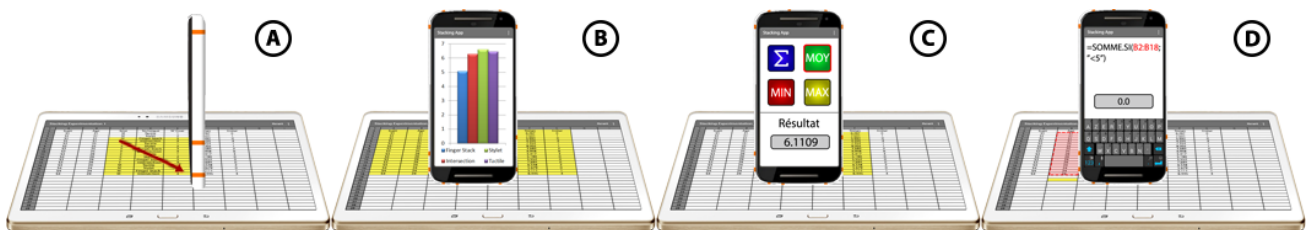


Figure 1 : Illustration du stacking d'un smartphone sur une tablette pour interagir avec une application tableur: sélection d'une plage (A), visualisation d'un graphique (B), calcul d'une moyenne (C) et édition d'une cellule (D). Dans cet article, nous focalisons nos travaux sur la sélection (A).

### ABSTRACT

Multitouch interaction with spreadsheets on mobile devices raises multiple interaction issues. In this paper, we present the design and evaluation of novel interaction techniques for spreadsheet based on the stacking paradigm, which consists in laying one edge/face/corner of a smartphone on a tablet screen. We first explore the available gestures and degrees of freedom of our prototype to delimitate the possible interaction's vocabulary. Then, we propose novel interaction techniques to perform range selection in a spreadsheet on a tablet. To detect the smartphone on the tablet screen, we conceive a conductive smartphone copper-shell. We evaluate our prototype against the regular tactile interactions provided by spreadsheet apps. Our studies reveal that some of our interaction techniques perform between 15% and 20% faster for range selection.

### CCS CONCEPTS

• **Human-centered computing** → Interaction techniques

### KEYWORDS

Interaction technique, Tablette, Smartphone.

### RÉSUMÉ

L'interaction multitouch avec des tableurs sur tablettes soulève de nombreuses problématiques d'interaction. Dans cet article, nous présentons la conception et l'évaluation de nouvelles techniques d'interaction pour tableurs basées sur une manipulation de type stacking, qui consiste à poser l'arête/le coin/la face d'un smartphone sur l'écran d'une tablette. Dans un premier temps, nous explorons l'espace des gestes et les degrés de libertés du stacking afin de délimiter le vocabulaire d'interaction possible. Ensuite, nous proposons plusieurs techniques d'interaction pour réaliser la tâche de sélection de plages de données dans un tableur sur tablette. Afin d'implémenter la détection du smartphone par l'écran de la tablette, nous construisons une coque conductive en cuivre. En utilisant notre prototype nous comparons nos techniques à l'interaction tactile classique proposée par les tableurs. Nos résultats montrent que certaines de nos techniques d'interaction sont entre 15% et 20% plus rapides que le tactile pour sélectionner une plage de données.

### MOTS-CLEFS

Technique d'interaction, Tablette, Smartphone.

### 1 INTRODUCTION

Les tableurs sont des logiciels largement utilisés pour la visualisation et la manipulation de données multidimensionnelles et/ou en grande quantité. Scaffidi [1] estime ainsi qu'il y a eu en 2012, 55 millions d'utilisateurs de tableurs aux Etats-Unis. Ils sont utilisés dans de nombreux métiers pour représenter des données, les analyser [2] ou réaliser des modèles mathématiques [3]. Ils possèdent donc

© ACM, 2017. This is the author's version of the work. It is posted here by permission of ACM for your personal use. Not for redistribution. The definitive version was published in Actes de la 29<sup>ème</sup> conférence francophone sur l'Interaction Homme- Machine, IHM '17, August 28–September 1, 2017, Poitiers, France  
<https://doi.org/10.1145/3132129.3132132>

une très grande variété de fonctionnalités ce qui rend leur utilisation particulièrement complexe.

Ils sont désormais également utilisés sur des dispositifs mobiles tels que les tablettes ou les smartphones [4]-[5], soulevant alors de nouvelles problématiques d'interaction. En effet, la petite taille des écrans de ces dispositifs et le vocabulaire d'interaction tactile limité en complexifient l'utilisation : le swipe et le pinch/spread sont associés aux actions de pan et de zoom et ne peuvent donc pas être utilisés pour l'interaction avec les données. De plus, les interactions multitouch peuvent rapidement s'avérer inconfortables [6]. Il en résulte soit une réduction du périmètre fonctionnel des tableurs, qui, en limitant les capacités de traitement et de mise en forme, simplifie leur usage [7], soit, une complexité accrue des interactions requises pour la plupart des commandes : des étapes d'interaction intermédiaires sont nécessaires et l'usage de multiples menus imbriqués également.

Dans cet article, nous visons à simplifier l'utilisation des tableurs sur tablette en conservant l'intégralité du périmètre fonctionnel. Pour ceci, nous utilisons le paradigme de stacking, qui consiste à mettre en contact une surface interactive complémentaire (e.g. un smartphone) avec une surface support, (e.g. une tablette) sur laquelle est exécutée le tableur. L'intérêt est triple : 1) La surface d'affichage initiale est étendue par la présence de l'écran complémentaire ; 2) Les capacités d'interaction avec la surface support (e.g. la tablette) sont accrues par la détection des manipulations physiques de la surface complémentaire mise en contact ; 3) La surface complémentaire (e.g. le smartphone) constitue une zone tactile additionnelle pour l'utilisateur offrant un moyen d'accès à de multiples commandes Figure 1. Dans le cadre de notre étude, nous avons considéré une tablette comme surface support et un smartphone comme surface complémentaire. Nous avons également choisi d'explorer uniquement la mise en contact de l'une des quatre arêtes du smartphone sur l'écran de la tablette.

Dans une première étude, nous avons exploré les différents gestes physiques, tactiles et combinés (physiques + tactiles) qui peuvent être réalisés avec la surface mise en contact. Les résultats de cette étude nous ont permis de délimiter un vocabulaire d'interaction confortable. Ensuite, nous avons conçu et évalué plusieurs techniques d'interactions basées sur le stacking visant à sélectionner des plages de données (cette tâche étant nécessaire à l'utilisation de la majorité des commandes, filtrages et créations de graphiques). Les résultats montrent que certaines de nos techniques exploitant le paradigme de stacking permettent de sélectionner une plage de cellules entre 15% et 20% plus rapidement qu'avec les interactions tactiles classiques des tableurs.

Nos contributions sont : 1) L'exploration de l'espace de conception du paradigme de stacking (gestes, orientations, combinaisons) pour les tableurs sur tablettes ; 2) Son opérationnalisation au travers de la conception d'une coque conductive et des algorithmes associés permettant la détection du smartphone sur l'écran de la tablette ; 3) La conception et l'évaluation de techniques d'interaction basées sur le paradigme de stacking pour la sélection de plages de cellules dans un tableur sur tablette.

## 2 ÉTAT DE L'ART

### 2.1 Préliminaire : utilisation d'un tableur sur tablette

L'utilisation d'un tableur sur tablette [4]-[5] repose sur les gestes tactiles classiques sur surfaces interactives. Ainsi, chaque élément du vocabulaire tactile (swipe, pinch/spread, tap, double tap) est associé à une action dans le tableur (navigation, zoom, sélection, édition). Par conséquent, toute autre action est réalisée par une combinaison de ces gestes de base. Par exemple, pour sélectionner plusieurs cellules une combinaison de trois gestes est nécessaire Figure 2 : 1) un tap sur une cellule pour la sélectionner ; 2) la sélection d'une des ancres située sur les angles de la cellule en plaçant un doigt sur l'ancre sans la relâcher ; 3) un geste de drag, qui permet d'étendre la sélection à plusieurs cellules.



Figure 2 : Sélection d'une plage de cellules avec l'interaction tactile : Tap sur une cellule (1), sélection d'une ancre (2) et drag pour étendre la sélection (3).

Pour réaliser des actions plus complexes (i.e. filtrer, construire ou paramétrer des graphiques, faire des sommes, etc.), la sélection d'une plage de cellules est suivie d'un premier tap sur le bouton des menus, généralement affiché dans le coin supérieur-droit de l'écran. Cette action fait apparaître un menu déroulant contenant de nombreuses sous-catégories pour chaque groupe d'actions spécifiques parmi lesquelles l'action recherchée doit être sélectionnée par un nouveau tap. Aucun raccourci clavier n'étant possible, il en résulte une interaction souvent longue et fastidieuse. C'est pourquoi plusieurs techniques d'interaction ont été proposées pour optimiser l'interaction avec un dispositif mobile.

### 2.2 Interaction avec dispositifs mobiles

Les travaux visant à simplifier l'interaction avec les dispositifs mobiles se focalisent majoritairement sur deux préoccupations : l'ajout de degrés de libertés (DDL) supplémentaires pour étendre le vocabulaire d'interaction tactile limité [8]-[10] et/ou l'extension de la surface d'affichage via l'ajout d'un écran supplémentaire 0-[13]. Ainsi, MagGetz [8] propose d'utiliser plusieurs interacteurs tangibles pouvant être liés à des actions spécifiques de l'application comme des menus ou des boutons. D'autres travaux proposent d'utiliser l'espace d'interaction autour du dispositif mobile [9]-[10]. Dans ce cas, l'interaction autour du dispositif permet d'ouvrir des menus de l'application ou peut être associé à la manipulation des données (i.e. manipuler un objet 3D).

Certains travaux proposent d'utiliser un second dispositif mobile afin d'étendre le vocabulaire d'interaction et la surface d'affichage. Par exemple, dans Thaddeus 0, la position du smartphone par rapport à la tablette (qui sert de référentiel) a un impact sur les manipulations disponibles. Dans BATMAN

[11], la position et l'orientation du smartphone permettent de configurer les actions réalisées sur la tablette.

Cependant, ces approches se basent sur l'utilisation de capteurs externes afin d'étendre l'interaction. Or, ces capteurs peuvent être coûteux et contraignent fortement les conditions d'utilisation, voire même les rendre inopérantes en dehors des environnements de recherche.

### 2.3 Stacking

Le stacking, i.e. la détection d'un dispositif mobile sur l'écran d'un dispositif support, ne nécessite aucun capteur supplémentaire. Par exemple, dans THAW [12], l'écran d'un ordinateur de bureau affiche un marqueur spécifique (motif coloré) qui est détecté par la caméra d'un smartphone. Ce marqueur permet au smartphone de calculer sa position par rapport à l'écran. L'utilisation de son IMU (Inertial Measurement Unit) embarquée est utilisée pour détecter son orientation. Ainsi, le smartphone peut être utilisé comme lentille magique ou directement comme objet tangible dans l'application (e.g. une plateforme 2D sur lequel le personnage du joueur peut sauter). Dans ACTUI [13], l'écran tactile de la tablette peut être utilisé pour détecter la position et l'orientation d'un smartphone. Pour ce faire, des motifs de cuivre ont été placés sur l'arête d'un smartphone. La tablette détecte ce motif particulier afin de calculer la position du smartphone sur l'écran. Le smartphone est utilisé pour explorer la 3<sup>ème</sup> dimension d'un modèle 2D affiché sur l'écran principal.

Toutes ces techniques d'interaction ont été conçues et testées pour offrir des fonctionnalités d'interaction originales et complémentaires aux interactions de base avec l'application. Cependant, ces formes d'interaction basées sur du stacking ou une utilisation avancée d'un dispositif mobile, n'ont pas été mises en œuvre, à notre connaissance, pour couvrir les fonctionnalités traditionnelles d'un environnement informatique, telles que le pointage, la sélection ou l'édition. Le stacking n'a pas non plus fait l'objet d'une étude plus systématique visant à caractériser ses conditions d'utilisation. Ces deux aspects sont au cœur des travaux présentés dans cet article.

## 3 DIMENSIONS DU STACKING

Dans notre contexte, le stacking consiste à mettre un smartphone au contact de l'écran tactile d'une tablette, sur laquelle est exécutée l'application tableur. Afin de décrire les dimensions disponibles, nous utilisons les deux axes de DECO [14], un espace de conception pour la composition de dispositifs : 1) l'arrangement physique (type d'assemblage, topologie des composants) et 2) les manipulations physiques (prise en main, actions utilisateurs).

### 3.1 Arrangement physique

Les quatre arêtes et les quatre coins du smartphone peuvent être mis en contact avec la tablette et reconnus par celles-ci. Ainsi, chaque arête et chaque coin peut être associé à des commandes ou menus spécifiques. Les faces avant et arrière du smartphone peuvent potentiellement être détectées également. Cependant, nous n'utilisons pas cette configuration dans nos travaux car le smartphone occulterait une surface trop importante de l'écran de la tablette, limitant ainsi la visualisation des données du tableur.

Dans chacune de ces configurations, l'orientation de l'écran du smartphone peut être prise en compte (face à l'utilisateur ou non), ainsi que l'angle d'inclinaison verticale du smartphone par rapport à l'écran de la tablette.

## 3.2 Manipulations physiques

### 3.2.1. Prise en main

Le smartphone est tenu par la main dominante. La tablette peut être tenue par la main non-dominante ou placée sur une surface plane (e.g. sur une table) : dans ce cas, l'utilisateur peut réaliser des gestes tactiles sur l'écran de la tablette à l'aide de sa main non-dominante. Nous limitons notre exploration à ce second cas. Dans ce contexte, le smartphone peut être considéré comme un interacteur tangible et tactile.

### 3.2.2. Actions utilisateurs

Un ensemble de gestes physiques, tactiles et composés (i.e. tactiles et physiques) peuvent être réalisés au cours du stacking. Des interactions tactiles peuvent être réalisées sur chacun des deux dispositifs. Les deux gestes délimitateurs du stacking sont le *stack* (i.e. poser le smartphone sur la tablette) et l'*unstack* (soulever le smartphone). Entre le *stack* et l'*unstack*, plusieurs manipulations physiques peuvent être réalisées avec le smartphone Figure 3 :

- *Translations* : le smartphone est déplacé sur l'écran de la tablette, et ce, sans perdre le contact.
- *Rotation à plat* : le smartphone est tourné sur l'écran de la tablette de telle sorte que l'arête/le coin actuellement stacké reste constamment détecté, i.e. au contact de l'écran tactile de la tablette.
- *Inclinaison* : le smartphone est incliné sur l'écran de la tablette. L'arête/le coin actuellement stacké reste en contact avec la tablette. Cependant, la valeur du gyroscope du smartphone change.
- *Rotation d'arête/de coin* : le smartphone est tourné de sorte que l'arête/le coin actuellement mis en contact change. Ainsi, il est possible de passer d'une arête à un coin et vice-versa.

Finalement, des gestes composés peuvent être réalisés. Ceux-ci sont basés sur la combinaison de deux gestes tactiles et/ou physiques.

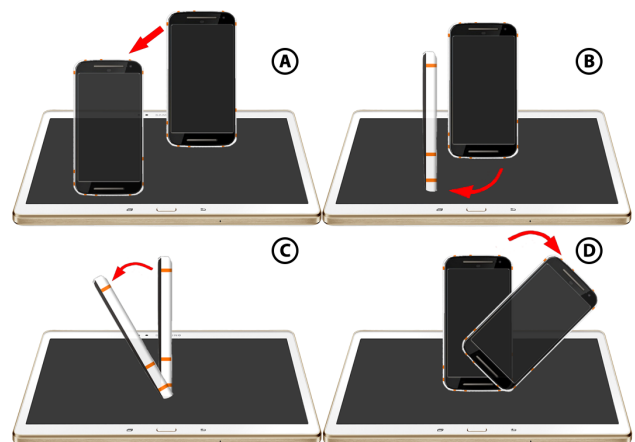


Figure 3 : Translation (A), rotation à plat (B), inclinaison (C) et rotation d'arête/de coin (D).

NB : dans la figure B les deux représentations du smartphone devraient être superposées (la rotation se faisant sans translation). Pour une meilleure lisibilité de la figure, nous les avons séparées.

#### 4 ÉTUDE PRÉLIMINAIRE : EXPLORATION DES GESTES

Le nombre de gestes possibles avec notre adaptation du stacking est particulièrement élevé (gestes tactiles sur les deux surfaces, manipulations physique du smartphone et gestes composés physique + tactile). Nous conduisons donc une étude préliminaire afin d'écartier les gestes inconfortables.

##### 4.1 Apparatus, tâche et données collectées

Dans cette étude préliminaire, nous avons utilisé une tablette Samsung Galaxy tab S (Android 5.1, octa-core à 1.9GHz et 1.3GHz, écran 10.5 pouces avec une résolution de 2560x1600 pixels) et un smartphone Samsung Galaxy S4 (Android 5.1, quad-core à 1.9GHz, écran 5 pouces avec une résolution de 1920x1080 pixels). Afin d'éviter les rayures sur la surface de la tablette, nous avons utilisé une protection plastique autocollante de tablette. Le smartphone était équipé d'une coque conductive afin d'évaluer notre dispositif dans des conditions d'utilisation réelles. Les deux dispositifs étaient éteints tout au long de l'étude. Par conséquent, leur écran était noir et n'affichait aucune information.

Six participants (âge moyen = 26.4, SD = 1.67) ont participé à cette étude préliminaire (dont 2 femmes). Ils étaient tous droitiers et familiarisés avec les dispositifs mobiles (6.9 en moyenne sur une échelle allant de 0 à 9). Aucune rémunération n'a été offerte aux participants. Les six participants devaient, en premier lieu, réaliser le geste de *stack* du smartphone sur l'écran de la tablette. Plusieurs orientations initiales pour le *stack* ont été testées (smartphone en mode Portrait et Paysage ; 4 rotations à plat pour les deux configurations ; Figure 4). Ensuite, les participants devaient réaliser (1) une des quatre manipulations physiques du smartphone (translation, rotation à plat, rotation d'arête, inclinaison – Figure 3), (2) un geste tactile sur l'écran du smartphone (touch, drag, pinch/spread) ou (3) un geste composé nécessitant une inclinaison du smartphone, puis un geste tactile sur son écran. Nous avons limité notre exploration des gestes composés à ceux basés sur une inclinaison suivie d'un geste tactile sur l'écran du smartphone. En effet, les gestes étant réalisés de manière séquentielle, réaliser un geste tactile après une translation, rotation à plat ou rotation d'arête revient à réaliser ce geste tactile suite à un *stack* initial dans une autre position/orientation : or ces conditions sont toutes deux testées dans le protocole. En revanche, l'inclinaison pouvant rendre l'écran partiellement inaccessible et n'étant pas intégrée dans le *stack* initial, nous avons choisi de l'évaluer dans cette étude préliminaire. D'autre part, deux gestes tactiles peuvent être réalisés consécutivement sans qu'aucun n'impacte la réalisation de l'autre; nous n'avons donc pas exploré les gestes composés qui en découlent. Pour finir, nous n'avons pas considéré les gestes tactiles sur l'écran de la tablette car leur réalisation est indépendante du mécanisme de stacking du smartphone (ils peuvent être réalisés avec la main non-dominante, que le smartphone soit tenu ou non).

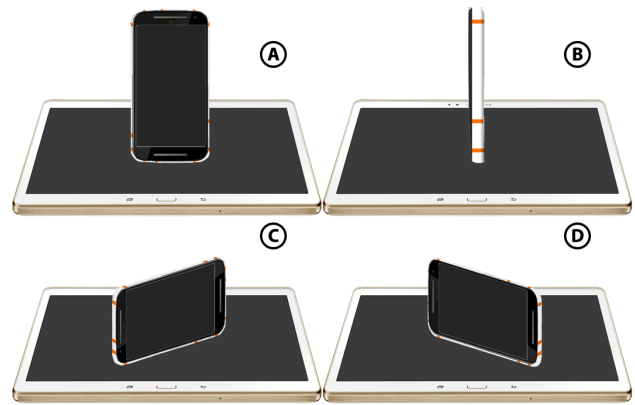


Figure 4 : Exemples d'orientation initiale: 0° et 90° en mode portrait (A et B) puis -45° et 45° en mode paysage (C et D).

Les gestes tactiles (2), combinés ou non à une inclinaison du smartphone (3), ont été réalisés de deux manières différentes : de façon unimanuelle en utilisant la main tenant le smartphone ou de façon bimanuelle, i.e. en impliquant la main non dominante ne tenant pas le smartphone.

Nous avons enregistré l'ensemble de l'expérimentation à l'aide d'une caméra. Après chaque geste, nous avons mesuré le niveau de confort ressenti par les participants à l'aide d'une échelle de Likert à 5 points. Nous avons aussi collecté les retours informels des utilisateurs. Cette étude a duré en moyenne 27 minutes pour chacun des six participants.

##### 4.2 Résultats

Toutes les manipulations physiques ont été considérées neutres, confortables ou très confortables (3 et plus) par la majorité des utilisateurs toutes configurations confondues (95% pour la translation ; 86% pour les rotations à plat ; 81% pour les rotations d'arête ; 89% pour les inclinaisons).

En ce qui concerne les gestes tactiles, les participants ont trouvé le pinch/spread moins confortable que le touch et le drag (66% pinch/spread ; 96% touch ; 89% drag). De plus, cette différence est accentuée lorsque les participants devaient réaliser les gestes tactiles en situation unimanuelle (52%, 96% et 81% ont noté 3 et plus respectivement pour le pinch/spread, le touch et le drag).

Lors de la réalisation de gestes composés, la majorité des utilisateurs a trouvé le pinch/spread inconfortable ou très inconfortable (70% l'ont noté 1 ou 2). Ainsi, plusieurs participants ont expliqué que : « L'écran n'est pas accessible » ou que « Faire un geste tactile est impossible dans cette configuration ». De plus, dans certaines orientations initiales du smartphone, les participants n'ont pas pu réaliser le geste tactile à cause de blocages physiques (notamment du poignet). En conséquence, nous avons choisi de ne pas considérer l'utilisation de gestes composés dans la suite de nos travaux et d'éliminer le geste du pinch/spread. De plus, nous avons remarqué durant l'étude que certaines zones de l'écran du smartphone étaient touchées lorsque les utilisateurs le prenaient en main. Dans l'orientation paysage, la moitié droite de l'écran était involontairement touchée par les participants lorsqu'ils tenaient le smartphone. En mode portrait, c'est la moitié basse de l'écran qui était touchée. D'autre part, ces zones de l'écran peuvent être amenées à varier selon la technique d'interaction utilisée. Il apparaît donc nécessaire de prendre en compte le fait que pour réaliser un geste tactile sur le smartphone, les utilisateurs devront toucher une section

spécifique de l'écran définie selon l'orientation du smartphone (portrait et paysage) mais aussi selon la technique d'interaction proposée.

## 5 TECHNIQUES DE STACKING POUR TABLEUR

Nous avons conçu trois techniques d'interaction basées sur le principe du stacking pour un tableur sur tablette. Ces techniques permettent de sélectionner une plage de données. En effet, cette tâche est nécessaire à l'utilisation de la majorité des commandes d'un tableur. Bien que l'intérêt principal du stacking réside dans la surface complémentaire que représente son écran (permettant d'activer des commandes ou visualiser des résultats et/ou graphiques), nous avons choisi de proposer des techniques d'interactions basées sur le stacking pour la phase préliminaire de sélection de plage de données dans un souci de cohérence et de fluidité.

Dans notre implémentation de ces techniques, seule une arête du smartphone (le coté bas) a été utilisé pour le stacking. Le smartphone est donc utilisé en mode portrait. Dans cette section, nous décrivons en détail chacune de ces techniques.

### 5.1 Intersection

Cette première technique exploite le fait qu'une cellule est à l'intersection d'une ligne et d'une colonne. Par conséquent, un ensemble de cellules adjacentes, une plage de cellules, se situe à l'intersection d'un ensemble de lignes et d'un ensemble de colonnes.

Afin de réaliser la sélection d'une plage avec cette première technique, l'utilisateur doit donc sélectionner l'ensemble des colonnes d'une part et l'ensemble des lignes d'autre part. Ces deux étapes sont indépendantes l'une de l'autre et peuvent être réalisées dans l'ordre souhaité. Pour sélectionner des colonnes, le *stack* doit être fait avec une orientation initiale verticale (rotation à plat à  $90^\circ \pm 25^\circ$ ) Figure 5 - Gauche. Pour sélectionner les lignes, un *stack* du smartphone avec une orientation initiale horizontale (rotation à plat à  $0^\circ \pm 25^\circ$ ) est requis Figure 5 - Droite. Dès qu'une première colonne/ligne est sélectionnée, et tant qu'il n'y a pas eu d'*unstack*, une translation du smartphone permet d'étendre la sélection. Lorsque le participant *unstack* le smartphone, les colonnes/lignes sélectionnées se figent.



Figure 5 : Illustration de la technique Intersection: *stack* (1) puis translation (2) pour sélectionner les colonnes et *stack* (3) suivi d'une translation (4) pour sélectionner les lignes.

Afin de modifier une sélection déjà réalisée, l'utilisateur peut produire un *stack* avec le smartphone sur la première ou dernière ligne/colonne de sa sélection : une translation du smartphone permet alors de modifier l'ensemble de lignes/colonnes sans affecter le reste de la sélection.

Dans le cas d'une plage de lignes/colonnes entières (par exemple, pour sélectionner l'intégralité des cellules des lignes 1 à 6), une seule étape de sélection suffit : aucune intersection n'est nécessaire.

Si la plage à sélectionner dépasse des limites de l'écran, le smartphone peut être approché près du bord afin de faire défiler l'écran. La fonction de transfert contrôlant la vitesse du défilement est identique à celle de la technique d'interaction tactile classique.

### 5.2 Stylet

Cette seconde technique s'inspire de l'utilisation classique d'un stylet pour tracer une forme rectangulaire : sélection d'un coin et déplacement jusqu'au coin opposé en diagonal. Un stylet (représenté par le smartphone) est différenciable du doigt et, par conséquent, ne déplace pas la vue lorsqu'il est placé puis glissé sur l'écran.

Pour utiliser cette technique, un *stack* initial du smartphone sélectionne une cellule. Cette cellule représente le coin haut-gauche de la plage sélectionnée. Puis, une translation du smartphone permet d'étendre la sélection. Un point rouge affiché sur l'écran de la tablette, 1cm devant la face avant du smartphone (au milieu de la longueur de l'arête en contact) évite le problème de l'occultation des cellules à sélectionner par le smartphone : il représente le « curseur » de sélection et donc en particulier la position de la cellule actuellement sélectionnée Figure 6 - Gauche. Lorsque le smartphone est translaté, le point rouge définit le coin bas-droit de la plage sélectionnée : un rectangle de la largeur/hauteur choisie par l'utilisateur est ainsi formé Figure 6 - Droite.



Figure 6 : Illustration de la technique Stylet : *stack* du smartphone (1) et translation (2) pour étendre la sélection.

Afin de modifier une sélection déjà réalisée, et de façon similaire à la technique tactile classique, des ancrs sont affichées sur les coins haut-gauche et bas-droit de la plage sélectionnée. Ainsi, il est possible de corriger une sélection en réalisant un *stack* sur une de ces ancrs. Il convient ensuite de translater le smartphone pour ajuster la sélection.

Dans le cas d'une plage de lignes/colonnes entières, il est nécessaire de placer le smartphone directement sur l'intitulé de la ligne/colonne. Si la plage à sélectionner dépasse des limites de l'écran, le principe de défilement des données affichées à l'écran ainsi que la fonction de transfert sont identiques à ceux de la technique Intersection.

### 5.3 Finger Stack

Cette technique d'interaction est une technique bimanuelle. Elle consiste à contrôler d'une main la position d'un coin de la plage sélectionnée et d'utiliser le stacking avec l'autre main pour définir la position du coin diagonal opposé de la plage sélectionnée.

Dans le cas d'un utilisateur droitier par exemple, pour réaliser la sélection, l'utilisateur doit donc poser un doigt de sa main gauche sur la tablette pour définir le coin haut-gauche de la sélection Figure 7 - Gauche. Puis il doit réaliser un *stack* du smartphone avec sa main droite pour définir le coin bas-droit Figure 7 - Droite de la sélection. Pour les utilisateurs gauchers, le touch peut être réalisé de la main droite et le *stack* de la

main gauche. L'arête du smartphone étant identifiée de manière unique, ce mécanisme de sélection est clairement différenciable d'un simple multi-touch qui déclencherait l'action de zoom par défaut. La position du smartphone et celle du doigt peuvent être ajustées par des translations, tant qu'aucun *unstack* n'est détecté. Un point rouge dont le fonctionnement est identique à celui défini pour *Styler* permet de connaître l'emplacement du « curseur » en évitant l'occlusion par le smartphone des cellules à sélectionner. Le principe de cette technique est de permettre de réaliser le rectangle de sélection directement quand la sélection est entièrement visible à l'écran de la tablette en limitant les translations du doigt ou du smartphone. Dans ce cas, un simple touch puis *stack* permet de sélectionner l'intégralité de la plage demandée Figure 7.

Afin de modifier la sélection, il est possible de placer le doigt ou le smartphone sur leur coin respectif de la plage sélectionnée : ces coins peuvent ensuite être déplacés via une translation du doigt/smartphone sur l'écran de la tablette.

Dans le cas d'une plage de lignes/colonnes entières, il est nécessaire de placer le doigt sur l'intitulé de ligne/colonne souhaité. Lors d'un *stack* du smartphone, sa position délimitera les lignes/colonnes entières sélectionnées.



**Figure 7 : Illustration de la technique Finger Stack : tap pour sélectionner le coin haut-gauche (1) puis *stack* pour choisir le coin bas-droit (2).**

Si la plage à sélectionner dépasse des limites de l'écran, le principe de défilement des données affichées à l'écran ainsi que la fonction de transfert sont identiques à ceux de la technique Intersection.

## 6 IMPLÉMENTATION DU PROTOTYPE

Afin d'implémenter le stacking du smartphone sur la tablette, nous avons besoin de pouvoir détecter la position et l'orientation d'une arête du smartphone sur la surface capacitive de la tablette.

### 6.1 Coque conductive

Différentes approches ont été proposées pour combiner interaction tangible et surface conductive. Elles s'appuient sur la détection des doigts autour d'un objet non conducteur contraignant la position des doigts [15], l'utilisation d'effecteurs [16] ou de capteurs [17] spécifiques. Pour éviter l'ajout de contrainte de prise ou main ou de technologie trop complexe et potentiellement coûteuse dans une coque de smartphone, notre approche s'apparente plus à celles de CapStones, ZebraWidgets et CapTUI [18]-[19].



**Figure 8 : Ecran principal de l'application (gauche), exemple de sélection (milieu) et validation automatique (droite). Au cours d'une sélection, la plage de donnée sélectionnée est affichée en jaune, une fois validée, la sélection est affichée en vert. Les intitulés de lignes/colonnes sont affichés en rouge lorsque leur sélection est nécessaire mais pas encore faite, en jaune lorsque leur sélection n'était pas demandée mais a été faite et en vert si la sélection est valide.**

Il s'agit d'une coque conductive : elle est basée sur une coque de smartphone classique en plastique que nous avons augmentée de cuivre. Sur chaque arête du smartphone sont disposées trois fines bandes de cuivre, des contacteurs. Ils sont en contact avec des feuilles de cuivre recouvrant l'arrière du smartphone pour maximiser la conductivité entre l'utilisateur et les fines bandes de cuivre. Le cuivre étant conducteur, lorsque les contacteurs sont mis en contact avec la surface capacitive de la tablette, celle-ci les détecte. Nous identifions chaque arête via une répartition spécifique des trois contacteurs en cuivre, en modifiant les distances relatives entre eux Figure 9.



**Figure 9 : La coque de cuivre (gauche) et un motif d'arête (droite).**

### 6.2 Détection des gestes

Afin de détecter la translation et les rotations à plat du smartphone, nous avons développé et implémenté sur la tablette un algorithme qui détecte chaque point des motifs du smartphone et les suit au cours du temps. Ceci nous permet de connaître la position, l'orientation et la direction des mouvements du smartphone. Concernant les rotations d'arête/de coin et les inclinaisons, les données de l'IMU du smartphone sont transmises à la tablette afin de réaliser les calculs d'orientation. Les résultats de ces calculs peuvent ensuite être transmis au smartphone afin de déclencher des commandes ou actions spécifiques. Toutes les communications entre la tablette et le smartphone peuvent être réalisées à l'aide d'un réseau Wi-Fi local ou du Bluetooth.

### 6.3 Faisabilité industrielle

Notre coque en cuivre augmente le prix de base de la coque en plastique de moins de 2€. Sa construction ne nécessitant aucun matériel rare ou complexe à assembler, la production de cette coque à grande échelle est envisageable. D'ailleurs, à la suite de la validation expérimentale de nos techniques (cf. Étude : SÉlection d'une plage de données), nous avons fabriqué en

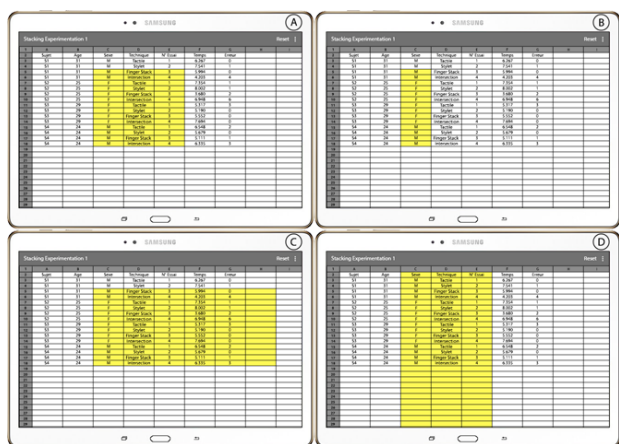
collaboration avec un lycée professionnel une coque imprimée en 3D avec des contacteurs métalliques ainsi qu'une coque intégralement usinée en aluminium afin de démontrer la faisabilité industrielle de notre prototype.

## 7 ÉTUDE : SÉLECTION D'UNE PLAGE DE DONNEES

Le but de cette expérience est d'évaluer la performance de nos trois techniques d'interaction en les comparant avec l'interaction tactile classique sur tablette pour une tâche de sélection de plages de données.

### 7.1 Tâche

Les participants devaient réaliser une tâche de sélection de plage de données dans un tableur sur tablette. La consigne était de réaliser cette tâche le plus rapidement possible avec chacune des quatre techniques d'interactions proposées. Afin de réduire les biais expérimentaux, nous avons désactivé les actions de pan et de zoom. Ces deux actions ne sont pas directement liées à la sélection d'une plage dans un tableur. Leur utilisation (fortement dépendante de la stratégie de l'utilisateur) affecte le temps de sélection, toutes techniques d'interaction confondues, ce qui pourrait causer une trop grande disparité entre les résultats des différents participants. Il était cependant possible de faire défiler l'écran avec chacune des techniques de sélection que nous avons évaluées.



**Figure 10 : Différentes plages (de droite à gauche et de haut en bas) : carré, une seule colonne, rectangle horizontal et colonnes entières.**

Afin de représenter la diversité des formes de sélections de cellules possibles dans un tableur, nous avons classé les différents types de plages possibles en sept catégories : lignes entières ; colonnes entières ; une seule ligne ; une seule colonne ; une plage carrée ; une plage rectangulaire horizontale et une plage rectangulaire verticale Figure 10. Ainsi, au cours de l'expérimentation, les sujets devaient réaliser la sélection de chacune de ces plages.

De plus, ces plages peuvent ne pas être intégralement contenues dans l'écran de démarrage. Nous avons donc testé deux conditions pour chacune des sept plages : une première condition dans laquelle la plage est intégralement comprise dans l'écran de la tablette et une seconde condition dans laquelle la plage sort partiellement de l'écran. Chacune des

techniques d'interaction permettait de faire défiler l'écran afin de compléter la sélection.

Un essai, i.e., une sélection de plage, se déroulait comme suit. Un premier écran affichait sur la tablette les informations de la plage à sélectionner sous format mathématique (par exemple :  $n * n$  pour le carré), en intitulé de cellules (par exemple : A6 à L35) ainsi qu'à l'aide d'une image schématique Figure 8 - Gauche. L'utilisateur appuyait sur un bouton tactile sur la tablette pour débiter l'essai et afficher les cellules du tableur. Les en-têtes des colonnes et des lignes de la plage à sélectionner étaient mise en évidence par un fond rouge. Si les cellules sélectionnées étaient incluses dans la plage visée, les en-têtes correspondantes passaient au vert, sinon ils devenaient jaunes Figure 8 - Milieu. L'utilisateur peut ainsi à chaque instant savoir quelles sont les cellules de la plage manquantes et celles sélectionnées à tort.

A tout instant, un bouton de réinitialisation placé dans la barre de menu de l'application permettait de replacer l'écran en position initiale et d'annuler la sélection. Ce mécanisme ne réinitialisait cependant pas le temps de sélection. Une fois que la zone de sélection était intégralement (et uniquement) sélectionnée, la validation se faisait automatiquement au bout de 500ms Figure 8 - Droite.

### 7.2 Techniques

Nous avons comparé les trois techniques de stacking décrites précédemment (Intersection, Stilet et Finger Stack) à une interaction tactile. L'interaction tactile est identique à celle disponible sur les tableurs existants [4]-[5] et décrite dans la section « Préliminaire : utilisation d'un tableur sur tablette ». Avec cette technique, si la plage à sélectionner dépasse des limites de l'écran, l'utilisateur peut approcher son doigt du bord de l'écran afin de faire défiler l'écran. La fonction de transfert contrôlant la vitesse du défilement est identique à celle de nos techniques basées sur le stacking.

### 7.3 Procédure

Notre étude suit un plan  $4*7*2$  intra-sujets avec comme facteurs : Technique d'interaction (Tactile, Intersection, Stilet, Finger Stack), Plage de sélection (lignes entières, colonnes entières, ligne seule, colonne seule, carré, rectangle horizontal, rectangle vertical) et Défilement (avec ou sans).

L'expérience est divisée en blocs, chaque bloc correspondant à une technique. L'ordre de passage des blocs pour chaque participant a été contrebalancé à l'aide d'un carré latin. Au sein d'un bloc, les participants avaient 14 essais d'apprentissage (7 plages avec et 7 plages sans défilement) pour se familiariser avec la technique d'interaction. Ensuite, ils devaient réaliser 42 sélections de plage mesurées (14 essais x 3 répétitions). L'ordre de ces essais était aléatoire. Entre chaque essai et entre chaque bloc, les participants pouvaient faire une pause de la durée de leur choix afin de limiter leur fatigue.

### 7.4 Matériel

Dans cette étude, nous avons utilisé une tablette Samsung Galaxy tab S (Android 5.1, octa-core à 1.9GHz et 1.3GHz, écran 10.5 pouces avec une résolution de 2560x1600 pixels) et un smartphone Samsung Galaxy S4 (Android 5.1, quad-core à 1.9GHz, écran 5 pouces avec une résolution de 1920x1080 pixels). Afin d'éviter les rayures sur la surface de la tablette, nous avons utilisé une protection plastique autocollante de tablette. Le smartphone était équipé de notre coque



conductive. Aucune information n'étant transmise ou récupérée du smartphone, celui-ci était éteint et par conséquent, son écran était noir. L'application tableur (développée par nos soins pour limiter les fonctionnalités au contexte de notre étude) était lancée sur la tablette tout au long de l'expérimentation. L'utilisateur était confortablement assis, la tablette posée sur la table devant lui et il devait tenir le smartphone avec sa main dominante.

## 7.5 Participants

Douze participants (âge moyen = 27.7, SD = 4.65) ont participé à cette étude (dont 5 femmes). Tous les sujets ont participé à notre étude préliminaire. Aucune rémunération n'a été offerte aux participants. Tous étaient droitiers et familiarisés avec les dispositifs mobiles (7.5 en moyenne sur une échelle allant de 0 à 9) et les tableurs sur ordinateur (7.0). En revanche, seul un sujet était familiarisé avec les tableurs sur tablette (7 pour cet utilisateur avec une moyenne de 2.1). L'expérimentation a duré en moyenne 64 minutes.

## 7.6 Données collectées

Au cours de cette étude, nous avons mesuré le temps mis par les participants pour réaliser chacune des tâches, soit un total de 4 techniques d'interaction x 42 essais (14 pages x 3 répétitions) x 12 participants = 2016 sélections.

A la fin de chaque bloc associé à une technique d'interaction, les utilisateurs devaient remplir un questionnaire SUS et pouvaient fournir des remarques informelles sur la technique. Il leur était aussi demandé de noter leur appréciation générale de la technique sur une échelle de Likert à 7 points. Une fois l'ensemble des blocs terminé, il leur était demandé de classer les techniques d'interaction selon leur préférence.

## 7.7 Résultats

Nous présentons dans un premier temps les résultats liés au temps de réalisation des sélections. Puis, nous présentons les résultats liés à la préférence et à l'utilisabilité des techniques d'interaction. Alors que dans le domaine de l'IHM les résultats quantitatifs sont habituellement analysés à l'aide d'un test de l'hypothèse nulle, ce type d'analyse est de plus en plus critiqué par la communauté statistique [20]-[21] et d'IHM [22]-[23]. Par conséquent, nous reportons nos résultats en utilisant des techniques d'estimation basées sur des amplitudes d'effets et des intervalles de confiance, conformément aux recommandations récentes de l'APA [24]. Les intervalles de confiance sont calculés à 95%.

### 7.7.1. Temps de réalisation

L'étude du temps de réalisation de la tâche de sélection montre une différence nette entre les quatre techniques d'interaction. En effet, notre analyse montre que les techniques Finger Stack et Intersection sont, en moyenne, respectivement 19.9% (CI [13.3%, 29.7%]) et 16.1% (CI [11.7%, 25.0%]) plus rapides que Tactile. En revanche, Stylet est 6.7% (CI [1.5%, 13.0%]) moins rapide que Tactile en moyenne Figure 11.

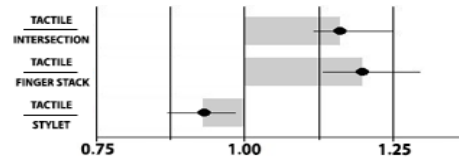


Figure 11 : Ratio du temps de sélection de Finger Stack, Intersection et Stylet par rapport à Tactile. Une valeur supérieure à 1 indique un temps plus court que Tactile.

Une analyse par type de plage (sans ajustement statistique pour comparaisons multiples) révèle que cette amélioration des performances en termes de vitesse de sélection avec les techniques de stacking par rapport au tactile reste vraie pour chacune des plages testées Figure 12 avec l'écart le plus important observé lors de sélections de lignes/colonnes entières.

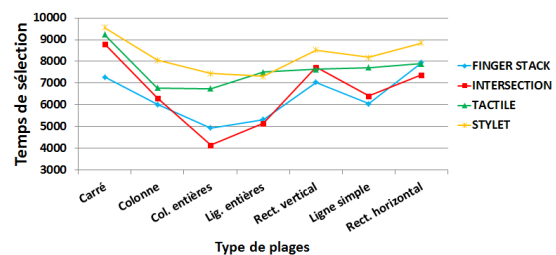


Figure 12 : Temps de sélection par type de plage et par technique d'interaction.

Une analyse par type de défilement (avec ou sans) révèle, que lorsqu'un défilement est nécessaire, le temps de sélection est allongé de manière similaire pour chaque technique Figure 13. Lorsqu'un seul défilement est requis, le temps nécessaire pour réaliser la sélection augmente, en moyenne, de 65.9%, CI [52.6%, 76.5%] pour Tactile ; de 63.0%, CI [54.4%, 74.5%] pour Stylet ; de 56.5%, CI [46.8%, 67.9%] pour Intersection et de 71.1%, CI [54.7%, 86.5%] pour Finger Stack.

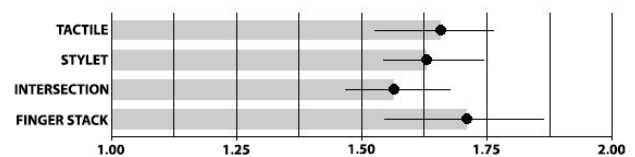


Figure 13 : Ratio du temps de sélection par technique en fonction du scroll (avec / sans). Une valeur supérieure à 1 indique un temps plus long avec le scroll que sans.

### 7.7.2. Erreurs

Les erreurs présentées ci-dessous correspondent au nombre moyen d'essais durant lesquels le bouton de réinitialisation a été utilisé au moins une fois. Les résultats obtenus montrent un nombre d'erreurs très variable selon la technique d'interaction (de 4.1% pour Tactile à 16.7% pour Brush) et selon les utilisateurs (9 utilisateurs ont moins de 5% d'erreurs, toutes techniques confondues alors que 2 autres utilisateurs ont 21% d'erreurs chacun). En effet, ces résultats dépendent de la stratégie utilisée au cours des essais pour chaque technique. Ainsi, et puisque nos mesures n'étaient pas assez précises pour permettre de différencier les erreurs selon les stratégies

utilisateurs, nous avons choisi de ne pas proposer une analyse plus poussée de ces résultats.

### 7.7.3. Utilisabilité et préférences utilisateurs

Les résultats des questionnaires SUS Figure 14 indiquent une meilleure utilisabilité des techniques Tactile et Intersection, respectivement 85.8 (CI [78.3, 91.3]) et 82.1 (CI [73.5, 87.3]), suivi par Finger Stack avec un score de 68.3 (CI [58.1, 76.9]) puis par Stylet avec un score de 52.7 (CI [36.7, 63.1]). L'appréciation générale des techniques, collectée à l'aide d'une échelle de Likert à 7 points, vient confirmer ces résultats avec 11 participants sur 12 ayant apprécié (5 et plus) Tactile, 10 ayant apprécié Intersection, 8 Finger Stack et 5 Stylet.

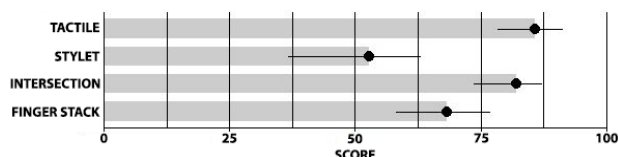


Figure 14 : Scores du questionnaire SUS par technique d'interaction

D'autre part, les remarques informelles fournies par les participants abondent dans le même sens que les résultats obtenus. Ainsi, la technique d'interaction Tactile a été décrite comme : « *Intuitive, cohérente avec les gestes tactiles classiques* ». Cependant, les utilisateurs ont trouvé « *la sélection avec le doigt complexe, on a du mal à viser la bonne case* ». La technique d'interaction Intersection a été qualifiée de « *Simple d'utilisation car les dimensions sont séparées, ce qui réduit la charge cognitive* ». En revanche, certains participants ont trouvé « *Difficile dans certains cas de voir les intitulés de colonne, le smartphone les cache* ». La technique d'interaction Finger Stack a été décrite comme « *Pratique car la sélection est quasiment immédiate si la plage est intégralement contenue dans l'écran* ». Cependant, il a été noté que « *L'utilisation des deux mains est contraignante* ». La technique d'interaction Stylet est considérée « *Agréable à utiliser une fois la sélection initiale réalisée* » mais « *Difficile à manipuler car j'ai du mal à viser la bonne cellule avec le point rouge* ».

### 7.7.4. Conclusion

Les résultats obtenus montrent que les techniques d'interaction Finger Stack et Intersection permettent de sélectionner une plage de données plus rapidement en moyenne qu'avec l'interaction Tactile (respectivement 19.9% et 16.1%). Ces résultats restent valides quelle que soit la plage de données à sélectionner et aussi dans le cas où un défilement de l'écran est nécessaire. En termes d'utilisabilité et de préférence utilisateur, la technique Intersection est aussi appréciée que Tactile (respectivement 82.1 et 85.8 en score SUS). Finger Stack est en revanche un peu moins aimée par les utilisateurs (68.3 en score SUS). Ce résultat s'explique par la difficulté de sélection qu'ont ressentie les utilisateurs à cause du viseur (point rouge) et par la nécessité d'utiliser les deux mains pour réaliser une sélection. D'autre part, la technique d'interaction Stylet est moins performante que Tactile (6.7% plus longue) et est beaucoup moins appréciée que Tactile (52.7 en score SUS). Nous discutons ces résultats et proposons des pistes d'améliorations pour certaines de nos techniques dans la section suivante.

## 8 DISCUSSION ET PERSPECTIVES

Les résultats obtenus révèlent la dimension particulièrement prometteuse du paradigme de stacking pour tableurs sur tablette. L'utilisation d'un interacteur tangible, possédé par une grande majorité de personnes, facilite la sélection des données. Néanmoins, nos techniques d'interaction peuvent être améliorées afin d'augmenter leurs performances, leur utilisabilité et leur acceptabilité.

### 8.1 Pistes d'amélioration

La technique d'interaction Stylet est peu appréciée des participants, notamment à cause des problématiques d'occultation qu'elle induit et de l'imprécision de la sélection due à la difficulté d'utilisation du viseur (point rouge). Ces limitations peuvent être palliées de trois manières différentes : le centre du smartphone (milieu de l'arête actuellement *stack*) pourrait être utilisé directement, ce qui augmente la précision de sélection, mais peut accentuer les problèmes d'occultation ; la coque pourrait intégrer un renflement au niveau du viseur pour matérialiser physiquement ce point de sélection ; le viseur pourrait être remplacé par le *stack* d'un coin du smartphone, plus précis et induisant moins d'occultation. Ces améliorations pourront être évaluées dans la suite de nos travaux.

La technique d'interaction Finger Stack permet une sélection rapide de plage, bien que son appréciation par les participants soit mitigée. Les critiques principales à l'encontre de cette technique sont liées à l'utilisation du viseur pour cibler la case à sélectionner. Les mêmes perspectives que celles mentionnées à ce sujet pour Stylet sont également valables ici. Une autre critique majeure de cette technique d'interaction concernait la nécessité d'une interaction bi-manuelle pour réaliser la sélection. Nous envisagerons l'utilisation de la main dominante pour réaliser le geste tactile initial (smartphone en main) sur l'écran de la tablette suivi du geste de *stack*. Nous évaluerons ces variantes de la technique d'interaction dans nos futurs travaux.

La technique d'interaction Intersection est performante et appréciée par les participants. Cependant, elle peut induire certaines problématiques d'occultation des intitulés, notamment lors de la sélection en colonne. En effet, lors de cette sélection, le smartphone (placé face à l'utilisateur) peut occulter les intitulés de colonnes qui se trouvent derrière. En revanche, lors d'une sélection en ligne, le smartphone peut être placé sur la ligne désirée sur la partie droite de l'écran, n'occultant pas les intitulés de ligne situés sur le bord gauche. Nous envisageons la miniaturisation des composants des smartphones, les rendant toujours plus fins, comme solution à cette problématique d'occultation. En l'attente d'une telle évolution technologique, nous proposons de permettre une rotation du smartphone entre le geste de *stack* et d'*unstack*. Seule l'orientation initiale au moment du *stack* définissant le type de sélection (lignes ou colonnes), ces rotations permettront d'éviter les occultations d'intitulés sans modifier l'interaction. Nous implémenterons et évaluerons cette amélioration dans la suite de nos travaux.

### 8.2 Nouvelles fonctionnalités de sélection

L'interaction tactile classique pour un tableur sur tablette ne permet pas de sélectionner deux plages de données distinctes. Or cette fonctionnalité est disponible sur les versions de bureau des tableurs (en maintenant la touche contrôle

appuyée). Nous n'avons donc pas intégré cette tâche à notre étude. Cependant, l'utilisation du stacking permet ce type de tâche : il est par exemple envisageable d'utiliser les différentes arêtes du smartphone pour sélectionner un maximum de quatre plages différentes dans le tableur. Il est aussi envisageable d'utiliser l'écran du smartphone comme un "modifier" (similaire aux touches du clavier CTRL, ALT, ...) pour configurer le mode de sélection et ainsi permettre leur remplacement (i.e. créer une nouvelle sélection) ou leur ajout (i.e. ajouter la nouvelle sélection à la précédente).

### 8.3 Limitations de nos travaux

En premier lieu, notre étude est limitée par notre prototype. En effet, la fabrication ayant été manuelle, il est nécessaire d'appliquer avec le smartphone une certaine pression sur l'écran de la tablette afin que les points formant le motif d'arête soient détectés correctement. Ainsi, l'utilisation du smartphone peut induire une fatigue accrue. Nous avons, par conséquent, fabriqué une nouvelle coque imprimée en 3D afin de limiter les défauts de fabrication manuelle. De plus, notre algorithme de détection, assez robuste pour gérer la disparition temporaire d'un marqueur du motif, peut mal fonctionner sur les bordures de l'écran : lorsque l'utilisateur fait sortir plus de la moitié de l'arête de la surface de l'écran (par exemple, pour réaliser un défilement), la détection peut être perdue par le système. Nous planifions d'améliorer notre algorithme de détection en nous inspirant des systèmes de détection de type OptiTrack [25]. Ces outils permettent de suivre un modèle d'objet préenregistré et de les suivre au cours du temps. Dans le cas où certains points du modèle ne sont pas détectés, le système réalise une estimation de la position de ces points afin de conserver la détection.

D'autre part, les techniques d'interaction que nous avons proposées étant basées sur des gestes simples, aucun problème de mémorisation ou d'apprentissage n'a été relevé. Certains sujets ont même dit ressentir une charge cognitive plus faible lors de l'utilisation de la technique Intersection par rapport à l'interaction tactile classique. Or, le vocabulaire d'interaction offert par le stacking est bien plus important que la portion que nous avons explorée à travers les techniques Stylet, Intersection et Finger Stack. Il sera donc nécessaire d'évaluer l'impact de gestes complexes (notamment les gestes composés) sur l'apprentissage et la mémorisation des techniques afin de mieux cerner les apports du stacking à l'interaction avec un tableur sur tablette.

Pour finir, notre étude ne prend pas en compte les actions de pan et de zoom. En effet, nous avons souhaité limiter nos travaux à la sélection de plage uniquement, sans prendre en compte les potentielles stratégies de l'utilisateur en termes de défilement et de zoom, qui auraient pu créer des disparités entre les participants. Nous sommes cependant conscients que de telles stratégies pourraient impacter les techniques d'interaction que nous proposons. Une prochaine étape de notre travail consiste donc à inclure ces deux actions afin d'étendre la portée de nos résultats.

### 8.4 Perspectives

Notre étude préliminaire a mis en évidence un vocabulaire d'interaction riche amené par l'utilisation du stacking. Or, dans le cadre de notre étude sur la sélection de plage, nous n'utilisons qu'une petite partie de ce vocabulaire. En effet, certaines propriétés, telle que l'utilisation de l'écran du

smartphone pour afficher des informations complémentaires, ne sont pas pertinentes pour cette tâche. Nous n'avons pas non plus tiré avantage du *stack* des différentes arêtes du smartphone, de l'utilisation des inclinaisons ou du *stack* des coins. Nous planifions de concevoir de nouvelles techniques d'interaction impliquant ces mécanismes complémentaires dans nos futurs travaux.

Par ailleurs, notre étude se focalise sur la tâche de sélection de plage de données dans un tableur sur tablette. Cette tâche, préalable à l'activation des commandes du logiciel, n'est qu'une première étape de nos travaux. Dans nos futurs travaux, nous nous intéresserons à l'application de commandes liées à cette plage de sélection, telles que la création et manipulation de graphiques, des mises en forme, des calculs mathématiques ou l'activation de filtres. Nous tirerons profit des dimensions actuellement non utilisées du stacking afin de concevoir les techniques d'interaction associées à ces tâches.

En dernier lieu, nous souhaitons nous intéresser aux capacités de personnalisation apportées par le smartphone. En effet, dans cet article, nous supposons que le smartphone utilisé pour le stacking provient directement de l'utilisateur. Il est donc envisageable d'imaginer qu'un utilisateur pourrait, à l'aide d'une méta-interface, configurer son smartphone afin d'associer des manipulations et/ou zones tactiles à certaines actions : par exemple, associer le *stack* de l'arête gauche du smartphone aux commandes mathématiques et l'arête droite à la création de graphiques.

## 9 CONCLUSION

Dans cet article, nous explorons l'utilisation d'un second dispositif (e.g. le smartphone) afin de simplifier l'utilisation d'un tableur sur tablette. Nous utilisons le paradigme du stacking, qui consiste à mettre en contact l'une des arêtes du smartphone avec l'écran de la tablette. Cette solution permet de détecter la position et l'orientation du smartphone sans nécessiter de capteurs externes et élargit le vocabulaire d'interaction tactile classique à travers l'utilisation d'un interacteur tactile et tangible. Dans une étude préliminaire, nous avons exploré l'espace d'interaction lié au stacking entre une tablette et un smartphone afin d'en délimiter le périmètre. D'autre part, nous avons conçu une coque conductive permettant la détection du smartphone sur l'écran de la tablette sans nécessiter l'utilisation de capteurs externes aux deux dispositifs. Puis, nous avons conçu trois techniques d'interaction (Intersection, Finger Stack et Stylet) basées sur le stacking permettant de sélectionner des plages de données dans un tableur. Dans une première étude, nous avons évalué la performance de ces techniques et nous avons comparé ces résultats à la technique d'interaction tactile classique proposée par les tableurs sur tablette. Nous avons montré que les techniques Intersection et Finger Stack permettent de sélectionner plus rapidement des plages de données (Tactile est respectivement 16.1% et 19.9% plus lent) qu'avec l'interaction tactile classique.

## REMERCIEMENTS

Ces travaux ont été réalisés dans le cadre d'une collaboration de recherche IRIT/Berger-Levrault. Les auteurs remercient le club MAGEL et l'ensemble des personnes qui ont participé à l'étude (participants aux tests et élus).

## RÉFÉRENCES

- [1] Christopher Scaffidi, Mary Shaw, and Brad Myers. 2005. Estimating the Numbers of End Users and End User Programmers. In Proceedings of the 2005 IEEE Symposium on Visual Languages and Human-Centric Computing (VLHCC '05). IEEE Computer Society, Washington, DC, USA, 207-214. DOI=<http://dx.doi.org/10.1109/VLHCC.2005.34>
- [2] Mark Gardener. 2015. Managing Data Using Excel. Pelagic Publishing, , United Kingdom.
- [3] Brian Albright. 2009. Mathematical Modeling with Excel (1st ed.). Jones and Bartlett Publishers, Inc., , USA.
- [4] <https://play.google.com/store/apps/details?id=com.microsoft.office.excel&hl=fr> (Dernière visite 09/07/2017)
- [5] <https://play.google.com/store/apps/details?id=com.google.android.apps.docs.editors.sheets&hl=fr> (Dernière visite 09/07/2017)
- [6] Cecil Lozano, Devin Jindrich, and Kanav Kahol. 2011. The impact on musculoskeletal system during multitouch tablet interactions. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11). ACM, New York, NY, USA, 825-828. DOI=<http://dx.doi.org/10.1145/1978942.1979062>
- [7] Tableau software: <https://www.tableau.com/fr-fr> (Dernière visite 30/03/2017)
- [8] Sungjae Hwang, Myungwook Ahn, and Kwang-yun Wahn. 2013. MagGetz: customizable passive tangible controllers on and around conventional mobile devices. In Proceedings of the 26th annual ACM symposium on User interface software and technology (UIST '13). ACM, New York, NY, USA, 411-416. DOI: <http://dx.doi.org/10.1145/2501988.2501991>
- [9] Louis-Pierre Bergé, Emmanuel Dubois, and Mathieu Raynal. 2015. Design and Evaluation of an "Around the SmartPhone" Technique for 3D Manipulations on Distant Display. In Proceedings of the 3rd ACM Symposium on Spatial User Interaction (SUI '15). ACM, New York, NY, USA, 69-78. DOI: <http://dx.doi.org/10.1145/2788940.2788941>
- [10] Khalad Hasan, David Ahlström, and Pourang Irani. 2013. Ad-binning: leveraging around device space for storing, browsing and retrieving mobile device content. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13). ACM, New York, NY, USA, 899-908. DOI: <https://doi.org/10.1145/2470654.2466115>
- [10] Paweł Woźniak, Lars Lischke, Benjamin Schmidt, Shengdong Zhao, and Morten Fjeld. 2014. Thaddeus: a dual device interaction space for exploring information visualisation. In Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational (NordiCHI '14). ACM, New York, NY, USA, 41-50. DOI: <http://dx.doi.org/10.1145/2639189.2639237>
- [11] Tommaso Piazza, Morten Fjeld, Gonzalo Ramos, AsimEvren Yantac, and Shengdong Zhao. 2013. Holy smartphones and tablets, Batman!: mobile interaction's dynamic duo. In Proceedings of the 11th Asia Pacific Conference on Computer Human Interaction (APCHI '13). ACM, New York, NY, USA, 63-72. DOI=<http://dx.doi.org/10.1145/2525194.2525205>
- [12] Sang-won Leigh, Philipp Schoessler, Felix Heibeck, Pattie Maes, and Hiroshi Ishii. 2015. THAW: Tangible Interaction with See-Through Augmentation for Smartphones on Computer Screens. In Proceedings of the Ninth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '15). ACM, New York, NY, USA, 89-96. DOI: <https://doi.org/10.1145/2677199.2680584>
- [13] Ming Li and Leif Kobbelt. 2015. ACTUI: Using Commodity Mobile Devices to Build Active Tangible User Interfaces. In Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct (MobileHCI '15). ACM, New York, NY, USA, 592-597. DOI: <http://dx.doi.org/10.1145/2786567.2792895>
- [14] Gary Perelman, Marcos Serrano, Mathieu Raynal, Celia Picard, Mustapha Derras, and Emmanuel Dubois. 2016. DECO: A Design Space for Device Composition. In Proceedings of the 2016 ACM Conference on Designing Interactive Systems (DIS '16). ACM, New York, NY, USA, 435-446. DOI: <https://doi.org/10.1145/2901790.2901893>
- [15] Rafael Morales González, Caroline Appert, Gilles Bailly, and Emmanuel Pietriga. 2016. TouchTokens: Guiding Touch Patterns with Passive Tokens. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16). ACM, New York, NY, USA, 4189-4202. DOI: <https://doi.org/10.1145/2858036.2858041>
- [16] Neng-Hao Yu, Li-Wei Chan, Lung-Pan Cheng, Mike Y. Chen, and Yi-Ping Hung. 2010. Enabling tangible interaction on capacitive touch panels. In Adjunct proceedings of the 23rd annual ACM symposium on User interface software and technology (UIST '10). ACM, New York, NY, USA, 457-458. DOI=[10.1145/1866218.1866269](http://dx.doi.org/10.1145/1866218.1866269)
- [17] Capteurs tangibles : <http://www.appmatestoys.com/> (dernière visite 09/07/2017)
- [18] Liwei Chan, Stefanie Müller, Anne Roudaut, and Patrick Baudisch. 2012. CapStones and ZebraWidgets: sensing stacks of building blocks, dials and sliders on capacitive touch screens. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12). ACM, New York, NY, USA, 2189-2192. DOI: <http://dx.doi.org/10.1145/2207676.2208371>
- [19] Blagojevic R., Plimmer B. (2013) CapTUI: Geometric Drawing with Tangibles on a Capacitive Multi-touch Display. In: Kotzé P., Marsden G., Lindgaard G., Wesson J., Winckler M. (eds) Human-Computer Interaction – INTERACT 2013. INTERACT 2013. Lecture Notes in Computer Science, vol 8117. Springer, Berlin, Heidelberg
- [20] Monya Baker. 2015. Statisticians issue warning over misuse of P values. Nature 531, 7593 (March 2015), 151. DOI:<http://dx.doi.org/10.1038/nature.2016.19503>
- [21] Geoff Cumming. 2014. The new statistics: Why and how. Psychological Science 25, 1 (Jan. 2014), 7–29. DOI: <http://dx.doi.org/10.1177/0956797613504966>
- [22] Pierre Dragicevic. 2016. Fair statistical communication in HCI. In Modern Statistical Methods for HCI, Judy Robertson and Maurits Kaptein (Eds.). Springer International Publishing, Cham, Switzerland, Chapter 13, 291–330. DOI: [http://dx.doi.org/10.1007/978-3-319-26633-6\\_13](http://dx.doi.org/10.1007/978-3-319-26633-6_13)
- [23] Pierre Dragicevic, Fanny Chevalier, and Stephane Huot. 2014. Running an HCI experiment in multiple parallel universes. In Extended Abstracts on Human Factors in Computing Systems. ACM, New York, 607–618. DOI: <http://dx.doi.org/10.1145/2559206.2578881>
- [24] Gary R. VandenBos (Ed.). 2009. Publication Manual of the American Psychological Association (6th ed.). American Psychological Association, Washington, DC. <http://www.apastyle.org/manual/>
- [25] Système de detection OptiTrack: <http://optitrack.com/hardware/> (Dernière visite 09/07/2017)

